Confidence and Venn machines and Their Applications to Proteomics

Dmitry Devetyarov

Computer Learning Research Centre and
Department of Computer Science,
Royal Holloway, University of London,
United Kingdom

2011

A dissertation submitted in fulfilment of the degree of Doctor of Philosophy.

Declaration

I declare that this dissertation was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Dmitry Devetyarov

Supervisor: *Prof Alex Gammerman*

Abstract

When a prediction is made in a classification or regression problem, it is useful to have additional information on how reliable this individual prediction is. Such predictions complemented with the additional information are also expected to be valid, i.e., to have a guarantee on the outcome. Recently developed frameworks of confidence machines, category-based confidence machines and Venn machines allow us to address these problems: confidence machines complement each prediction with its confidence and output region predictions with the guaranteed asymptotical error rate; Venn machines output multiprobability predictions which are valid in respect of observed frequencies. Another advantage of these frameworks is the fact that they are based on the i.i.d. assumption and do not depend on the probability distribution of examples. This thesis is devoted to further development of these frameworks.

Firstly, novel designs and implementations of confidence machines and Venn machines are proposed. These implementations are based on random forest and support vector machine classifiers and inherit their ability to predict with high accuracy on a certain type of data. Experimental testing is carried out.

Secondly, several algorithms with online validity are designed for proteomic data analysis. These algorithms take into account the nature of mass spectrometry experiments and special features of the data analysed. They also allow us to address medical problems: to make early diagnosis of diseases and to identify potential biomarkers. Extensive experimental study is performed on the UK Collaborative Trial of Ovarian Cancer Screening data sets.

Finally, in theoretical research we extend the class of algorithms which output valid predictions in the online mode: we develop a new method of constructing valid prediction intervals for a statistical model different from the standard i.i.d. assumption used in confidence and Venn machines.

Acknowledgements

I am grateful to my supervisor Alex Gammerman for suggesting the original subject of the thesis and providing constant support during my research. I would also like to thank Martin J. Woodward, Nicholas G. Coldham and Muna F. Anjum from Veterinary Laboratories Agency of DEFRA for their collaboration and co-supervision. I am very grateful to Volodya Vovk for directions and help in theoretical research, and Ilia Nouretdinov for his support and fruitful discussions regarding my work.

I also thank other members of the Computer Learning Research Centre, especially Alexey Chervonenkis, Zhiyuan Luo, Brian Burford, Mikhail Dashevsky and Fedor Zhdanov, who collaborated on different projects during my PhD. Thanks to all fellow PhD student of the Department of Computer Science of Royal Holloway, University of London, for supportive and friendly environment.

This work was funded by the VLA grant "Development and Application of Machine Learning Algorithms for the Analysis of Complex Veterinary Data Sets". I am grateful to Adriana Gielbert, Maurice Sauer and Luke Randall from VLA for providing data for experimental studies.

I would like to thank our collaborators in MRC project "Proteomic Analysis of the Human Serum Proteome" Ian Jacobs, Usha Menon, Rainer Cramer, John F. Timms, Ali Tiss, Jeremy Ford, Stephane Camuzeaux, Aleksandra Gentry-Maharaj, Rachel Hallett, Celia Smith, Mike Waterfield for collecting the original UKCTOCS and UKOPS data and carrying out mass spectrometry experiments.

Finally, I am grateful to the Computer Science department for financial support, which made it possible to present results of my work at conferences.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In many applications of machine learning, it is crucial to know how reliable predictions are rather than have predictions without any estimation of their accuracy. For example, in medical diagnosis, this would give practitioners a reliable assessment of risk error; in drug discovery, such control of the error rate in experimental screening is also desirable since the testing is expensive and time consuming.

In addition, it would be useful to obtain information regarding how strongly we believe in each individual prediction rather than a whole group of predictions for all objects. We will call complementing a prediction with such additional information *hedging a prediction*. In medical diagnosis, this would allow us to distinguish more confident predictions from uncertain ones; in drug discovery, this would make it possible to select compounds that are more likely to exhibit bio-active behaviour for further experimental screening.

In this thesis, we are interested in machine learning algorithms which address both of these problems: they provide a guarantee on the overall outcome and hedge each individual prediction (provide additional information regarding how strongly we believe in it).

There are different approaches that allow users to assess total accuracy or hedge each prediction. Among them are such well known ones as statistical learning theory (probably approximately correct learning, or PAC learning), Bayesian learning and hold-out estimates.

In PAC learning [61], we can preset a small probability δ and have a theoretical guarantee that with a probability at least $(1 - \delta)$ predictions will be wrong in at most ϵ cases, where the error rate bound ϵ is calculated depending on δ. However, these bounds of error may often be useless as ϵ is likely to be very large. Only problems with the large number of objects can benefit from PAC learning application. In addition, PAC learning does not provide any information on the reliability of a prediction for each object.

In contrast, Bayesian learning [28] and other probabilistic algorithms may complement each individual prediction with such additional information. However, the main disadvantage of these algorithms is that they often depend on strong statistical assumptions used in the model. When the data conforms well with the statistical model, Bayesian learning outputs valid predictions. However, if the data do not match the model (or the a priori information is not correct), which is usually the case for real-world data, predictions may become invalid and misleading ([65], Section 10.3).

This thesis focuses on machine learning frameworks of confidence and Venn machines, which were introduced in [65] and represent a new generation of hedged prediction algorithms. These newly developed methods have several advantages. Firstly, they hedge every prediction individually rather than estimate an error on all future examples as a whole. As a result, the supplementary information which is assigned to predictions and reflects their reliability is tailored not only to the previously seen examples (a training set) but also to the new object. Secondly, both frameworks of confidence and Venn machines produce valid results. *Validity* is an important property of algorithms and in this case has a form of a guarantee that the error rate of region predictions converges to or is bounded by a pre-defined level when the number of observed examples tends to infinity or that a set of output probability distributions on the possible outcomes agrees with observed frequencies. The property of validity is based on a simple i.i.d. assumption (that examples are independent and identically distributed) or a weaker exchangeability assumption and does not depend on the probability distribution of examples. The latter assumption can be often satisfied when data sets are randomly permuted. The property of validity is theoretically proved [65] in the online mode, when examples are given one by one and the prediction is made on the basis of the preceding examples. Throughout this thesis, we will refer to the class of confidence machines (together with their modification, category-based confidence machines) and Venn machines as *algorithms with online validity.*

Most of the methods considered in this thesis are based on the i.i.d. assumption. However, in Chapter 5 we extend a class of algorithms with online validity beyond this assumption. Our first move in this direction is a new algorithm with the property of validity based on the following model: a linear regression model with the i.i.d. errors with a known distribution.

Let us first briefly cover algorithms with online validity based on the i.i.d. assumption.

Confidence machines [24; 65] and category-based confidence machines [65; 66] allow us to assign confidence to each individual prediction. This notion of confidence should not be confused with confidence in statistical conclusions with confidence intervals (see Section 2.2.2 for details).

Confidence machines are region predictors: in the event that a unique prediction cannot be achieved with required confidence, the method outputs a set (region) of possible labels. We will call such region prediction erroneous if it does not contain a true label. The main advantage of confidence machines is their property of validity: the rate of erroneous region predictions does not asymptotically exceed the preset value ϵ, called significance level. Please note that here and every time when referring to the error rate or accuracy of confidence machines, we imply the error rate of region predictions rather than singleton predictions. Confidence machines can also be forced to output singleton predictions, but in this case we will refer to *forced accuracy.*

Category-based confidence machines, which are the development of confidence machines, allow us to split all possible examples (combinations of an object and a label) into several categories and set significance levels ϵ_k, one for each category k. Category-based confidence machines can guarantee that asymptotically we make errors on objects of type k with frequency at most ϵ_k. Again, by errors we imply region, not singleton, predictions that do not contain a true label.

Thus, category-based confidence machines allow us to tackle the following problems.

Firstly, we can guarantee not only an overall accuracy in terms of region predictions but also a certain level of accuracy within each category of examples. In particular, in medical diagnosis we can preset the level of accuracy within groups of healthy and diseased samples, which is similar to controlling specificity and sensitivity. This will allow avoiding classifications when low region specificity is compensated by high region sensitivity, or the other way around.

Secondly, if we preset different significance levels in different categories, we can treat accuracy within these categories in a different way. E.g., in medical diagnosis, we can put region sensitivity or specificity first and consider misclassification of a diseased sample more serious that misclassification of a healthy sample.

Thus, confidence machines and category-based confidence machines output a set of possible labels for a new object. In different applications, it can be more useful to predict a probability of a label; e.g., in medicine clinicians may need to predict the probability of a disease. There is a range of methods that can output a probability distribution of a new label. However, these methods are usually based on strong statistical assumptions about example distribution. Hence, if the assumed statistical model is not correct, predicted probabilities may be incorrect too. We suggest producing a set of probability distributions by the use of another framework — Venn machines [65; 67]. A Venn machine outputs several probability distributions, one for each candidate label. This output is called *multiprobability prediction*. Similarly to confidence machines, Venn machines are valid regardless of the example distribution: the only assumption made is i.i.d.

Confidence machines, category-based confidence machines and Venn machines are not single algorithms but flexible frameworks: each of them depends on a core element, and practically any machine learning method can be used to define this core element (it is called an *underlying algorithm* in this case). These core elements are a strangeness measure for confidence machines; a strangeness measure and a taxonomy for category-based confidence machines; a Venn taxonomy for Venn machines. Thus, the framework can give rise to a set of different algorithms which may potentially perform well on different types of data.

This thesis covers several problems but all of them are devoted to development of algorithms with online validity.

The first area of research is devoted to novel designs and implementations of such algorithms. Algorithms with online validity are flexible: practically any known machine learning algorithm can be used as an underlying algorithm. While these algorithms output valid predictions, the question is how informative these predictions are. For example, if the confidence machine outputs all possible labels as a prediction, this prediction is vacuous. We refer to how well an algorithm can make informative predictions as *efficiency*. Algorithms with online validity usually inherit advantages of their underlying algorithms, and their efficiency tends to be in line with accuracy of the underlying algorithm and therefore varies across the range of underlying algorithms and also depends on the type of data analysed. For this reason, it is crucial to develop new implementations of algorithms with online validity that could result in efficient predictions.

In this research we focused on random forest and support vector machine (SVM) classifiers as underlying algorithms since both of them proved to perform well on certain types of data. We designed confidence and Venn machines to inherit the abilities of SVMs and random forests to perform with high accuracy on many data sets. As a result, we developed several new strangeness measures derived from random forests (which could be used in confidence machines or category-based confidence machines), several versions of Venn taxonomies based on random forests and a few implementations of Venn taxonomies which deploy SVMs. Some of these algorithms were applied to the analysis of microarray data of Salmonella provided by the Veterinary Laboratories Agency (VLA) of the Department for Environment, Food and Rural Affairs. The results are provided in Appendix C.

Another big part of research investigates application of algorithms with online validity to data from mass spectrometry experiments, which represent an attractive analytical method in clinical proteomic research.

The aim of this investigation was to develop algorithms which, on the one hand, could hedge predictions by providing a measure of reliability tailored to each individual patient and, on the other hand, are adjusted to the analysis of mass spectrometry data. These algorithms take into account the nature of mass spectrometry experiments and format of mass spectrometry data as well as special features of the data we analysed. After pre-processing is applied, mass spectrometry data are represented by intensities of mass spectrometry profile peaks, some of which can be crucial for different medical and veterinary problems. Our methods could help identify profile peaks which would allow solving such problems.

Originally, the algorithms designed in this thesis for mass spectrometry data analysis were applied to the veterinary data provided by VLA. The objective of this study was to differentiate the vaccine Salmonella strains from wild type strains of the same serotype (see Appendix C for data description and the analysis results). However, the sample size was not big enough; therefore, to illustrate our algorithms we carried out experiments on the data of the UK Collaborative Trial of Ovarian Cancer Screening (UKCTOCS). The results of these experiments are presented in Chapter 3.

The UKCTOCS data pertains to mass spectrometry samples taken from women diagnosed with ovarian cancer, breast cancer or heart disease, and healthy controls. The advantage of these data is that for each diseased sample it is known how long in advance of the moment of clinical diagnosis or the moment of death it was taken. In addition, for ovarian cancer, we can also observe the dynamics of diseased patients: the data comprises serial measurements taken at different moments from the same ovarian cancer patients.

These features of the UKCTOCS data allow us to investigate more complex issues and investigate a problem of early diagnosis of diseases. Therefore, we aimed at developing methods which would be able to contribute to medical research and to answer the following questions:

- How early in advance of the moment of clinical diagnosis / the moment of death can we make reliable predictions of disease diagnosis?

- Which mass spectrometry profile peaks carry information important for identifying diseased patients and could be potential biomarkers for early diagnosis of diseases?

We are interested in the answer to the first question because, for such diseases as ovarian cancer, it is crucial to identify the disease as soon as possible: if ovarian cancer is diagnosed at the early stage, it may be possible to cure the patient. Thus, we are aiming at designing the methodology which would allow us to determine how well in advance of the moment of diagnosis/death we can make reliable diagnosis predictions.

The second question is important since the identification of informative mass spectrometry profile peaks would reduce the amount of work and time required to make a new prediction.

Thus, the main thrust of the work presented in this thesis is devoted to development of the frameworks of confidence, category-based confidence and Venn machines, all of which are based on the i.i.d. assumption. However, as it was mentioned earlier, in the final part of the thesis, we consider a statistical model different from the standard i.i.d. assumption and extend the class of algorithms with online validity. We design a new algorithm of constructing prediction intervals for the linear regression model with the i.i.d. error with a known distribution but not necessarily Gaussian. Even though this algorithm is not based on the i.i.d. assumption, it has the property of validity similar to the property of validity of confidence machines: in the online mode the errors made by prediction intervals are independent of each other and are made with the same probability equal to the significance level.

The code for implemented algorithms can be found on `http://clrc.rhul.ac.uk/publications/techrep.htm`.

1.2 Main Contributions

The following theoretical and experimental results were achieved during the work on this thesis.

1.2.1 Design of Algorithms with Online Validity

New implementations of known algorithms with online validity were designed. Among them are:

- several strangeness measures based on random forests, which can be used in confidence machines and category-based confidence machines
- several versions of Venn taxonomies derived from random forests
- several versions of Venn taxonomies based on SVMs

We performed extensive experimental study on different data sets (including Salmonella microarray data provided by VLA) to ensure that proposed algorithms are applicable; we further gave recommendations on their use.

1.2.2 Algorithms with Online Validity for Proteomics

Several algorithms with online validity were developed for mass spectrometry data analysis. These algorithms take into account the nature of mass spectrometry experiments, the format of mass spectrometry data and special features of the analysed data: serial samples and triplet setting. In addition, they allow us to pinpoint important mass spectrometry profile peaks, which could be potential biomarkers for early diagnosis of diseases.

The designed algorithms are the following:

- a category-based confidence machine with the strangeness measure based on linear rules
- a Venn machine with the Venn taxonomy derived from logistic regression (developed in collaboration with Ilia Nouretdinov)
- confidence machines in the triplet setting

Extensive experimental study was performed on the UKCTOCS data sets in order to confirm algorithm applicability. The methods were also applied to the mass spectrometry data provided by VLA (see Appendix C).

Besides application of algorithms with online validity, we carried out other types of analysis of mass spectrometry data:

- triplet statistical analysis of serial samples of the UKCTOCS ovarian cancer data set (see Appendix B)
- machine learning analysis of the UK ovarian cancer population study (UKOPS) data [56; 59]

All these studies allowed us to make tentative conclusions related to medical research. Firstly, we achieved good classification results on experimental mass spectrometry data of ovarian cancer and breast cancer. Secondly, proposed methodologies allowed us to estimate how long in advance we can output accurate predictions for these diseases. Thirdly, developed algorithms with online validity confirmed mass spectrometry profile peaks which were identified in the triplet analysis as carrying statistically significant information for discrimination between healthy and diseased patients. These mass spectrometry profile peaks could be potential biomarkers.

1.2.3 An Algorithm with Online Validity in the Linear Regression Model

A new method of constructing region predictions for the linear regression model with the i.i.d. error with a known distribution, not necessarily Gaussian, was designed. The method has the property of validity. The coverage probability of prediction intervals is equal to the preset confidence level not only unconditionally but also conditionally given a natural σ-algebra of invariant events. As a result, in the online mode the errors made by prediction intervals are independent of each other and are made with the same probability equal to the significance level. The experiments were carried out on artificially generated data and the real-world ChickWeight data ([14], Example 5.3; [30], Table A.2).

My contribution to this research comprises a proof of Lemma 5.1, which made the construction of prediction intervals consistent, and computational experiments laid out in Section 5.7.

1.3 Publications

Research covered in this thesis was presented at various conferences and resulted in a number of publications. This is a list of the publications in chronological order.

1. Dmitry Devetyarov, Ilia Nouretdinov and Alex Gammerman. Confidence machine and its application to medical diagnosis. *Proceedings of the 2009 International Conference on Bioinformatics and Computational Biology*, pages 448–454, 2009.

2. Fedor Zhdanov, Vladimir Vovk, Brian Burford, Dmitry Devetyarov, Ilia Nouretdinov and Alex Gammerman. Online prediction of ovarian cancer. *Proceedings of the 12th Conference on Artificial Intelligence in Medicine*, pages 375-379, 2009.

3. Dmitry Devetyarov. Machine learning analysis of proteomics data for early diagnostic. *Proceedings of the Medical Informatics Europe (MIE) Conference*, page 772, 2009.

4. Peter McCullagh, Vladimir Vovk, Ilia Nouretdinov, Dmitry Devetyarov and Alex Gammerman. Conditional prediction intervals for linear regression. *Proceedings of the 8th International Conference on Machine Learning and Applications (ICMLA 2009)*, pages 131–138, 2009.

5. John F. Timms, Rainer Cramer, Stephane Camuzeaux, Ali Tiss, Celia Smith, Brian Burford, Ilia Nouretdinov, Dmitry Devetyarov, Aleksandra Gentry-Maharaj, Jeremy Ford, Zhiyuan Luo, Alex Gammerman, Usha Menon and Ian Jacobs. Peptides generated ex vivo from abundant serum proteins by tumour-specific exopeptidases are not useful biomarkers in ovarian cancer. *Clinical Chemistry.* 56:262–271, 2010.

6. Dmitry Devetyarov, Martin J. Woodward, Nicholas G. Coldham, Muna F. Anjum, Alex Gammerman, A new bioinformatics tool for prediction with confidence. *Proceedings of the 2010 International Conference of Bioinformatics and Computational Biology*, pages 24–28, 2010.

7. Dmitry Devetyarov, Ilia Nouretdinov. Prediction with confidence based on a random forest classifier. *Proceedings of the 6th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI 2010)*, pages 37–44, 2010.

8. Ali Tiss, John F. Timms, Celia Smith, Dmitry Devetyarov, Aleksandra Gentry-Maharaj, Stephane Camuzeaux, Brian Burford, Ilia Nouretdinov, Jeremy Ford, Zhiyuan Luo, Ian Jacobs, Usha Menon, Alex Gammerman and Rainer Cramer. Highly accurate detection of ovarian cancer using CA125 but limited improvement with serum MALDI-TOF MS profiling. *International Journal of Gynecological Cancer.* 2010, in press.

The following papers are currently under review for publication:

1. Dmitry Devetyarov, Ilia Nouretdinov, Brian Burford, Stephane Camuzeaux, Alex Gentry-Maharaj, Ali Tiss, Celia Smith, Zhiyuan Luo, Alexey Chervonenkis1, Rachel Hallett, Volodya Vovk, Mike Waterfield, Rainer Cramer, John F. Timms, Ian Jacobs, Usha Menon and Alex Gammerman. Prediction with Confidence prior to Cancer Diagnosis. Submitted to *International Journal of Proteomics.*

2. John F. Timms, Usha Menon, Dmitry Devetyarov, Ali Tiss, Stephane Camuzeaux, Katherine McCurrie, Ilia Nouretdinov, Brian Burford, Celia Smith, Aleksandra Gentry-Maharaj, Rachel Hallett, Jeremy Ford, Zhiyuan Luo, Volodya Vovk, Alex Gammerman, Rainer Cramer and Ian Jacobs. Early detection of ovarian cancer in pre-diagnosis samples using CA125 and MALDI MS peaks. Submitted to *Gynecologic Oncology.*

Some results were also presented in a poster presentation at the Proteomic Forum 2009 in Berlin: Dmitry Devetyarov, Zhiyuan Luo, Nick Coldman, Muna Anjum, Martin Woodward, Alex Gammerman, “Machine learning data analysis of TSE proteomic data”.

1.4 Outline of the Thesis

This introductory chapter has given the motivation behind the research carried out in this thesis and has briefly described areas of research. The main contributions and publications have also been summarised.

The rest of the thesis is organised as follows.

Chapter 2 gives the background of the problem. It is devoted to known algorithms with online validity (confidence machines, category-based confidence machines and Venn machines) and compares them to other algorithms which hedge predictions or estimate algorithm accuracy.

In Chapter 3, new implementations of algorithms with online validity are proposed and investigated: confidence machines constructed by the use of random forests, Venn machines based on random forests and Venn machines with a taxonomy derived from an SVM.

In Chapter 4, we design and apply methodologies which provide valid predictions for mass spectrometry data analysis.

Chapter 5 extends the class of algorithms with online validity and introduces a new interval predictor which has the property of exact validity under the linear regression model with i.i.d. errors with a known distribution.

Chapter 6 gives the conclusion to the thesis, outlines its main contributions and provides directions for further research.

In Appendix, the reader can find additional experimental results, the triplet analysis of the UKCTOCS ovarian cancer data set and results of the application of algorithms with online validity to the data sets provided by VLA.

Chapter 2

Overview of Algorithms with Online Validity

In this chapter, we describe known algorithms which estimate algorithm accuracy or hedge each individual prediction complementing it with additional information about how strongly we trust it.

Firstly, we cover the methods we are focusing on in this thesis: confidence machines [24; 65] and category-based confidence machines [65; 66], which output region predictions, as well as Venn machines [65; 67], which output multiprobability predictions. We unite these methods under the term of algorithms with online validity. We give precise definitions and describe related notions that will be used throughout the thesis. We explain how performance of their predictions is measured by means of validity and efficiency and what guarantees are provided by these methods. We also show some implementations.

In addition, we demonstrate advantages of frameworks with online validity: we compare them with other known approaches that estimate overall accuracy or hedge individual predictions, including confidence intervals, statistical learning theory and probabilistic approaches.

2.1 Algorithms with Online Validity

Most of the definitions and notation presented in this section follow [65], where algorithms with online validity were proposed and described in detail.

2.1.1 The Problem and Assumptions

Throughout the thesis, we consider the problem laid out below.

Let us assume that we are given a training set of successive pairs

$$(x_1, y_1), \ldots, (x_{n-1}, y_{n-1}),$$

which are called *examples*. Each example consists of an *object* $x_i \in \mathbf{X}$ (a vector of attributes) and a *label* $y_i \in \mathbf{Y}$. Objects are elements of a measurable space $\mathbf{X}$ called the *object space*, labels are elements of a measurable space $\mathbf{Y}$ called the *label space*. We denote examples by $z_i = (x_i, y_i)$, and they are elements of a measurable space $\mathbf{Z} = \mathbf{X} \times \mathbf{Y}$ called the *example space*.

Finally, we are given a new object x_n and are later announced its label y_n. Our general goal is to predict the label y_n for x_n.

According to the type of the label space, the problem usually falls into one of the following two categories: classification and regression. If the space of labels consists of a finite number of labels, that is, $\mathbf{Y} = \{y_n\}, n = 1, \ldots, N$, this problem is called a *classification* problem. This category includes problems of medical diagnosis and hand-written digit recognition. The problem of predicting a label out of a set of real numbers ($\mathbf{Y} = \mathbb{R}$) is called *regression*. This type of problems is considered in stock price prediction and many econometric problems. There are problems different from both classification and regression (for instance, ordinal regression), but we are not considering them in this thesis.

To construct a reliable algorithm, we need to make some assumptions on the data generating mechanism. Our standard assumption used in the most of the thesis (for confidence machines, category-based confidence machines and Venn machines) is the *i.i.d.* assumption. The examples z_i are assumed to be generated independently by the same probability distribution P on $\mathbf{Z}$, i.e., the infinite sequence of examples $z_1, z_2, \ldots$ is drawn from the power probability distribution P^∞ on $\mathbf{Z}^\infty$ ($\mathbf{Z}^\infty$ is the set of all infinite sequences of elements of $\mathbf{Z}$).

Usually the assumption which is needed is slightly weaker. This is the *exchangeability* assumption that the infinite sequence $z_1, z_2, \ldots$ is drawn from the probability distribution Q on $\mathbf{Z}^\infty$, which is exchangeable. This means that for every positive integer n, every permutation π of $\{1, \ldots, n\}$ and every measurable set $E \subseteq \mathbf{Z}^N$,

$$\begin{aligned} &P\{(z_1, z_2, \ldots) \in \mathbf{Z}^\infty : (z_1, \ldots, z_n) \in E\} \\ &\qquad = P\{(z_1, z_2, \ldots) \in \mathbf{Z}^\infty : (z_{\pi(1)}, \ldots, z_{\pi(n)}) \in E\}\,. \end{aligned} \tag{2.1}$$

Both exchangeability and i.i.d. assumption are much weaker than most probabilistic assumptions since we do not require to know the distribution itself. The exchangeability assumption can be often satisfied when data sets are randomly permuted.

2.1.2 Confidence Machines

If in a problem of classification or regression we simply attempt to predict a label for a new object, we look for a function of the type

$$F : \mathbf{Z}^* \times \mathbf{X} \to \mathbf{Y},$$

which we call a *simple predictor.* Such predictor for any finite sequence of labelled objects $(x_1, y_1), \ldots, (x_{n-1}, y_{n-1})$ and a new object x_n without a label outputs $F(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n)$ as its prediction for a new label y_n.

However, as it was mentioned in the introduction, it is useful to have any information regarding how much we trust this predictions. For this reason, we would like a predictor to output a range of predicted labels, each one complemented with a degree of its reliability. Such predictor would output smaller subsets of the label space which it finds less reliable and bigger subsets which are more reliable. This can be achieved by the use of confidence machines, whose framework was introduced and described in detail in [24; 65]. Here we lay out the basic concepts and mostly follow the notation used in these publications.

According to the type of their output, confidence machines are *confidence predictors* rather than simple predictors. Confidence predictors have an additional parameter $\epsilon \in (0, 1)$ called the *significance level.* Its complementary value $1 - \epsilon$ is called the *confidence level* and reflects our confidence in the prediction. Confidence predictor for any given finite sequence of labelled objects $(x_1, y_1), (x_2, y_2), \ldots$, a new object x_n without a label and significance level ϵ outputs a subset of the label space:

$$\Gamma^{\epsilon}(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n),$$

so that

$$\Gamma^{\epsilon_1}(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n) \subseteq \Gamma^{\epsilon_2}(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n) \qquad (2.2)$$

for any $\epsilon_1 \geq \epsilon_2$. This means that prediction regions for different ϵ represent nested subsets of $\mathbf{Y}$ and by changing the significance level ϵ we can regulate the size of the output prediction.

Thus, a confidence predictor is a measurable function $\Gamma : \mathbf{Z}^* \times \mathbf{X} \times (0, 1) \to 2^{\mathbf{Y}}$ that satisfies (2.2) for all significance levels $\epsilon_1 \geq \epsilon_2$, all $n \in \mathbb{N}$ and all data sequences $x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n$.

We say that a confidence predictor makes an *erroneous prediction* if the output region $\Gamma^{\epsilon}(x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n)$ does not contain a true label y_n. When the error rate or accuracy of confidence predictors is mentioned in this thesis, we imply errors made by region predictions rather than singleton predictions.

The main advantage of confidence machines is their property of *validity*: the asymptotic number of errors, that is, erroneous region predictions, can be controlled by the significance level — the error rate we are ready to tolerate which is predefined by the user (the prediction is considered to be erroneous if it does not contain the true label). All precise definitions will be given later.

However, the property of validity is achieved at the cost of producing *region predictions*: instead of outputting a single label as a prediction, we may produce several of them any of which may be correct. Predictions that contain no labels are called *empty predictions*, those that contain one label are called *certain predictions*, and those comprising more than one label *multiple predictions.* Such multiple predictions are not mistakes: they are output when the confidence machine is not provided with sufficient information for producing valid predictions at a certain error rate. Informativeness, or in other words *efficiency*, of a confidence machine can be translated as its ability to produce as small region predictions as possible. Thus, we have to balance validity (the error rate) and efficiency (the number of labels in each prediction): lower error rates will result in larger region predictions, and vice versa. This feature makes confidence machines a very flexible tool.

2.1.2.1 Definitions

The general idea of confidence machines is to try every possible label y as a candidate for x_n's label and see how well the resulting pair (x_n, y) conforms with $(x_1, y_1), \dots, (x_{n-1}, y_{n-1})$. The ideal case is when exactly one y conforms with the rest of the sequence and all others do not — we can then be confident in this prediction.

First, we need to define the notion of a strangeness measure, which is the core of confidence machines. *A strangeness measure* is a set of measurable mappings $\{A_n : n \in N\}$ of the type

$$A_n : \mathbf{Z}^{(n-1)} \times \mathbf{Z} \to (-\infty, +\infty]\,,$$

where $\mathbf{Z}^{(n-1)}$ is the set of all bags (multisets) of elements of $\mathbf{Z}$ of size $n-1$. This strangeness measure will assign a *strangeness score* $\alpha_i \in \mathbb{R}$ to every example in the sequence $\{z_i,\ i = 1, \dots, n\}$ including a new example and will evaluate its 'strangeness' in comparison with the rest of the data:

$$\alpha_i := A_n(\wr z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_n \wr, z_i),\ i = 1, \dots, n\,, \tag{2.3}$$

where $\wr \dots \wr$ denotes a multiset. A specific strangeness measure A_n depends on a particular algorithm to be used and can be based on many well-known machine learning algorithms.

When considering a hypothesis $y_n = y$ and after finding the corresponding strangeness scores $\alpha_1, \dots, \alpha_n$ for a full sequence with label y for the last example, a natural way to compare α_n to the other α_is is to look at the ratio of examples that are as least as strange as the new example, that is, to calculate

$$p_n(y) = \frac{|\{i = 1, \dots, n : \alpha_i \geq \alpha_n\}|}{n}.$$

This ratio is called the p-value associated with the possible label y for x_n. Thus, we can compliment each label with a p-value that shows how well the example with this label conforms with the rest of the sequence in comparison with other objects in the sequence.

Finally, the p-values calculated above can produce a confidence predictor: the *confidence machine* determined by the strangeness measure $A_n, n \in N$ and a significance level ϵ is a measurable function

$$\Gamma : \mathbf{Z}^* \times \mathbf{X} \times (0, 1) \to 2^{\mathbf{Y}}$$

($2^{\mathbf{Y}}$ is a set of all subsets of $\mathbf{Y}$) that defines the prediction set $\Gamma^{(\epsilon)}(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n)$ as the set of all labels $y \in \mathbf{Y}$ such that $p_n > \epsilon$. Thus, for any finite sequence of examples with labels, $(x_1, y_1, \dots, x_{n-1}, y_{n-1})$, a new object without a label x_n and a significance level ϵ, the confidence machine outputs a region prediction $\Gamma^{(\epsilon)}$ — a set of possible labels for a new object.

Confidence machines defined above are *conservatively valid* [65, Section 2.1]. To explain what it means, we need to introduce some formal notation. Let $\omega = (x_1, y_1, x_2, y_2, \dots)$ denote the infinite sequence of examples. Let us express the fact of making an erroneous prediction as a number:

$$err_n^\epsilon(\Gamma, \omega) := \begin{cases} 1 & \text{if } y_n \notin \Gamma^\epsilon(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n), \\ 0 & \text{otherwise.} \end{cases}$$

If ω is drawn from a probability distribution P, which is assumed to be exchangeable, the error at the n-th step $err_n^\epsilon(\Gamma, \omega)$ is the realised value of a random variable, which may be denoted by $err_n^\epsilon(\Gamma, P)$.

Confidence predictor is called *conservatively valid* if for any exchangeable probability distribution P on $\mathbf{Z}^\infty$ there exist a probability distribution with two families

$$(\xi_n^{(\epsilon)} : \epsilon \in (0,1), n = 1,2,\dots),\ (\eta_n^{(\epsilon)} : \epsilon \in (0,1), n = 1,2,\dots)$$

of $\{0,1\}$-valued random variables such that:

- for a fixed ϵ, $\xi_1^{(\epsilon)}$, $\xi_2^{(\epsilon)}$, ... is a sequence of independent Bernoulli variables with parameter ϵ, i.e., the sequence of independent random variables each of which is equal to one with probability ϵ and zero with probability $1-\epsilon$;
- $\eta_n^{(\epsilon)} \leq \xi_n^{(\epsilon)}$ for all n and ϵ;
- the joint distribution of $err_n^\epsilon(\Gamma, P)$, $\epsilon \in (0,1)$, $n = 1,2,\dots$, coincides with the joint distribution of $\eta_n^{(\epsilon)}$, $\epsilon \in (0,1)$, $n = 1,2,\dots$.

To put it simply, a confidence predictor is conservatively valid if it is dominated in distribution by a sequence of independent Bernoulli random variables with parameter ϵ.

It can be shown [65, Proposition 2.2] that the property of conservative validity leads to the property of asymptotical conservativeness: asymptotically, the frequency of errors made by a confidence machine (that is, cases when the prediction set Γ^ϵ does not contain a real label) does not exceed ϵ subject to the i.i.d. assumption. Strictly speaking, confidence predictor is called *asymptotically conservative* if for any exchangeable probability distribution P on $\mathbf{Z}^\infty$ and any significance level ϵ,

$$\limsup_{n\to\infty} \frac{\sum_{i=1}^n err_n^\epsilon(\Gamma)}{n} \leq \epsilon$$

with probability one.

It is shown in [65] that all confidence machines are conservatively valid and therefore asymptotically conservative. Throughout this thesis, when using the term *validity* with respect to confidence machines, we will imply both properties of conservative validity and asymptotical conservativeness (since the latter is the consequence of the former).

The property of validity is proved only for the *online mode*, that is, when we observe each example one by one and make each prediction taking into account the information only regarding the examples considered before rather than predict on the basis of a certain rule extracted from a fixed set of examples. The latter setting is called the *offline mode*. Nevertheless, validity is empirically proved to remain in the offline mode [65].

For each individual object, it is possible to choose such a significance level that the confidence machine outputs a singleton prediction. It is equivalent to predicting a single label with the highest p-value (as for now let us assume that there exist the only highest p-value). However, in this case significance levels will vary across the range of objects and the property of validity will not hold. We will refer to this alternative way of presenting the results of confidence machine application as *forced prediction* and will use it in this thesis for artificial comparison with simple predictors, which output singleton predictions. The accuracy of forced prediction is called *forced accuracy*.

Finally, each prediction can be complemented with two indicators:

- *confidence*: $\sup\{1 - \epsilon : |\Gamma^{\epsilon}| \leq 1\}$
- *credibility*: $\inf\{\epsilon : |\Gamma^{\epsilon}| = 0\}$

In the case of classification, credibility is equal to the maximum value of all possible p-values, and confidence equals 1 less the second maximum p-value. When the number of classes is two, credibility is a maximum of two p-values, confidence equals 1 less the other p-value.

Confidence and credibility can be very informative when forced predictions are made. The confidence shows how confident we are in rejecting the other labels, and high confidence means that the alternative hypothesis is excluded by having a low p-value. The credibility demonstrates how well the chosen label conforms with the rest of the set, so high credibility checks whether the prediction itself does not have too small a p-value.

Thus, these two characteristics reflect how reliable predictions are. The forced prediction is considered to be reliable if its confidence is close to 1 and credibility is not close to 0 (because if a label does not match an object, the p-value must be close to 0). An interesting case of low credibility indicates that a new object itself is not representative of the training set.

P-values in Statistics and Confidence Machines

The definition of p-values introduced in this section differs from the classical p-value definition in statistics. These two types of p-values are different notions, but they bear the same name because of similar properties. For confidence machines, the probability of the event that the p-value does not exceed $0 < \gamma \leq 1$ is not greater than γ for any i.i.d. probability distribution on $\mathbf{Z}^\infty$. Moreover, for smoothed confidence machines, which are the modification of confidence machines and are described in Section 5.1, similar property coincides with the property of statistical p-values:

$$\mathcal{P}(\text{p-value} \leq \gamma) = \gamma$$

for any $0 < \gamma \leq 1$ and any i.i.d. probability distribution $\mathcal{P}$ on $\mathbf{Z}^\infty$.

In order to avoid confusion, it should be noted that in this thesis we are not working in a classical statistical context: there is no estimation of the risk — the probability that the classifier errs — on the whole population of objects. On the contrary, we calculate p-values for each object and each hypothetical label, aim at rejecting the hypothesis that the resulting sequence is i.i.d. and estimate our confidence in individual prediction.

Throughout the thesis we always use p-values as defined for confidence machines, not statistical p-values. The only exception is Appendix B, where we carry out statistical analysis of the UKCTOCS ovarian cancer data set and calculate statistical p-values by the use of the Monte-Carlo method in order to estimate statistical significance of classification results we obtain.

2.1.2.2 Strangeness Measure Examples

There are different ways to define the strangeness measure, the core element of any confidence machine. Almost any machine learning algorithm can be used to construct it. There are known implementations based on such algorithms as SVMs [27; 52], k-nearest neighbours [49], nearest centroid [6], linear discriminant [60], naive Bayes [60], kernel perceptron [37]. The most successful and the most widely used ones have been strangeness measures derived from k-nearest neighbour and SVM algorithms. Confidence machines based on these strangeness measures will be referred to as **CM-kNN** (where k is a number of nearest neighbours) and **CM-SVM**, respectively.

A k-nearest-neighbour strangeness measure proved to produce confidence machines highly efficient on many data sets in spite of its primitivity [49; 65]. It is applicable in the case of classification. We are given a bag of examples $\Lbag(x_1, y_1), \ldots, (x_n, y_n)\Rbag$ and need to define a strangeness score of example (x_i, y_i):

$$\alpha_i = A_n(\Lbag(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)\Rbag, (x_i, y_i))\,.$$

We assume that the objects are vectors in a Euclidian space. We then define the strangeness measure using the idea of the k-nearest neighbour algorithm.

We calculate distances from the object x_i to all other objects in a bag $d(x_j, x_i)$, $j = 1, \ldots, i-1, i+1, \ldots, n$ and find the k objects that are the closest to x_i among those who have the same label y_i as x_i. We denote these selected k examples by (x_{i_s}, y_{i_s}), $s = 1, \ldots, k$. Similarly, we find the k objects that are the closest to x_i among the ones with labels other than y_i; they will be denoted by (x_{j_s}, y_{j_s}), $s = 1, \ldots, k$. Finally, we define the strangeness measure as

$$\begin{aligned} &A_n(\Lbag(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)\Rbag, (x_i, y_i)) \\ &\qquad := \frac{\sum_{s=1}^{k} d(x_i, x_{i_s})}{\sum_{s=1}^{k} d(x_i, x_{j_s})}\,. \end{aligned} \tag{2.4}$$

This implies that an object is considered to be nonconforming if it is far from objects with the same label and close to objects labelled in a different way.

Another strangeness measure considered in this thesis is based on the SVM algorithm, which was proposed in [61]. This strangeness measure was originally designed and used in [25; 27; 52; 65] for the problem of binary classification when possible labels are $\mathbf{Y} = \{-1, 1\}$.

We assume that objects in the bag $\Lbag(x_1, y_1), \ldots, (x_n, y_n)\Rbag$ are vectors in a dot product space H and consider the quadratic optimisation problem

$$\frac{1}{2}(\omega \cdot \omega) + C\left(\sum_{i=1}^{n} \xi_i\right) \to \min,$$

where $C > 0$ is fixed and the variables $w \in S$, $\xi = (\xi_1, \dots, \xi_n)' \in \mathbb{R}^n, b \in \mathbb{R}$ are subject to the constraints

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, i = 1, \dots, n,$$

$$\xi_i \geq 0, i = 1, \dots, n.$$

If this optimisation problem has a solution, it is unique. We will denote it the same way: w, $\xi = (\xi_1, \dots, \xi_n)'$, b. The hyperplane $w \cdot x + b = 0$ is called the optimal separating hyperplane. It determines predictions for new objects: if $w \cdot x + b > 0$, then we output 1 as prediction, -1 otherwise.

If we apply a transformation $F : \mathbf{X} \to H$ mapping objects into the feature vectors $F(x_i) \in H$, where H is a dot product space, this will replace x_i by $F(x_i)$ in the optimisation problem above. Then one can apply the Lagrange method assigning a Lagrange multiplier α_i to each inequality above. If we define $K(x_i, x_j) = F(x_i) \cdot F(x_j)$, the modified problem (also called the dual problem) is the following:

$$\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \to \max,$$

$$\sum_{i=1}^{n} y_i \alpha_i = 0, 0 \leq \alpha_i \leq C, i = 1, \dots, n.$$

Lagrange multipliers α_i found as solutions of this problem can be interpreted the following way: $\alpha_i > 0$ only for support vectors, which are boundary examples, define the hyperplane and are therefore considered as the least conformal training examples; $\alpha_i = 0$ for examples which conform well with the SVM model. Hence the solutions of the dual problem α_i can be used as strangeness scores.

The SVM strangeness measure introduced above is applicable only to binary classification problems. However, we can also use it when addressing multilabel classification (i.e., when $|Y| > 2$). In such cases, we will apply the *one-against-one* procedure: when calculating strangeness scores, we will consider several auxiliary binary classification problems instead of one multilabel classification. In these auxiliary problems, we will discriminate between every two available classes.

If A is an SVM strangeness measure, the strangeness measure A' for multilabel classification is calculated as

$$A'(\wr(x_1, y_1), \ldots, (x_l, y_l)\wr, (x, y)) := \max_{y' \neq y}(A(B_{y,y'}, (x, 1))),$$

where $B_{y,y'}$ is the bag obtained from the original bag $\wr(x_1, y_1), \ldots, (x_l, y_l)\wr$ the following way: we remove all examples (x_i, y_i) with $y_i \notin \{y, y'\}$, replace each (x_i, y) with $(x_i, 1)$ and replace each (x_i, y') with $(x_i, -1)$. In words, each strangeness score is the maximum one out of all strangeness scores obtained in auxiliary binary classification problems.

Thus, when computing one strangeness score, we consider $|Y| - 1$ auxiliary binary classification problems. When applying a conformal predictor, we have to compute strangeness scores for all examples and for all hypotheses $y \in \mathbf{Y}$, and $3|Y|(|Y| - 1)/2$ auxiliary binary classification problems are required.

2.1.3 Category-Based Confidence Machines

Confidence machines allow us to obtain a guaranteed error rate which does not exceed the predetermined value. However, we may encounter certain applications, when we know that certain objects are easier to correctly classify than others. For example, in medical diagnosis men may be more easily diagnosed than women, or it is more likely to misclassify a healthy patient than a diseased one. In this case, confidence machines will guarantee the overall error rate; however, they may result in the higher actual error rate on harder groups of objects and the lower one on easier groups of objects. We will therefore not be able to guarantee the error rate within these groups.

Category-based confidence machines, also known as Mondrian conformal predictors in [65; 66], represent the extension of confidence machines and allow us to tackle this problem. They split all possible examples into categories (such as, healthy and diseased patients, or categories according to their sex, age etc) and set significance levels ϵ_k, one for each category k. As a result, category-based confidence machines can guarantee that asymptotically the predictions for objects of each type k are erroneous with frequency at most ϵ_k.

Thus, category-based confidence machines allow us to solve two main problems:

- We can guarantee not only an overall accuracy, but also a certain level of accuracy within each category of examples. In particular, in medical diagnosis we can preset required accuracy rates among healthy and diseased samples. We will call these rates *regional specificity* and *regional sensitivity*, respectively. This will allow avoiding classifications when low regional specificity is compensated by high regional sensitivity or the other way around.

- If we preset different significance levels for different categories, we can treat them in a different way: e.g., in medical diagnosis we could put regional sensitivity first and consider a misclassification of a diseased sample more serious that misclassification of a healthy sample.

The difference in constructing category-based confidence machines is that we compare strangeness of (x_n, y) not with all examples in the sequence but only with the category that can correspond to certain types of labels, objects and (or) the ordinal number of the example. This approach will allow us to achieve validity within categories (or *conditional validity*): the asymptotic error rate within these categories will not exceed the significance level determined beforehand.

2.1.3.1 Definitions

Let us again assume that we are given a training set of examples $(x_1, y_1), \ldots, (x_{n-1}, y_{n-1})$ and our goal is to predict the classification y_n for a new object x_n.

Division into categories is determined by a *Mondrian taxonomy*, or simply *taxonomy*. It is a measurable function $\kappa : \mathbb{N} \times \mathbf{Z} \to K$, where K is the measurable space (at most countable with the discrete σ-algebra) of elements called *categories*, with the following property: the elements $\kappa^{-1}(k)$ of each category $k \in K$ form a rectangle $A \times B$, for some $A \subseteq \mathbb{N}$ and $B \subseteq \mathbf{Z}$. In words, a taxonomy defines a division of the Cartesian product $\mathbb{N} \times \mathbf{Z}$ into categories.

A category-based strangeness measure related to a taxonomy κ is a family of measurable functions $\{A_n : n \in \mathbb{N}\}$ of the type

$$A_n : K^{n-1} \times (\mathbf{Z}^{(*)})^K \times K \times \mathbf{Z} \to \bar{\mathbb{R}},$$

where $(\mathbf{Z}^{(*)})^K$ is a set of all functions mapping K to the set of all bags of elements of $\mathbf{Z}$. This strangeness measure will again assign a strangeness score α_i to every example in the sequence $z_i := (x_i, y_i)$, $i = 1, \dots, n$ including a new example and will evaluate 'nonconformity' between a set and its element:

$$\alpha_i := A_n(\kappa_1, \dots, \kappa_{n-1}, \\ (k \mapsto \Lbag z_j : j \in \{1, \dots, i-1, i+1, \dots, n\} \;\&\; \kappa_j = k \Rbag), \kappa_n, z_i),$$

where $\kappa_i := \kappa(i, z_i)$ for $i = 1, \dots, n$ such that $\kappa_i = \kappa_n$.

When calculating a p-value, we will compare α_n not to all other α_is but only to those within the category of the new example, that is, the p-value associated with the possible label y for x_n is defined as

$$p_n(y) = \frac{|\{i = 1, \dots, n : \kappa_i = \kappa_n \;\&\; \alpha_i \geq \alpha_n\}|}{|\{i = 1, \dots, n : \kappa_i = \kappa_n\}|}.$$

Finally, the *category-based confidence machine* determined by the category-based strangeness measure A_n and a set of significance levels ϵ_k, $k \in K$ is defined as a measurable function $\Gamma : \mathbf{Z}^* \times \mathbf{X} \times (0,1)^K \to 2^{\mathbf{Y}}$ such that the prediction set $\Gamma^{(\epsilon_k : k \in K)}(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n)$ is defined as the set of all labels $y \in \mathbf{Y}$ such that $p_n > \epsilon_{\kappa(n,(x_n,y))}$. Thus, for any finite sequence of examples with labels $(x_1, y_1, \dots, x_{n-1}, y_{n-1})$, a new object without a label x_n and a set of significance levels ϵ_k, $k \in K$ for each category, the category-based confidence machine outputs a region prediction $\Gamma^{(\epsilon_k : k \in K)}$ — a set of possible labels for a new object.

The category-based confidence machine defined above is *conditionally conservatively valid*: asymptotically, the frequency of errors made by category-based confidence machine (that is, cases when prediction set Γ^{ϵ_k} does not contain a real label) on examples in category k does not exceed ϵ_k for each k.

Strictly speaking, for any exchangeable probability distribution P on $\mathbf{Z}^\infty$, any category $k \in K$ and any significance level ϵ_k,

$$\limsup_{n\to\infty} \frac{\sum_{1\le i\le n, \kappa(i,(x_i,y_i))=k} err_n^{\epsilon_k}(\Gamma)}{|\{i : 1 \le i \le n, \kappa(i,(x_i,y_i)) = k\}|} \le \epsilon_k$$

with probability one, where $err_n^{\epsilon_k}(\Gamma)$ is equal to 1 when the prediction set Γ^{ϵ_k} does not contain a real label y_n and 0 otherwise. Thus, we guarantee the asymptotical error rate not only within all examples but also within categories. Similarly to validity, the property of conditional validity is proved only for the online mode, but it is empirically shown to remain in the offline mode [65]. When referring to conditional validity of category-based confidence machines throughout this thesis, we will always imply the property of conditional conservative validity.

Category-based confidence machines can be forced to make singleton predictions the same way as confidence machines: they can output labels with the highest p-values. In this case, we can similarly compute forced predictions, their confidence, credibility and overall forced accuracy. Examples of the output of category-based confidence machines are given in Table 4.1, which provides true labels ('True diagnosis'), forced predictions ('Predicted diagnosis'), p-values for two possible labels (0 and 1), confidence and credibility. The detailed explanation is also provided in Section 4.4.1.1.1.

2.1.3.2 Taxonomy Examples

Category-based confidence machines are defined by two elements: a strangeness measure and a taxonomy. Any strangeness measure embedded in confidence machines could be used when defining a category-based strangeness measure.

Important types of category-based confidence machines according to the type of their taxonomies are the following.

- **Confidence machines.** A category-based confidence machine with one single taxonomy $\kappa(n,(x_n,y_n)) = 1$ turns into a confidence machine. Hence confidence machines represent a type of category-based confidence machines, not the other way around.

- **Label-conditional confidence machines.** The category of an example is determined by its label $\kappa(n,(x_n,y_n)) = y_n$, i.e., the taxonomy consists of several categories each of which corresponds to a single label. Hence p-values are calculated as follows:

  $$p_n(y) = \frac{|\{i=1,\dots,n-1 : y_i = y \ \& \ \alpha_i \geq \alpha_n\}| + 1}{|\{i=1,\dots,n : y_i = y\}|}\,, \tag{2.5}$$

  For example, in medical diagnosis we can consider categories of healthy and diseased patients. This taxonomy will allow us to guarantee the accuracy within these classes: regional specificity and regional sensitivity.

- **Attribute-conditional confidence machines.** The category of an example is determined by its attributes: $\kappa(n,(x_n,y_n)) = f(x_n)$. For instance, we can consider categories which correspond to old/young patients, men/women or different combinations of these features.

- **Inductive confidence machines.** The category of an example is determined only by its ordinal number in the sequence. We fix the ascending sequence of positive integers $0 < m_1 < m_2 < \dots$, which are the borders of different categories, and consider examples with ordinal numbers $\{1,\dots,m_1\}$, $\{m_1+1,\dots,m_2\}$, $\{m_2+1,\dots,m_3\}$ etc as examples of categories 1, 2, 3 etc, respectively.

  The p-values are then defined in the following way. If $n \leq m_1$,

  $$p_n(y) := \frac{|i=1,\dots,n : \alpha_i \geq \alpha_n|}{n},$$

  where

  $$\alpha_i := A_n(\wr(x_1,y_1),\dots,(x_{i-1},y_{i-1}),(x_{i+1},y_{i+1}),\dots,(x_{n-1},y_{n-1}),$$
  $$(x_n,y)\int,(x_i,y_i)),\, i = 1,\dots,n-1\,,$$

  $$\alpha_n := A_n(\wr(x_1,y_1),\dots,(x_{n-1},y_{n-1})\int,(x_n,y)).$$

Otherwise, we find the k such that $m_k < n \leq m_{k+1}$ (e.i., find the category of the sample) and set

$$p_n(y) := \frac{|\{i = m_k + 1, \ldots, n : \alpha_i \geq \alpha_n\}|}{n - m_k},$$

where the strangeness scores α_i are defined by

$$\alpha_i := A_{m_k+1}(\Lbag (x_1, y_1), \ldots, (x_{m_k}, y_{m_k}) \Rbag, (x_i, y_i)), i = m_k + 1, \ldots, n - 1,$$

$$\alpha_n := A_{m_k+1}(\Lbag (x_1, y_1), \ldots, (x_{m_k}, y_{m_k}) \Rbag, (x_n, y)).$$

2.1.4 Venn Machines

Machine learning applications may require prediction of a label complemented with the probability that this prediction is correct. For example, in medical diagnosis, one may need to predict the probability of a disease (disease risk) rather than make a diagnosis. Different machine learning methods can output probabilistic predictions, i.e., a probability distribution of the unknown label y for a new object x_n. We will call this type of methods *probability predictors*. However, most of probability predictors are based on strong statistical assumption which do not hold true for real-world data. Therefore, when the assumed statistical model is incorrect, the algorithm may output invalid prediction (Detailed description of limitations of probabilistic methods, including Bayesian approach, is given in Section 2.2.4.). The framework of Venn machines, which were introduced in [65; 67], also allows us to produce probability distributions, but their predictions are valid under a simple i.i.d. assumption.

Venn machines output multiprobability predictions — a set of probability distributions of a label. This output can be also interpreted in a different way: as a prediction with the assigned interval of probability that this prediction is correct. Venn machine outputs are always valid (precise definitions will be given later). The property of validity is based only on the i.i.d. assumption, that the data items are generated independently from the same probability distribution. This assumption is much weaker than any probabilistic assumption, which allows Venn machines to produce valid predictions without knowing a real distribution of examples.

Venn machines represent a framework that can generate a range of different algorithms. Similarly to confidence machines, practically any known machine learning algorithm can be used as an *underlying algorithm* in this framework and thus result in a new Venn machine. However, regardless of the underlying algorithm, Venn machines output valid results.

In brief, Venn machine functionality can be described as follows. First, we are given a division of all examples into categories. Then since we do not know the true labels of the new object, we try every possible label as a candidate for its label. For each hypothesis about the possible label, we classify the new object into one of the categories and then use empirical probabilities of labels in the chosen category, that is, frequencies of true labels, as predictable distribution of the new object's label. As a result, the category assigned to an example depends not only on the example itself but also on its relation to the rest of the data set. Thus, the Venn machine outputs several probability distribution rather that one, one for each hypothesis about the new label.

2.1.4.1 Definitions

Venn machines can be applied only to the problem of classification ($|\mathbf{Y}| \in \mathbb{N}$). Let us consider a training set consisting of object, x_i, label, y_i, pairs: (x_1, y_1),…,(x_{n-1}, y_{n-1}). To predict a label y_n for a new object x_n, we check different hypotheses

$$y_n = y\,, \tag{2.6}$$

each time including the pair (x_n, y_n) into the set.

The idea of Venn machines is based on a *taxonomy function* $A_n : \mathbf{Z}^{(n-1)} \times \mathbf{Z} \to T$, $n \in \mathbb{N}$, which classifies the relation between an example and the bag of the other examples:

$$\tau_i = A_n\left((x_i, y_i), \wr (x_1, y_1), \dots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \dots, (x_n, y_n) \int\right)\,. \tag{2.7}$$

Values τ_i are called categories and are taken from a finite set $T = \{\tau_1, \tau_2, \dots, \tau_k\}$. Equivalently, a taxonomy function assigns to each example (x_i, y_i) its category τ_i, or, in other words, groups all examples to a finite set of categories.

This grouping should not depend on the order of examples within a sequence.

As one can see, Venn taxonomies are different from Mondrian taxonomies used in category-based confidence machines. The category assigned in a Mondrian taxonomy does not depend on other examples in the training set but may be dependent on the ordinal number of the example in the sequence. In contrast, categories of Venn taxonomies are determined by the rest of the training set but cannot be dependent on their order in the sequence.

The conventional way of using Venn ideas was as follows. Categories are formed using only the training set. For each non-empty category τ, the following values are calculated: N_τ is the total number of examples from the training set assigned to category τ, and $N_\tau(y')$ is the number of examples within category τ that are labelled with y'. Then empirical probabilities of an object within category τ to have a label y are found as

$$P_\tau(y') = \frac{N_\tau(y')}{N_\tau}\,. \tag{2.8}$$

Now, given a new object x_n with the unknown label y_n, one should assign it somehow to the most likely category of those already found using only the training set; let τ^* denote it. Then the empirical probabilities $P_{\tau^*}(y')$ are considered as probabilities of the object x_n to have a label y'. The idea of confidence machines allows us to construct several probability distributions of a label y' for a new object. First we consider a hypothesis that the label y_n of a new object x_n is equal to y ($y_n = y$). Then we add the pair (x_n, y) to the training set and apply the taxonomy function A to this extended sequence $(x_1, y_1), \ldots, (x_{n-1}, y_{n-1}), (x_n, y)$. This groups all the elements of the sequence to categories. Let $\tau^*(x_n, y)$ be the category containing the pair (x_n, y). Now for this category we calculate, as previously, the values N_{τ^*}, $N_{\tau^*}(y')$ and empirical probability distribution

$$P_{\tau^*(x_n,y)}(y') = \frac{N_{\tau^*}(y')}{N_{\tau^*}},\ y' \in \mathbf{Y}\,. \tag{2.9}$$

This distribution depends implicitly on the object x_n and its hypothetical label y. Trying all possible hypotheses of the label y_n being equal to y, we obtain a set of distributions $P_y(y') = P_{\tau^*(x_n,y)}(y')$ for all possible labels y. These distributions in general will be different as when changing the value of y, we, in general, change grouping into categories, the category $\tau^*(x_n, y)$, containing the pair (x_n, y), the numbers N_{τ^*} and $N_{\tau^*}(y')$. Thus, as the output of Venn predictors, we obtain as many probability distributions as the number of possible labels.

Venn machines are valid in the sense of agreeing with the observed frequencies (for details, see [65]). Among the first writers on frequentist probabilities we could name John Venn ([62]) and Richard von Mises ([41], [42]). The validity of Venn machines is based on special testing by supermartingales and is a generalisation of the notion of valid probabilistic prediction. A formal definition of validity is beyond the scope of the thesis and can be found in [65]. We will just state a corresponding theorem here:

Theorem 2.1 (Vovk, Gammerman and Shafer, 2005) *Every Venn predictor is an N-valid multi-probability predictor.* □

In this thesis we do not consider theoretical properties of Venn machines but run an empirical study of different implementations of this framework.

The original output of Venn machines is complex: it consists of several label probability distributions. However, this output can be interpreted in a simpler way. We can force Venn machines to make singleton predictions so that each prediction is complemented with an interval that the prediction is correct. Similarly to confidence machines, we will call this type of singleton predictions *forced predictions* and corresponding accuracy — *forced accuracy.*

Forced predictions are made as follows. After calculating empirical probability distributions $P_y(y')$, $y, y' \in \mathbb{Y}$ we compute the quality of each prediction y': $q(y') = \min_{y \in \mathbf{Y}} P_y(y')$ and then predict the label with the highest quality $y_{\text{pred}} = \arg\max_{y' \in \mathbf{Y}} q(y')$. We complement this singleton prediction with a *probability interval*

$$[\min_{y \in \mathbf{Y}} P_y(y_{\text{pred}}), \max_{y \in \mathbf{Y}} P_y(y_{\text{pred}})] \tag{2.10}$$

as the interval for the probability that this prediction is correct. If this interval is denoted by $[a, b]$, the complementary interval $[1 - b, 1 - a]$ is called the *error* *probability interval*, and its ends $1-b$ and $1-a$ are referred to as *lower error probability* and *upper error probability*, respectively.

In a binary classification problem (when $\mathbf{Y} = \{0,1\}$), Venn predictor output can be translated in the following way. It comprises only two probability distributions, both of which can be represented by $P_y(1)$ — the probability of the event $y_n = 1$. Thus, the output of Venn predictor can be interpreted as the interval

$$[P^-_{\text{new}}, P^+_{\text{new}}] = [\min\{P_0(1), P_1(1)\}, \max\{P_0(1), P_1(1)\}] \,, \tag{2.11}$$

which is an estimation of probability that $y_n = 1$. We will refer to P^-_{new} and P^+_{new} as *lower Venn prediction* and *upper Venn prediction*, respectively.

The examples of Venn machine output for a binary classification problem are provided in Table C.5. This table contains true labels, lower Venn predictions P^-_{new} and upper Venn predictions P^+_{new}. Interpretation of Venn predictions is also given in Section 4.4.2.1.

2.1.4.2 Venn Taxonomy Example

A Venn machine is entirely defined by its Venn taxonomy, which can be constructed by the use of practically any machine learning algorithm. Here is an example of a taxonomy based on a 1-nearest neighbour algorithm. We will denote it by **VM-1NN** and will use throughout the thesis.

We assume that all examples are vectors in a Euclidean space and set the category of an example equal to the label of its nearest neighbour

$$A_n\left((x_i,y_i), \wr (x_1,y_1),\ldots,(x_{i-1},y_{i-1}),(x_{i+1},y_{i+1}),\ldots,(x_n,y_n) \int\right) = y_j \ \ ,$$

where

$$j = \arg \min_{j=1,\ldots,i-1,i+1,\ldots,n} ||x_i - x_j|| \ \ .$$

This Venn machine was proposed in [65] and proved to output accurate predictions with narrow prediction intervals.

2.2 Comparison with Other Approaches

Confidence machines, category-based confidence machines and Venn machines represent one type of algorithms which produce predictions complemented with the information on their reliability. In this section we compare them with other approaches.

Firstly, we compare algorithms with online validity with two big classes of algorithms: simple predictors (that output a label but do not provide any additional information) and probability predictors (that output a probability distribution of a new label).

Secondly, we will briefly describe other methods that provide information on how reliable predictions are, compare them with confidence and Venn machines and demonstrate their limitations. These methods include confidence intervals, statistical learning theory (PAC theory) and probabilistic approaches.

2.2.1 Comparison with Simple Predictors and Probability Predictors

To begin with, we classify different types of algorithms considered so far in Table 2.1 according to their output: first, according to the output element (a label or a label probability distribution) and, second, according to a number of such elements in the output (one or several). This table demonstrates how algorithms with online validity relate to other machine learning algorithms: simple predictors and probability predictors.

Table 2.1: Classification of algorithms according to their output

Output	. . . label(s)	. . . probability distribution(s)
One . . .	Simple predictor (e.g., SVM)	Probability predictor (e.g., logistic regression)
A set of . . .	Confidence machine, category-based confidence machine	Venn machine

In contrast to simple predictors, confidence and Venn machines hedge predictions, i.e., express how much a user can rely on them. In the introduction of this thesis we described two measures of performance of confidence and Venn machines: validity and efficiency. Validity demonstrates how correct predictions are; efficiency is concerned with how informative they are.

For confidence machines, validity implies that the number of errors is close to the preset significance level, and efficiency means outputting as few as possible multiple predictions.

For Venn machines, validity results in output probability distributions agreeing with observed frequencies. A probability interval output by Venn machine is efficient if it is narrow and close enough to 1.

Table 2.2: Comparison of confidence and Venn machines with simple and probability predictors

Predictor type	Simple predictors	Confidence machines	Probability predictors	Venn machines
Output	Singleton prediction	Set of predictions	Probability distribution	Multiprobability prediction
Validity	Depends on the algorithm	Guaranteed	Guaranteed under strong statistical assumption	Guaranteed
Efficiency	Guaranteed	Depends on the strangeness measure	N/a	Depends on the Venn taxonomy

When considering simple predictions, we can also use notions of validity and efficiency. In this respect, simple predictors and confidence machine demonstrate opposite approaches to learning. This is summarised in Table 2.2. In simple predictions, the efficiency is guaranteed since each prediction is singleton, but validity is not. In confidence machines, the validity of predictions is guaranteed, but efficiency depends on the strangeness measure. In addition, confidence machines allow us to balance efficiency and the error rate:

lower preset error rates produce larger region predictions, and vice versa. This feature makes confidence machines a flexible tool.

The applicability of different algorithms depends on the learning goal of the application, that is, what we put first: validity or efficiency. In the case of confidence machines, we sacrifice efficiency of predictions to control the risk of error. This is suitable for application where low risk of error is required, for example, in medical diagnosis. In simple predictions, we put validity first instead.

Probability predictors usually produce valid predictions under statistical assumptions that are stronger than i.i.d., whereas Venn machines produce predictions with guaranteed validity subject to the i.i.d. assumption. One should also keep in mind that the property of Venn machine validity is expressed differently from the same property of confidence machines due to the difference in the output.

2.2.2 Comparison with Confidence Intervals

The term of the *confidence level*, which we use when defining confidence machines and category-based confidence machines, is also widely deployed in statistics when constructing confidence intervals. Here we emphasize that this term has different meanings when applied in the context of confidence machines and confidence intervals. Hence values of such confidence levels should be directly compared when used as different notions.

Confidence intervals are deployed in statistics as an interval estimate of a population parameter: we produce confidence intervals (instead of a single value) that estimate the parameter, and a confidence level indicates how likely the value of the parameter lies within the confidence interval. The calculation of a confidence interval generally requires assumptions about the statistical model (usually it is a parametric population). For example, it may be based on the assumption that the distribution of the sample population is normal.

Confidence intervals can be also applied in machine learning. In this case, we fix a simple predictor and consider an error rate of a predictor (i.e., an expectation of the loss of the simple predictor) as a parameter to estimate. We can then produce confidence intervals which estimate predictor's error relying on errors at different steps. This approach is called *hold-out estimates* [65, Section 10.1].

- Thus, the machine learning algorithm is fixed, and confidence intervals provide an estimate of a true error rate of this algorithm with a confidence level as an indicator of interval's reliability. The upper bound of this confidence interval could be a useful estimate of the algorithm error rate. In confidence machines, we generate a new algorithm with the accuracy rate which is asymptotically guaranteed under simple statistical assumptions and is equal to the preset confidence level. Hence, confidence levels in confidence machines and confidence levels in confidence intervals can be both used in machine learning but for different purposes. Their interpretations are therefore not similar, and their values should not be compared.

- Confidence intervals output an estimation of the accuracy of the algorithm in general and do not supply any information regarding how reliable a prediction for each individual object is. However, algorithms with online validity — confidence and Venn machines — provide the level of uncertainty for each individual prediction in the form of confidence and credibility in the case of (category-based) confidence machines or probability interval in the case of Venn machines.

- Confidence intervals and algorithms with online validity use different statistical assumption. Confidence and Venn machines are based on the i.i.d. assumption or a weaker exchangeability assumption, which can be usually satisfied by randomly permuting the data. Meanwhile, confidence intervals when used in hold-out estimates are based on the assumptions that the errors (rather than examples) are i.i.d. However, this assumption does not take the nature of learning process into consideration, because in real learning, algorithms take into account the information on previously made errors. This approach has been therefore criticised ([13], Section 4.1).

2.2.3 Comparison with Statistical Learning Theory

Statistical learning theory began as Vapnik-Chervonenkis theory in the 1960s and was partially rediscovered by Valiant (see [61] for the review).

In brief, in statistical learning theory, a simple label prediction is produced for each object based on the training set, and there is a theoretical guarantee that these predictions become more accurate with greater probability while the training set of objects increases in size: they are probably approximately correct. This implies that the probability of error will not exceed a certain threshold ϵ unless an event of the preset probability δ has happened. We set the value of probability δ and calculate ϵ, which is an expected loss of the total population and is called *true risk*.

- The main problem of applying statistical learning theory is that the derived error bounds ϵ are too loose to be informative: these bounds often have large values (there may be some exceptions) and sometimes are greater than one. The only problems that can benefit from these error bounds are the "easy ones" with the large number of objects.

  Loose error bounds have been obtained even for the relatively clean United States Postal Service data described in [61, Section 12.2]. Loose error bounds were also confirmed in [20; 34; 46]. Thus, experiments demonstrate that PAC bounds are not a good practical tool for estimating the true accuracy of a learning algorithm. By contrast, confidence machines are valid and provide tight and useful bounds.

- Error bounds provided by statistical learning theory are not preset but are computed; while confidence machines allow users to control error bounds by predefined confidence levels.

- Statistical learning theory estimates the approximate error rate for the algorithm in general without giving a measure of uncertainty for an individual object; but both confidence machines and Venn machines complement each individual prediction with the information which indicates how reliable each prediction is, and this information depends on the object.

2.2.4 Comparison with Probabilistic Algorithms

There is a range of probabilistic algorithms, which are based on posterior probabilities and use strong statistical assumptions on the data distribution. Bayesian classifiers (e.g., a naive Bayes algorithm and Bayesian ridge regression [65, Section 10.3]) and Platt's calibration [48] are among them.

- The main drawback of such algorithms is that they assume a model of data distribution. We can be sure in the statistical model only for artificially generated data. However, for real-world data model assumption may not hold true and be misleading. For example, Bayesian ridge regression allows us to output prediction intervals in regression problems. When the chosen probability distribution conforms with the statistical model, output intervals are valid. Otherwise, the produced intervals may make more errors than expected. This was in explored in [65, Section 10.3].

  Meanwhile, confidence machines and Venn machines rely only on data exchangeability, which can be usually satisfied when data examples are randomly permuted.

  If a Venn taxonomy depends on a statistical model, this underlying model is not required to be correct for the corresponding Venn machine to be valid. The worst thing which may happen if the underlying model is incorrect is that the Venn machine will output uninformative predictions (wide probability intervals not close enough to 0 or 1) as opposed to misleading prediction intervals that can be produced by probabilistic algorithms. Empirical comparison of Venn machines with a probabilistic algorithm is carried out in Section 4.4.2 of this thesis. It demonstrates that a probability prediction may be misleading while Venn machine outputs a narrow probability interval which almost always covers the empirical label probability.

- When applied to classification problems, most probabilistic algorithms output a singleton prediction with assigned probability, which is a label with the highest posterior probability. This implies that the probability of the output is not controlled whereas when applying confidence machines we can control the total error asymptotically. However, the procedure can be changed so that we output the set of labels with highest posterior probabilities with the sum greater than the preset confidence level. Such approach will allow us to output region prediction with the preset posterior probability.

In general, posterior probabilities and p-values represent different notions; hence outputs of confidence machines and probabilistic algorithms are interpreted in different ways (for example, such probabilistic predictions cannot guarantee the asymptotical error rate in the online mode) and therefore should not be directly compared.

- Some Bayesian methods assume the independence between features (for example, a naive Bayes classifier). But this is a further restriction on the model, and it does not usually hold true for real-world data. Another option is to accept that different features are not independent. Then a model of dependencies, called a Bayesian belief networks, has to be learned or imposed. However, in this case the solution of the problem gets computationally inefficient (see [43], Section 6.11).

  At the same time, confidence machines and Venn machines produce valid predictions without these restrictions.

2.3 Summary

In this chapter we introduced the frameworks of confidence machines, category-based confidence machines and Venn machines. These algorithms were recently introduce in [65] and represent a new generation of algorithms with online validity. They assign information on how reliable the prediction is to each individual example, and this information is valid when the online mode is applied.

We demonstrated that these algorithms have distinct advantages over other methods that hedge predictions or estimate overall performance. These advantages could be summarised as follows:

1. Algorithms with online validity can express our confidence about each individual prediction rather than all predictions in total, and this confidence in a new prediction is tailored to the new object.

2. Property of validity is based on a simple i.i.d. assumption (or a weaker exchangeability assumption), which can be often satisfied when data sets are randomly permuted. Valid predictions do not depend on the assumed probability distribution of examples.

3. In the case of confidence machines and category-based confidence machines, the error rate can be controlled rather than computed, which makes these methods a flexible tool.

4. Methods with online validity are not single algorithms, but a flexible framework: each of them depends on a core element (a strangeness measure for confidence machines; a strangeness measure and a taxonomy for category-based confidence machines; a Venn taxonomy for Venn machines), and practically any machine learning algorithm can be used to define this core element. Thus, the framework can give rise to a set of different algorithms which can perform well on different types of data.

Chapter 3

Design of Algorithms with Online Validity

Algorithms with online validity represent a flexible framework and can use practically any known machine learning method as an underlying algorithm for designing a strangeness measure (for confidence machines and category-based confidence machines) or a Venn taxonomy (for Venn machines).

Since these algorithms have theoretically guaranteed property of validity, their performance can be measured by their efficiency. Good performance in terms of accuracy of the underlying algorithm is usually translated into good efficiency of the corresponding confidence or Venn machine. Therefore, efficiency of algorithms with online validity varies across the range of underlying algorithms and depends on the data analysed. For this reason, we are looking for new strangeness measures and Venn taxonomies that could result in efficient predictions on certain types of data.

In this chapter we propose new implementations of algorithms with online validity: confidence machines constructed by the use of random forests, Venn machines based on random forests and Venn machines with a taxonomy derived from SVMs. We expect the designed confidence and Venn machines to inherit advantages of their underlying algorithms — random forests and SVMs — and to maintain the property of validity.

3.1 Designed Algorithms

3.1.1 Confidence Machines Based on Random Forests

We propose several strangeness measures based on a random forest classifier. These strangeness measures can be used in confidence machines or category-based confidence machines.

We can speculate that confidence machines based on a random forest will not suffer drop in forced accuracy in comparison with random forest accuracy. As a result, confidence machines are likely to provide high forced point prediction accuracy and good efficiency of region predictions.

We also expect designed confidence machines to inherit advantages of random forests, in particular to perform well on noisy data.

3.1.1.1 Random Forests

The random forest is a classifier which proved to have certain advantages over different machine learning methods. In this work we consider the type of random forests described in [7].

The initial motivation to apply random forests in our research was the fact that they outperformed other machine learning methods in the Leiden Clinical Mass Spectrometry Proteomic Diagnosis Competition [3; 29], whose data we used in our analysis along with other data sets. Theoretical results [7] demonstrate that the random forest do not overfit when more trees are added. It also empirically proved to have the following advantages [7; 8]:

1. It produces high accuracy for many data sets.

2. It can process data with a large number of features where each feature is weak, that is, carries a small amount of information.

3. It is relatively robust to mixed variable types, missing data, outliers and noisy data.

4. Constructing random forests is relatively fast (faster than bagging and boosting).

In brief, a random forest is a classifier that consists of decision trees, each of which provides a vote for a certain class. An important problem, especially in medical and veterinary diagnosis, is that we often have to process data with a large number of features, each of which contains small amount of information. For this reason, a single decision tree classifier may overfit the data and not result in high accuracy. Combining a large number of trees in a random forest can lead to more reliable predictions.

When growing decision trees, random forests apply bagging and random feature selection. We construct a large number of decision trees as follows. Each tree has its own training set, which is drawn with replacement from the original training set (i.e., is a bootstrap sample). After defining the training set, we grow a tree randomly selecting at each node a small number q of variables to split on; q is fixed for the whole random forest.

At each node, we then calculate the best split based on selected q variables in the training set. The best split is defined as one that does the best job of separating the data into groups where a single class predominates in each group. This is done by the use of purity or diversity measure, which evaluates a potential split: at each node we make a split that has the highest purity measure (or the lowest diversity measure). One of the examples of diversity measure is the Gini measure, which is equal to a sum of the squared proportions of the classes in nodes produced by the split.

All trees are grown fully, that is, remain unpruned. After a large number of trees is generated, each tree can be considered as a separate classifier that makes a prediction for a new object. These predictions, that may be different for different trees, are called *votes*. As a result, trees vote for the most popular class, and the random forest predicts the class with the highest rate of votes.

Since for each decision tree, a training set is a bootstrap sample taken from the original training set, each example is not considered in this training set in about one third of decision trees. This allows us to obtain an internal estimate of the accuracy of a classifier without applying a random forest to a test set or launching the leave-one-out procedure. For each example, we select the trees where this example was *an out-of-bag example*, that is, was not in the training set, and consider votes of these trees only. As a result, we obtain predictions for each example in the original training set. This is called *the out-of-bag classifier.*

Another useful feature of random forests we are going to use is a proximity matrix. Proximities $\mathrm{Prox}(i, j)$, $i, j = 1, \ldots, n-1$, where $n-1$ is the number of examples in a training set, are defined by a given random forest as follows. First, set all proximities equal to 0. After a random forest is grown, the data are run down each decision tree. If objects i and j both land at the same terminal node, $\mathrm{Prox}(i, j)$ increases by 1. At the end of the run, all proximities are divided by the number of trees in the run. Thus, all proximities $\mathrm{Prox}(i, j)$ form a matrix with diagonal elements equal to 1.

Random forest proximities provide a measure of how close to each other two objects are regardless of their labels and can substitute Euclidian distance in some algorithms. Benefiting from robustness of random forests, proximities have proved to be robust to mixed variable type, noisy and missing data. The advantages of random forest proximities were verified in clustering [8; 50] and locating outliers [8].

3.1.1.2 A Strangeness Measure Derived from Random Forests

A confidence machine is defined by its strangeness measure. In this subsection we designed strangeness measures based on random forests.

In each case we use the following notation: suppose we are given a bag $\wr(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\int$, $(x_i, y_i) \in \mathbf{Z}$, and we need to define a strangeness score $A(x, y) = A(\wr(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\int; (x, y))$. Alternatively, we can define a conformity score $B(x, y) = 1 - A(x, y)$ when it is more intuitive.

The strangeness/conformity measures we propose are the following:

1. A random forest is constructed from a training set $\{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}$. The conformity score of another example (x, y) is then equal to the percentage of correct predictions given for x by decision trees in the constructed random forest, that is, the proportion of trees in the random forest that vote for label y for object x.

2. Conformity measure 1 is the most natural one, however, it is computationally inefficient: when considering example n we have to construct

random forests nL times, where L is the number of labels. For this reason we propose another conformity measure, which will require only one random forest when making a prediction for a new object. This conformity measure is based on the internal error rate of a random forest. The random forest is grown for the union of a bag $\wr(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\wr$ and another example (x, y). Recall that for each decision tree, the training set is a bootstrap sample so that a new example is not included in this training set in about one third of decision trees. For each (x, y) we aggregate the votes for this example only of those decision trees where this example is out-of-bag. The conformity score is then equal to the proportion of correct votes for (x, y) among these trees.

We will refer to a confidence machine based on this strangeness measure as **CM-RF**.

3. These measures are based on random forest proximities $\text{Prox}(i, j)$, $i, j = 1, \dots, l+1$. To calculate a strangeness measure, we construct a random forest for the union of a bag $\wr(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\wr$ and a separate example (x, y) and form the corresponding $(l+1) \times (l+1)$ matrix of proximities for objects $x_1, x_2, \dots, x_l, x_{l+1} = x$.

   a) For this type of confidence machine, the strangeness measure of an example is the ratio of the average proximity of the example with examples of other classes to the average proximity of the example to examples of the same class.

      Strictly speaking, the strangeness score is defined as follows:

$$A(x, y) = \frac{A(x, y)^-}{A(x, y)^+},$$

      where

$$A(x, y)^+ = \frac{\sum_{y_i = y} \text{Prox}(i, l+1)}{|\{i = 1, \dots, l : y_i = y\}|},$$

$$A(x, y)^- = \frac{\sum_{y_i \neq y} \text{Prox}(i, l+1)}{|\{i = 1, \dots, l : y_i \neq y\}|}$$

are the mean values of proximities with examples from the same class and examples from the other classes, respectively. This strangeness measure was proposed by Huazhen Wang and Fan Yang in our personal communication. Below you can find its more efficient modification.

b) This strangeness measure is similar to the one described above. The modification is analogous to the k-nearest neighbour method: when calculating $A(x,y)^-$ and $A(x,y)^+$, we consider only proximities of those k examples that have the greatest values of proximities among examples of the same class y and among all the other examples, respectively:

$$A(x,y)^+ = \sum_{s=1}^{k} \mathrm{Prox}(i_s, l+1),$$

$$A(x,y)^- = \sum_{s=1}^{k} \mathrm{Prox}(j_s, l+1),$$

where i_s and j_s are the numbers of examples with s-st greatest value of proximity with example (x,y) among examples labelled with the same label y and among all the other examples, respectively.

This strangeness measure can be also obtained from the k-nearest neighbour strangeness measure described in Section 2.1.2 by substituting Euclidian distance by random forest proximities.

We will refer to a confidence machine based on this strangeness measure as **CM-RF-kNN**, where k is a number of nearest neighbours considered.

3.1.2 Venn Machines Based on Random Forests

We embedded random forests in confidence machines. Now we will deploy this algorithm when constructing taxonomies for Venn machines. We propose several implementations of a Venn taxonomy based on a random forest classifier. Similarly to confidence machines, we expect designed Venn machines to inherit advantages of random forests.

The notation we are going to use is the following: suppose $\mathbf{Y} = \{1, 2, \ldots, M\}$; we are given a bag $\Lbag (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \Rbag$, $(x_i, y_i) \in \mathbf{Z}$, $i = 1, \ldots, n$; and we need to partition this bag into categories.

We have to define the set of categories $T = \{\tau_1, \ldots, \tau_K\}$ and taxonomy functions $A_n : \mathbf{Z}^{(n-1)} \times \mathbf{Z} \to T$, $n \in \mathbb{N}$, which classify the relation between an example and the bag of the other examples:

$$\tau_i = A_n \left((x_i, y_i), \Lbag (x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n) \Rbag \right) . \quad (3.1)$$

Even after fixing the underlying algorithms, we can define categories in different ways. We propose partition into categories according to random forest predictions, votes or proximities. The sets of categories and taxonomy functions are laid out below.

1. The first version of Venn machines is based on random forest *predictions* and is referred to as **VM-RF1**. A random forest is grown for the whole bag $\Lbag (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \Rbag$, and out-of-bag predictions $\tilde{y}_1, \ldots, \tilde{y}_n$ are calculated for each object.

   Each category corresponds to a single label: $T = \mathbf{Y}$; and the class of a relation between an example and a bag of the other examples is an out-of-bag prediction for this example:

   $$A_n \left((x_i, y_i), \Lbag (x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n) \Rbag \right) = \tilde{y}_i.$$

   In other words, two examples belong to the same category if and only if their out-of-bag predictions coincide.

   The number of categories in this case is equal to the number of classes M.

2. Another version of Venn machine taxonomies is based on random forest *votes*. Again, a random forest is grown for the whole bag $\Lbag (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \Rbag$, and for each example (x_i, y_i) we calculate its votes $v_1^i, \ldots, v_M^i$ for different classes output by the out-of-bound classifier.

A) In a *binary* classification problem ($M = 2$), votes produce two values for each example: v_1^i and $v_2^i = 1 - v_1^i$. We first fix a parameter, a positive integer K', which will be equal to the number of categories K. We consider the set of out-of-bag votes for class 1: $\{v_1^i\}$, $i = 1, \dots, n$ and divide them into K' groups of similar size by the use of quantiles as described below.

Let $L_0 = 0$ and $L_1, L_2, \dots, L_{K'}$ be the integers closest to n/K', $2n/K'$, $3n/K', \dots, n$. Let $\{w_1^i\}$, $i = 1, \dots, n$ be the sorted sequence of $\{v_1^i\}$, $i = 1, \dots, n$. We partition the set of non-negative real numbers by division points $w_1^{L_0}, w_1^{L_1}, \dots, w_1^{L_{K'}}$. The category τ_i of example (x_i, y_i) is then defined as the number of the interval formed by these division points where value v_1^i falls:

$$A_n\left((x_i, y_i), \lbag (x_1, y_1), \dots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \dots, (x_n, y_n) \rbag\right) \\ = \left\{j = 1, \dots, K' : w_1^{j-1} < v_1^i \le w_1^j\right\}.$$

Thus, two examples belong to the same category if and only if their votes for class 1 fall into the same interval formed by quantiles of the set of these votes.

We construct categories of equal size because the small size of categories may result in overfitting and will lead to probability distributions very different from each other. This will be punished by the large diameter of a probability intervals. On the other hand, large categories may result in underfitting and may be punished by producing forced predictions with probability intervals close to 0. In both cases, output predictions will be not reliable.

This kind of Venn machine is further denoted by **VM-RF2A**. In this case, the number of categories is not fixed and can be equal to any positive integer $K = K'$.

B) In a *multilabel* classification problem ($M > 2$), votes produce at least three values for each example, and the intuitive approach developed for a binary classification is not natural to apply. In the case of a multilabel problem, we will combine both partitions by predictions and partitions by votes.

We first fix a positive integer K'. It will determine, but will not equal, the total number of categories. We divide all example into groups according to their out-of-bag predictions. Then within these groups we perform subpartitioning according to votes. However, instead of the set of votes for class 1 as in binary classification, we consider the following sets (let us assume we partition a group of examples with the out-of-bag predictions equal to l):

- maximum votes for each example $v_i := \max_j v_j^i = v_l^i$, $i = 1, \dots, n : \tilde{y}_i = l$, that is, votes for class l;
- or differences between the maximum vote (a vote for class l) and the second maximum vote for this example $v_i = v_l^i - \max\{v_j^i : j = 1, \dots, l-1, l+1, \dots, K'\}$, $i = 1, \dots, n : \tilde{y}_i = l$.

Corresponding Venn machines are denoted **VM-RF2B1** and **VM-RF2B2**, respectively.

Then within each group with the same out-of-bag prediction l we put $L_0 = 0$ and define $L_1, \dots, L_{K'}$ similarly to the way used in a binary classification problem: as integers closest to n_l/K', $2n_l/K'$, $3n_l/K'$, $\dots$, n_l, where n_l is the number of examples predicted to be of class l. We then sort a corresponding set of values $\{v_i : i = 1, \dots, n : \tilde{y}_i = l\}$ into a sequence $\{w_l^i\}$, $i = 1, \dots, n_l$ and partition the set of non-negative real numbers by division points $w_l^{L_0}, w_l^{L_1}, \dots, w_l^{L_{K'}}$. The category τ_i of example (x_i, y_i) is then defined as a pair of numbers: an out-of-bag prediction $\tilde{y}_i$ and the number of the interval formed by division points $\{w_l^i\}$ where value v_i falls:

$$\begin{aligned} &A_n\left((x_i, y_i), \wr(x_1, y_1), \dots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \dots, (x_n, y_n)\int\right) \\ &\qquad = \left\{\tilde{y}_i, \{j = 1, \dots, K' : w_{\tilde{y}_i}^{j-1} < v_i \le w_{\tilde{y}_i}^{j}\}\right\}. \end{aligned}$$

Thus, two examples belong to the same category if and only if they are predicted to be of the same class and their votes for this class (or difference between them and the second maximum votes) fall into the same interval.

The total number of categories is $K = M \times K'$ and can be any number divisible by the number of classes M. Taxonomies VM-RF2B1 and VM-RF2B2 can be also applied to binary classification problems; in this case they produce identical Venn machines.

3. The last version of Venn taxonomies is based on random forest *proximities* and is denoted by **VM-RF3**. A random forest is grown for the whole bag $\wr(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\int$, and proximities between each pair of examples $\mathrm{Prox}(i, j)$, $i, j = 1, \ldots, n$ are generated. We then launch the procedure used when constructing a 1-nearest neighbour taxonomy (see Section 2.1.4) but maximise random forest proximities $p(i, j)$ instead of minimizing Euclidian distances $d(x_i, x_j)$, that is, the category of example x_i, y_i is defined as the label of the example with the highest proximity in the bag:

$$A_n\left((x_i, y_i), \wr(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)\int\right) = y_j \ ,$$

where

$$j = \arg \max_{j=1,\ldots,i-1,i+1,\ldots,n} \mathrm{Prox}(i, j) \ .$$

This taxonomy has the number of categories K equal to the number of classes M.

All Venn taxonomies designed above use random forests in the out-of-bag mode. These implementations can launch random forests in a leave-one-out procedure, but it will be computationally inefficient. Venn machines defined above require to grow M random forests when a new object x_n is added, where M is a number of classes. Should we use the leave-one-out mode, the number of random forests to be grown increases in n times. For this reason, in all computational experiments out-of-bag versions of Venn taxonomies are applied.

3.1.3 Venn Machines Based on SVMs

The SVM is a widely used machine learning classifier. There are known implementation of confidence machines based on SVMs (see Section 2.1.4), but no Venn machines have been developed by the use of SVMs so far. In this section we are filling this gap and proposing several implementations of Venn taxonomies based on SVMs. Our implementations are based on SVM predictions, Lagrange multipliers and distances to the optimal separating hyperplane. They are all developed for a binary classification problem.

We are going to use the same notation for a classification problem: suppose $\mathbf{Y} = \{-1, 1\}$; we are given a bag $\wr(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\int$, $(x_i, y_i) \in \mathbf{Z}$, $i = 1, \ldots, n$; and we need to partition this bag into categories. We are required to define the set of categories $T = \{\tau_1, \ldots, \tau_K\}$ and taxonomy functions $A_n : \mathbf{Z}^{(n-1)} \times \mathbf{Z} \to T$, $n \in \mathbb{N}$, which classify the relation between an example and a bag of the other examples:

$$\tau_i = A_n\left((x_i, y_i), \wr(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)\int\right).$$

The notation related to SVMs is introduced in Section 2.1.2.

1. The first option is based on *predictions* output by an SVM and is denoted by **VM-SVM1**. An SVM is constructed for the bag $\wr(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\int$ with the optimal separating hyperplane $w \cdot x + b = 0$. The categories correspond to different labels: $T := \mathbf{Y} = \{-1; 1\}$. The relationship between an example and other examples is set to be a label which is predicted by this SVM

$$A_n\left((x_i, y_i), \{(x_1, y_1), \ldots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \ldots, (x_n, y_n)\}\right) \\ := \operatorname{sgn}(w \cdot x_i + b).$$

   In other words, objects belong to the same category if and only if they are predicted as of the same category by the SVM. As a result, the number of categories is equal to two.

2. This Venn taxonomy is based on signed *distances* to the optimal separat-

ing hyperplane and is further referred to as **VM-SVM2**. It represents a generalisation of the first version of a Venn machine which allows us to have any number of categories.

We fix the number of categories $K = K'$. We then launch an SVM for the whole bag $\wr(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\int$ and construct the optimal separating hyperplane $w \cdot x + b = 0$. For each example we calculate the signed distance to this hyperplane $d_i = w \cdot x_i + b$ (or we can equivalently consider $w \cdot x_i$) and divide the set of d_i into K' categories of approximately the same size the same way as we did when constructing Venn taxonomy based on random forest votes.

We set $L_0 = 0$ and $L_1, L_2, \dots, L_{K'}$ to be integers closest to n/K', $2n/K'$, $3n/K', \dots, n$. Let $\{d'_i\}$, $i = 1, \dots, n$ be the sorted sequence of $\{d_i\}$, $i = 1, \dots, n$. We then partition the set of non-negative real numbers by division points $d'_{L_0}, d'_{L_1}, \dots, d'_{L_{K'}}$. The category τ_i of example (x_i, y_i) is then defined as the number of the interval formed by these division points where the corresponding distance d_i falls:

$$A_n\left((x_i, y_i), \{(x_1, y_1), \dots, (x_{i-1}, y_{i-1}), (x_{i+1}, y_{i+1}), \dots, (x_n, y_n)\}\right) \\ := \left\{j = 1, \dots, K' : d'_{j-1} < d_i \leq d'_j\right\}.$$

3. The last version of a Venn machines is based on *Lagrange multipliers* (denoted by **VM-SVM3**). We launch an SVM for the bag $\wr(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\int$ and calculate the optimal values of Lagrange multipliers obtained by solving the dual problem α_i, $i = 1, \dots, n$.

   Since values of Lagrange multipliers reflect their strangeness, we can use them for constructing Venn machines grouping together examples with close Lagrange multiplier values. Examples with $\alpha_i = 0$ are the ones which conform with the SVM model and will form two separate categories:

   (a) $\{x_i : \alpha_i = 0 \;\&\; w \cdot x_i + b > 0\}$,

   (b) $\{x_i : \alpha_i = 0 \;\&\; w \cdot x_i + b < 0\}$.

We therefore group non-strange SVM examples in two categories discriminating between those examples that are located on different sides of the optimal separating hyperplane $w \cdot x_i + b = 0$.

The other examples — with $\alpha_i \neq 0$ — represent support vectors, which define the separating hyperplane. Again, we consider separately the ones on different sides of the optimal hyperplane, and within these two groups we partition examples in a preset number K' of categories of approximately similar size. The partitioning of these groups ($\{\alpha_i : i = 1, \ldots, n,\ \alpha_i \in (0; C]\ \&\ w \cdot x_i + b > 0\}$ and $\{\alpha_i : i = 1, \ldots, n,\ \alpha_i \in (0; C]\ \&\ w \cdot x_i + b < 0\}$) is carried out by their K'-quantiles. The procedure is analogous to the one described in the previous Venn machine implementation.

Thus, the total number of categories is equal to $K = 2 + 2K'$, where $K' \in \mathbb{N}$.

3.2 Algorithmic Testing

Several implementations of confidence and Venn machines were designed in the previous section. Here we perform testing of proposed algorithms. We describe the set of experiments and discuss the obtained results.

3.2.1 Data

To check how useful the new confidence machines and Venn machines can be, we applied them to real-world data. The data sets we analysed include medicine-related data sets, among them mass spectrometry and microarray data. We pay much attention to medical data because medicine is an important area of machine learning application, and medical data sets are usually difficult to classify. More details on mass spectrometry can be found in Section 4.1.

In our experiments we used six mass spectrometry data sets, two medical non-proteomic data sets and two non-medical data sets. Below we first give the data set name used throughout the thesis and then its brief description.

Proteomic data sets comprise:

- *UKOPS*: pre-processed mass spectrometry data of ovarian cancer from the UKOPS [56]. Each object is represented as a list of peak intensities and is classified as healthy, benign and malignant. The data comprises all patients of both training and test sets without borderline samples. Out of all mass spectrometry profile peaks we considered only those 109 that are presented in at least 10% of mass spectra. More detail about mass spectrometry can be found in Section 4.1.2. The data were pre-processed as described in Section 4.2.1.

- *UKCTOCS OC*, *UKCTOCS BC*, *UKCTOCS HD*: pre-processed mass spectrometry data of ovarian cancer, breast cancer and heart disease samples, respectively, collected in the UKCTOCS [38; 57]. For more detail on mass spectrometry data, see Section 4.1.2; more information on the UKCTOCS data can be found in Section 4.2. Each object is a list of peak intensities. In the data sets, all samples provided in triplets are considered; if there are several measurements taken from the same case patient in the UKCTOCS OC data, we consider only the one closest to the moment of the diagnosis, eliminating the others together with corresponding controls. Only those peaks are analysed that are present in at least one third of all mass spectra in the corresponding disease data set.

- *Competition*: mass spectrometry data provided by the Leiden Clinical Mass Spectrometry Proteomic Diagnosis Competition and preprocessed as described in [22]. The data comprise 2 classes: 77 healthy controls and 76 breast cancer patients.

- *7 biomarkers*: Ciphergen's 7 biomarkers of the UKOPS data [45].

The other medical data sets are:

- *Abdominal pain*: shortened Abdominal pain data [23], which comprises 135 binary features, symptoms of acute abdominal pain. Each patient is diagnosed with one of several diseases or labelled as 'Other'. In this thesis we considered only diagnoses 1, 5 and 8 out of 9 available. In addition, out of these 1,889 samples, we randomly selected 300 samples.

- *Microarray*: a shortened training set of microarray data of lung cancer, colon cancer and breast cancer patients provided in the International Conference on Machine Learning and Applications (ICMLA) 2009 Challenge [44]. In this thesis, we consider only the training set (400 samples instead of 650 of the combined training and test sets); 447 features were preselected out of 54,613 available features by means of statistical tests.

Two non-medical data sets were taken from the University of California, Irvine, (UCI) Machine Learning Repository [1]:

- *Sonar*: Sonar data comprising signals obtained by bouncing off metal cylinders and rocks from a variety of different aspect angles;

- *Iris*: Iris plant characteristics classified into different types of Iris.

The number of examples, features and classes for these data sets is summarised in Table 3.1. *Majority rate* shown in the last column is the percentage of examples in the dominating class, that is, the accuracy of a primitive simple predictor which predicts all objects to be of this class. We will further compare accuracy of algorithms with corresponding majority rates.

Confidence machines based on random forests were also tested on the Salmonella microarray data provided by VLA. However, it is demonstrated in Appendix C that this data set is so clean that we can achieve the accuracy of up to 100% by both confidence machines and simple predictors. For this reason, no Venn machines were applied to this data set.

3.2.2 Noise Robustness Testing

Random forests are relatively robust to outliers and noise [7]. And we expect noise robustness of this underlying algorithm to be translated into noise robustness of confidence machines and Venn machines based on random forest classifiers. Since noise and outliers are often present in data and medical data in particular, noise robustness is a desirable property. We performed testing of proposed confidence and Venn machines for robustness against noise

[1] http://archive.ics.uci.edu/ml/datasets.html

Table 3.1: Data sets used in algorithmic testing

Data set	Number of examples	Number of features	Number of classes	Majority rate
Mass spectrometry data				
UKOPS	321	109	3	53.0%
UKCTOCS OC	312	68	2	66.7%
UKCTOCS BC	162	79	2	66.7%
UKCTOCS HD	561	41	2	66.7%
Competition	153	392	2	50.3%
7 biomarkers	327	8	3	52.6%
Other medical data				
Abdominal pain	300	135	3	42.3%
Microarray	400	447	3	50.0%
Non-medical data				
Sonar	208	60	2	53.4%
Iris	150	4	3	33.3%

and compared it with noise robustness of other known confidence and Venn machines.

The following procedure was used. We applied the same algorithm in the leave-one-out mode in several runs. The first run was on the data without any noise introduced. All other runs are carried out with 10% noise injected: we changed at random labels of 10% of examples in the training set selecting a new label uniformly from other labels. In each case we calculate forced accuracy and other characteristics of performance: rates of erroneous region predictions, multiple, certain, correct certain and empty predictions for confidence machines; the average ends and length of prediction intervals for Venn machines. We run five repetitions with injected noise, and performance characteristics are averaged across these repetitions. We then compute an increase/dicrease in these values due to the noise.

This method was applied to newly implemented confidence and Venn machines as well as to known machines. Noise robustness of different methods was compared.

3.2.3 Results on Confidence Machines

For experimental testing, we implemented confidence machines CM-RF and CM-RF-kNN (for details, see Section 3.1.1). The experiments were carried out in two settings: online, to demonstrate the advantages of region predictions, and offline in the leave-one-out procedure, to compare the performance of confidence machines with simple predictors and to maximise the size of the training set.

We used the following parameters for random forest construction: the number of trees is 1000, the number of features selected at each node to split on equals a square root of the number of features. These values are recommended in [7], where it is also theoretically proved that the results converge when we increase the number of trees in random forests. In addition, our further investigation demonstrated that performance of designed confidence machines based on random forests does not considerably depend on the number of features to split on at each node. The results are given in Table A.1 in Appendix A.

We compare designed confidence machines with known implementations CM-kNN and CM-SVM (for details see Section 2.1.2.2) since both of these machines demonstrated ability to produce accurate predictions [65]. Comparison with methods other than confidence machines is beyond the scope of this thesis.

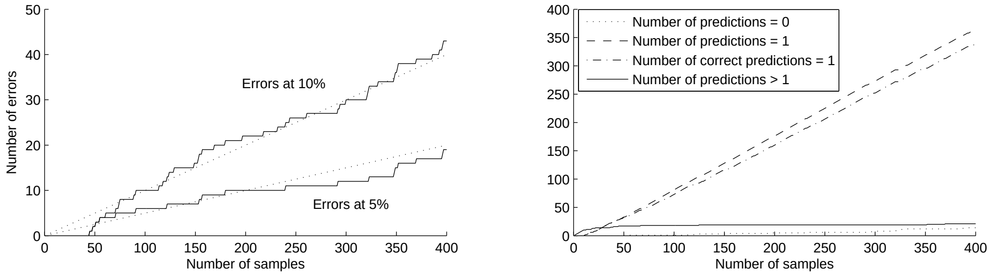
Before CM-kNN and CM-SVM are applied, the data sets are pre-processed by standardisation so that all features have zero mean and the same variance:

$$x'_{i,j} := \frac{x_{i,j} - \bar{x}_j}{\sigma_j}\,, \tag{3.2}$$

where $x_i = \{x_{i,1}, \ldots, x_{i,m}\}$; $\bar{x}_j$ and σ_j are a mean and a standard deviation of $x_{i,j}$, $i = 1, \ldots, n$. We did not apply any normalisation to the data before applying CM-RF or CM-RF-kNN because performance of random forests is not affected by normalisation.

First and foremost, the designed confidence machines proved to be valid, that is, for a given significance level $\epsilon > 0$ the rate of erroneous predictions

(predictions not containing an actual label) is close to ϵ up to statistical fluctuations. The example of the erroneous prediction dynamics is shown in Figure 3.1a. The figure demonstrates validity of the CM-RF-1NN applied to the Microarray data for significance levels $\epsilon = 5\%$ and 10% : solid lines, which represent the actual number of errors, are close to dotted lines, which demonstrate the expected number of errors for different significance levels. Validity was confirmed for both settings: online and offline, even though it is theoretically proved for the online setting only.



(a) Validity

(b) Efficiency

Figure 3.1: Validity and efficiency of CM-RF-1NN applied to the Microarray data in the online mode

Figure 3.1b demonstrates the dynamics of efficiency characteristics at the significance level of 10% of the CM-RF-1NN applied to the Microarray data in the online mode. The characteristics shown are the number of multiple predictions (with more than one label), the number of certain predictions (comprising one label) and the number of empty predictions (with no labels output). The figure demonstrates that while the number of analysed examples is low, they do not carry enough information to make certain predictions without losing validity. But starting from example 50, we have accumulated enough information so that multiple predictions cease to occur and most of region predictions contain exactly one label, which is in most cases correct. The dynamics on the plot also conforms with the empirical fact established in [65] that when multiple predictions disappear, empty predictions start to occur.

As mentioned before, all implemented confidence machines have a theoretically proved property of validity, and the general aim is to design a strangeness measure that could improve efficiency, that is, make the algorithm output as few multiple predictions and as many empty predictions as possible. Comparison of efficiency of CM-RF and CM-RF-kNN with known confidence machines demonstrated that CM-RF and CM-RF-kNN produce at least as few multiple predictions and as many correct certain and empty predictions as the known machines, and they perform much better in terms of efficiency on all mass spectrometry data sets. This allows us to speculate that confidence machines based on random forests benefit from the advantages of the underlying algorithm and perform well on noisy data and data with a lot of weak inputs. Tables 3.2, 3.3 and 3.4 summarise rates of multiple predictions, empty predictions and correct certain predictions for different confidence machines.

Table 3.2: The rate of multiple predictions for significance level $\epsilon = 10\%$ in the leave-one-out mode

	CM-						
Data	RF	RF- 1NN	RF- 5NN	1NN	5NN	SVM RBF, 5	SVM poly, 5
UKOPS	46.1%	47.0%	45.8%	74.8%	72.0%	59.2%	69.8%
UKCTOCS OC	16.0%	16.0%	13.8%	44.6%	30.8%	38.5%	79.8%
UKCTOCS BC	77.8%	78.4%	77.8%	80.9%	80.9%	81.5%	82.7%
UKCTOCS HD	56.0%	58.1%	57.2%	64.7%	59.5%	66.1%	64.0%
Competition	11.1%	18.3%	17.0%	26.8%	19.6%	32.7%	30.7%
7 biomarkers	51.1%	55.4%	53.8%	67.0%	61.2%	97.9%	96.9%
Abdominal pain	0.3%	1.0%	0.0%	3.0%	0.0%	0.0%	1.0%
Microarray	0.0%	1.5%	0.3%	13.5%	3.5%	8.5%	40.4%
Sonar	14.9%	11.1%	13.0%	13.9%	16.4%	32.2%	30.8%
Iris	0.0%	0.0%	0.0%	0.0%	0.0%	86.7%	8.0%

Confidence machines have been developed to provide region predictions with the preset error rate. However, in order to compare them with bare predictions output by simple predictors, we can ignore the nature of confidence machines and force them to always make a certain prediction. After assigning a p-value for each label to every object, we can output forced prediction — a single label with the highest p-value. If several labels have the highest p-

Table 3.3: The rate of empty predictions for significance level $\epsilon = 10\%$ in the leave-one-out mode

	CM-						
	RF	RF-	RF-	1NN	5NN	SVM	SVM
Data		1NN	5NN			RBF, 5	poly, 5
UKOPS	0.0%	0.0%	0.0%	0.0%	0.3%	0.0%	0.0%
UKCTOCS OC	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	0.0%
UKCTOCS BC	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
UKCTOCS HD	0.0%	0.0%	0.0%	0.0%	0.0%	0.5%	0.4%
Competition	0.0%	0.0%	0.0%	0.0%	0.0%	2.6%	4.6%
7 biomarkers	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Abdominal pain	5.3%	2.3%	2.7%	0.0%	4.0%	5.0%	2.0%
Microarray	5.3%	4.5%	5.3%	0.0%	0.5%	2.0%	0.6%
Sonar	0.0%	1.9%	0.0%	0.0%	0.0%	6.7%	7.7%
Iris	9.3%	18.0%	7.3%	6.0%	8.7%	6.7%	3.3%

Table 3.4: The rate of correct certain predictions for significance level $\epsilon = 10\%$ in the leave-one-out mode

	CM-						
	RF	RF-	RF-	1NN	5NN	SVM	SVM
Data		1NN	5NN			RBF, 5	poly, 5
UKOPS	46.1%	45.5%	45.5%	17.8%	21.8%	33.0%	24.0%
UKCTOCS OC	73.7%	73.7%	76.9%	45.5%	59.3%	51.6%	10.7%
UKCTOCS BC	13.0%	12.3%	13.0%	9.3%	9.3%	8.6%	7.4%
UKCTOCS HD	33.9%	32.8%	32.6%	25.3%	30.5%	23.9%	26.0%
Competition	78.4%	73.2%	74.5%	63.4%	70.6%	57.5%	59.5%
7 biomarkers	38.5%	36.4%	37.9%	25.7%	31.5%	0.6%	0.9%
Abdominal pain	89.0%	89.7%	90.7%	87.3%	90.0%	90.0%	89.0%
Microarray	89.8%	89.3%	89.5%	79.0%	87.8%	83.8%	49.7%
Sonar	77.4%	80.3%	78.8%	76.4%	74.0%	58.2%	59.6%
Iris	89.3%	80.7%	90.7%	90.0%	90.0%	3.3%	82.0%

value (we call this situation a *tie*), we make a random prediction. Thus, we launched confidence machines and random forests in the leave-one-out mode

Table 3.5: Accuracy of forced point predictions in the leave-one-out mode

		CM-						
Data	RF	RF	RF- 1NN	RF- 5NN	1NN	5NN	SVM RBF, 5	SVM poly, 5
UKOPS	72.6%	72.3%	71.5%	72.6%	55.1%	61.7%	66.7%	55.5%
UKCTOCS OC	84.9%	84.8%	83.8%	84.6%	72.3%	80.6%	78.9%	77.9%
UKCTOCS BC	66.0%	66.7%	59.0%	62.4%	50.3%	62.4%	56.2%	54.3%
UKCTOCS HD	71.8%	72.3%	69.2%	71.4%	63.2%	67.9%	62.4%	62.1%
Competition	83.7%	85.3%	83.7%	83.3%	82.0%	84.6%	86.3%	87.6%
7 biomarkers	74.6%	74.8%	72.2%	73.9%	64.5%	73.7%	60.9%	59.0%
Abdominal pain	91.7%	92.7%	91.8%	91.5%	88.0%	92.2%	91.7%	90.7%
Microarray	92.0%	91.3%	92.8%	91.4%	86.1%	89.4%	88.3%	89.5%
Sonar	85.1%	84.6%	88.7%	85.6%	86.3%	82.9%	84.6%	85.3%
Iris	95.3%	94.7%	95.0%	95.3%	93.3%	97.0%	89.3%	89.7%

and compared the accuracy of forced predictions made by confidence machines and bare predictions output by random forests.

Experiments demonstrated that, when forced to make point predictions, confidence machines perform similarly to random forest algorithm (see Table 3.5). This can be explained by the fact that each random forest is a combination of a large number of trees constructed randomly and each sample is not included in the training set for about one third of all trees in a random forest.

This implies that we can add the framework of conformal prediction to the random forest algorithm without losing in accuracy while benefiting from conformal predictions: we can produce valid region predictions and compliment each prediction with confidence.

The results of comparison of forced accuracy of different confidence machines (Table 3.5) were in line with efficiency comparison: CM-RF and CM-RF-kNN considerably outperformed other predictors on most mass spectrometry data sets and were at least as good as the known confidence machines on all data sets.

Confidence machines based on random forests were also tested on robustness against noise (see Section 3.2.2 for algorithm description). We expected accuracy of CM-RF and CM-RF-kNN to change insignificantly when noise is introduced and also aimed at comparing noise robustness of CM-RF and CM-RF-kNN with other confidence machines. The experiments demonstrated that CM-RF and CM-RF-kNN proved to be robust against noise in the data: there was a slight loss of at most 2.5% in forced accuracy when 10% noise was introduced in the training set. This can be seen in Tables A.2 and A.3 in Appendix A. The tables represent the difference in accuracy, erroneous prediction rate, empty prediction rate, certain prediction rate, correct certain prediction rate and multiple prediction rate at significance level of 10% caused by injected 10% noise. The UKCTOCS BC data set is not included in the tables because all experiments on this data set resulted in forced accuracy below the majority rate.

Other confidence machines also appeared to be noise robust; however, they can experience higher loss in forced accuracy: up to 5.1% and 8.3% for CM-SVM and CM-kNN, respectively, when 10% is injected. In addition, one should take into account that CM-RF and CM-RF-kNN have substantially higher forced accuracy than other confidence machines on mass spectrometry data and their performance does not suffer decrease when noise is introduced.

When we consider performance of confidence machines as region predictors, one could notice that after noise injection the number of erroneous predictions plunged due to an increase in the number of multiple predictions. Meanwhile corresponding characteristics of CM-SVM and CM-kNN changed only slightly. This fact allows us to speculate that confidence machines with strangeness measures derived from random forests treat noise as not sufficient information for making certain predictions.

3.2.4 Results on Venn Machines

We investigated designed Venn machines based on random forests and SVMs and compared their performance with the known implementation VM-1NN based on a 1-nearest-neighbour algorithm described in Section 2.1.4.2. Comparison of designed multiprobability predictors with methods other than algorithms of the same class (that is, Venn machines) is beyond the scope of this thesis.

When constructing random forests we used the same parameters as for confidence machines based on random forests: the number of features to split on in each node is equal to the square root of the total number of features. Investigation (given in Table A.4 in Appendix A) demonstrated that the results do not substantially depend on the number of features to split on at random forest nodes.

The experiments were carried out in the offline mode by means of the leave-one-out procedure. Before the Venn machines based on an SVM and a 1-nearest neighbour are applied, the data sets are pre-processed by standardisation so that all features have zero means and the same variance. There was no normalisation applied before Venn machines based on random forests since the performance of random forests does not depend on normalisation.

Let us also recall that Venn machines constructed by the use of an SVM can be applied only to a binary classification problem while Venn machines derived from random forest and 1-nearest-neighbour algorithms can be used for any number of classes.

3.2.4.1 Validity

The output of Venn machine is a set of label probability distributions and is therefore complex. However, it can be interpreted as a singleton prediction with the interval of probability that it is correct (probability interval), or its complementary interval — error probability interval (see Section 2.1.4).

The plots that were constructed for error probability intervals of designed Venn machines applied to different data sets confirmed validity of Venn machine predictions. An example of Venn machine validity demonstration can be seen on Figure 3.2: it represents results of VM-RF2A applied with 5 categories to the Sonar data set. The horizontal axis shows the number of observed examples. The vertical axis shows the cumulative values of: (1) errors (a solid line); (2) lower and (3) upper error probabilities (two dot-dashed lines). It can be seen from the plot that the cumulative number of errors is covered or almost covered by the area between the cumulative lower error probabilities

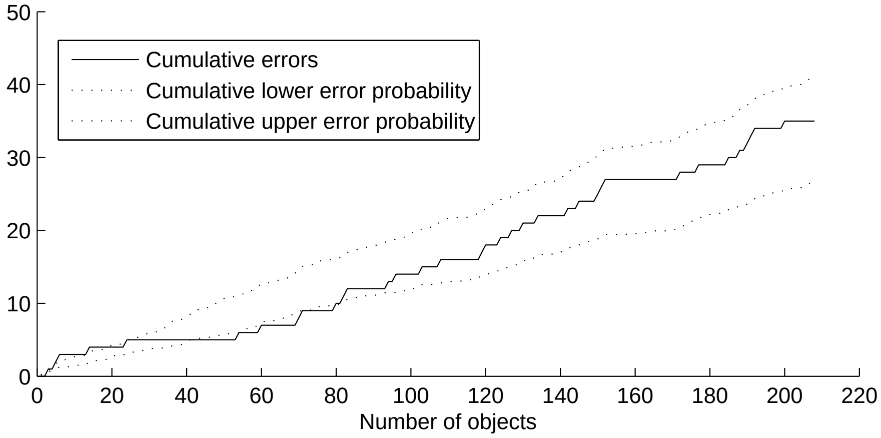


Figure 3.2: Validity of Venn machine VM-RF2A with 5 categories applied to the Sonar data in the leave-one-out mode: the number of errors lies between the cumulative lower error probabilities and cumulative upper error probabilities up to statistical fluctuations

and cumulative upper error probabilities.

One of implications of validity is that forced accuracy on the whole data set falls between the average lower and upper probabilities or is close to one of them. This can be seen in further experimental results in Tables A.5–A.12 in Appendix A.

If we applied a probability prediction method (i.g., Bayesian approach), we could output a singleton label prediction with assigned probability that this prediction is correct. But we can see from Figure 3.2 that error probability intervals output by Venn machines are narrow (0.06 on average); hence their interpretation is close to singleton probabilities. This also makes probability predictions informative since the upper error probability is close to the error rate and therefore is not too high on average.

However, this is not the case for all data sets, Venn machines and their parameters we used in our experiments. For this reason, when considering different Venn machines, we should compare how accurate forced predictions are by calculating forced accuracy. Second, we should evaluate efficiency — how informative probability intervals are. If lower probability of a predicted label is small, this prediction cannot be considered as reliable. There may be another label with empirical probabilities slightly worse than probabilities of a predicted label. In this case, since both of labels have low empirical probabilities, we cannot be sure in either of them.

3.2.4.2 Preliminary Analysis on the Sonar Data

We designed several taxonomies based on SVMs and random forests, and each of them depends on a number of parameters. We carry out thorough analysis of all taxonomy types with the different number of categories and different values of taxonomy parameters (e.g., SVM kernels) only on the Sonar data. We will make preliminary conclusions and will then apply only selected taxonomies to the rest of the data to confirm these conclusions. The Sonar data set comprises only two classes; we can therefore apply all types of Venn taxonomies designed in this chapter, including the ones based on SVMs.

Table 3.6 summarised which Venn machines and with which parameters are applied to the Sonar data. The corresponding number of categories in Venn taxonomies is shown in the last column.

Table 3.6: Venn taxonomies applied to the Sonar data set, their parameters K' and the corresponding number of categories K

Venn taxonomy	K'	Number of categories K
VM-1NN	N/a	2
VM-RF1	N/a	2
VM-RF2A	2, 5, 10	2, 5, 10
VM-RF2B2	1, 3, 5, 7	2, 6, 10, 14
VM-RF3	N/a	2
VM-SVM1	N/a	2
VM-SVM2	2, 5, 10	2, 5, 10
VM-SVM3	1, 2, 4	4, 6, 10

When Venn machines based on SVMs are applied, the following kernels and their parameters are used:

- Gaussian radial basis function (RBF) with a scaling factor $\sigma = 0.2, 1, 5$
- linear kernel

- polynomial kernel of order 5 and 10

The detailed results of these experiments can be found in Tables A.5 and A.6 in Appendix A.

The experiments allowed us to make the following tentative conclusions on **Venn machines based on random forests**. Forced accuracy of VM-RFs does not substantially depend on the type of Venn taxonomy and the number of taxonomy categories. The accuracy is always in the range of 84.1–86.5%, while accuracy of a bare random forest is 85.1%. Thus, the framework of Venn machine allowed us to obtain singleton predictions with the same accuracy as the underlying algorithm, but each individual prediction is complemented with the interval of probability that this prediction is correct, and these probability intervals are valid.

As for probability intervals, the larger the number of categories, the more output probabilities differ from each other, hence the wider the probability intervals we obtain (when the type of Venn machine is fixed). This was confirmed in our experiments. Output probability intervals were narrow (at most 0.08 in most cases). As a result, due to the property of validity, which holds true, average lower probabilities were close to forced accuracy, but for larger numbers of categories lower probability tends to be lower than for smaller numbers of categories.

Since Venn machines based on random forests did not demonstrate considerable dependance on the type of Venn machines, on the other data sets we will consider only several types of them: VM-RF1, VM-RF2A for binary classification or RF2B2 for multilabel classification, but we will vary the number of categories where it is possible.

When it comes to **Venn machines based on SVMs**, we can observe that forced accuracy fluctuates from 53.4% (which is the ratio of the dominating class in the Sonar data) to 100% and considerably depends on Venn taxonomy type, kernel selection, a kernel parameter and sometimes on the number of categories.

Comparison with bare SVMs with corresponding kernels and parameters (their accuracy is provided in Table A.7 in Appendix A) does not reveal any apparent connection between accuracy of SVMs and forced accuracy of Venn machines based on these SVMs.

The probability interval output by Venn machines based on SVMs can be very wide (up to the whole $[0, 1]$ interval, in which case the multiprobability prediction is vacuous).

The results show that VM-SVM3 does not perform well on the Sonar data: either the accuracy is close to the majority rate (53.4%) or the probability interval is too wide (at least 0.75). For this reason, we will exclude VM-SVM3 from experiments run on the other data sets.

VM-SVM2 performs better: there are combinations of kernels, their parameters and the number of categories which provide accuracy considerably higher than the majority rate (such as 73.6–88.5%) and average probability intervals with the width of 0.20–0.63. Therefore, distances to the optimal separating hyperplane may be more useful than Lagrange multipliers for constructing Venn taxonomies.

VM-SVM1 has very few implementations with good performance, and we will not use this Venn machine in our further analysis of the other data sets. We also prefer VM-SVM2 to VM-SVM1 because these VM-SVM2 are more flexible in terms of the number of categories.

Thus, VM-SVM2 is the only Venn machine based on SVMs which will be applied to the other data sets. The following kernels/paramteres of SVMs will be considered:

- a linear kernel
- a polynomial kernel of order 5
- an RBF kernel with a scaling factor $\sigma = 5$

The number of categories is varied ($K = 2$, 5, 10).

When Venn machines based on different algorithms are compared, one can see that all underlying algorithms can provide accuracy of about 87–88%, but random forests produce it regardless of the taxonomy type and the number of categories while only certain combination of a Venn machine type, a kernel and its parameter result in such high accuracy. As for the probability interval length, VM-1NN outputs the narrowest intervals (average width of 0.01), and Venn machines based on SVMs output the widest intervals.

3.2.4.3 Analysis of the Other Data Sets

Table 3.7 represents the shortened list of designed Venn machines which are applied to data sets other than Sonar. As one can see, the list of applied Venn machines depends on the number of classes: two or three. VM-SVM2 is applied only with kernels and parameters listed at the end of the previous subsection.

Table 3.7: Venn taxonomies applied to data sets other than Sonar, their parameters K' and the corresponding number of categories K

Number of classes	Venn taxonomy	K'	Number of categories K
2	VM-1NN	N/a	2
	VM-RF1	N/a	2
	VM-RF2A	2, 5, 10	2, 5, 10
	VM-SVM2	2, 5, 10	2, 5, 10
3	VM-1NN	N/a	3
	VM-RF1	N/a	3
	VM-RF2B2	1, 2, 3	3, 6, 9

Detailed results of designed Venn machine performance on different data sets can be found in Appendix A: Tables A.8–A.11 for two-class data sets other than Sonar and Table A.12 for three-class data sets. These tables represent forced accuracy and average start, end, length of output probability intervals.

First, we compared forced accuracy of designed Venn machines with the accuracy of corresponding underlying algorithms. Experiments demonstrated that forced accuracy of Venn machines based on random forests was similar to the accuracy of a bare random forest and substantially higher on the UKOPS data: forced accuracy of 77.1–84.3% versus random forest accuracy of 72.6%.

For VM-SVM2, there was no always connection between forced accuracy and accuracy of bare SVMs with the same kernels and parameters. For example, on the Competition data, bare SVMs can achieve accuracy (88.2%) higher than the one provided by random forests or Venn machines based on random forests (up to 84.3%), but VM-SVM2 does not achieve this accuracy (up to 77.1%). Similarity of forced accuracy and accuracy of the underlying SVM was still detected on the UKCTOCS OC data (when linear and RBF kernels were used) and the UKCTOCS HD data (when linear and polynomial kernels were used).

The experiments show that when the random forest is used as an underlying algorithm, forced accuracy does not substantially depend on the Venn machine type (VM-RF1 or VM-RF2A) and the number of categories. Forced accuracy of VM-SVM2 considerably depends on the kernel and often on the number of categories.

Now we can compare forced accuracy of different Venn machines with each other. We will skip the UKCTOCS BC data because all Venn machines and underlying algorithms perform badly on this data set: accuracy does not exceed the ratio of the dominating class in the data. On the other data sets, Venn machines based on random forests outperform all other machines (except for the Sonar data where all underlying algorithms can produce the accuracy of 87–88%). As for SVMs, VM-SVM2 can usually provide high accuracy, at least higher than the one of VM-1NN. However, this accuracy is not robust to changing kernels and the number of categories.

When investigating informativeness of probability intervals, we will mainly compare interval width as it reflects how close lower probability is to forced accuracy (subject to Venn machine validity). The experiments confirm preliminary conclusions made on the Sonar data. The narrowest probability intervals are output by VM-1NN (at most 0.015). Prediction intervals output by Venn machines based on random forests are reasonably narrow: not wider than 0.09 and in most cases much narrower. Finally, VM-SVM2 may produce vacuous predictions with probability interval covering almost the whole $[0, 1]$ interval.

3.2.4.4 Noise Robustness for Venn Machines Based on Random Forests

Random forests are known to be noise robust [7], and we expect Venn machines based on random forests to inherit this property from their underlying algorithm. We applied the test robustness proposed in Section 3.2.2 to the designed Venn machines based on random forests. Corresponding results can be found in Tables A.13–A.15 in Appendix A. The experiments are included in these tables only if they result in forced accuracy higher than the majority rate of the data (majority rates are provided in Table 3.1. For example, no experiments are presented for the UKCTOCS BC data since all of them provided accuracy at most 66.7%, which is the ratio of the dominating class in this data set.

The tables show that Venn machines derived from random forests demonstrated high noise robustness: when 10% noise was introduced, their forced accuracy increased by at most 4%. Meanwhile, similar noise injections led to forced accuracy rises of up to 6% and 7% when we applied VM-1NN and VM-SVM2, respectively.

3.2.4.5 UKCTOCS OC Test for Venn Machines Based on SVMs

In all experiments above, Venn machines based on SVMs were outperformed by the ones based on random forests. Our previous experience of application of bare SVMs showed that this algorithm usually performs well when the number of informative features is comparable to the half of the total number of features. Similar performance is expected from Venn machines based on SVMs.

We decided to investigate our hypothesis of SVM applicability on the UKCTOCS OC data set. The triplet analysis of this data showed that most of the information statistically significant for discrimination between two classes is contained in three features (CA125, peak 2 and peak 3; see Chapter 4 and Appendix B for more details). We could therefore process the UKCTOCS OC data changing the number of features contained in the object but always keeping three informative features among them.

Figure 3.3 demonstrates forced accuracy achieved when different numbers of features (68, 21, 6 and 4) are considered in each object. The plot represents experiments with VM-1NN (dashed line), VM-RF1 and VM-RF2A with 5 or 10 categories (solid lines) and VM-SVM2 with a linear kernel, a polynomial kernel of order 5 and an RBF kernel with $\sigma = 5$ with 5 or 10 categories (dotted lines). Results that correspond to the same processing algorithm but have different numbers of processed features are connected with a line.

On the one hand, when we reduce the number of peaks, we may get rid of some useful information; on the other hand, this way we can eliminate excessive

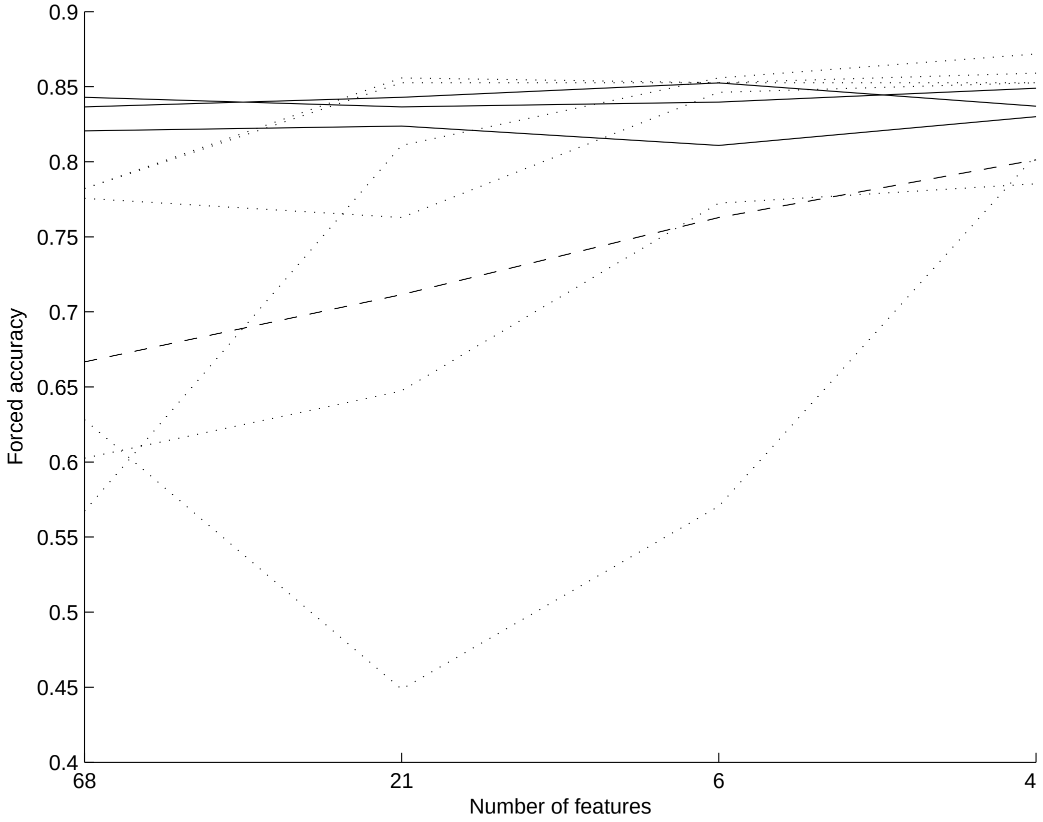


Figure 3.3: Forced accuracy of VM-1NN (dashed line), VM-RF1/VM-RF2A (solid lines) and VM-SVM2 (dotted lines) applied to the UKCTOCS OC data with the different number of features

noise. And the latter is assumed to be the case as we know that most useful information is concentrated in only three features.

The plot demonstrates that, when we consider all 68 features, all Venn machines based on random forests outperform the others, while VM-1NN and some implementations of VM-SVM2 have forced accuracy below 66.7% — the accuracy that we could achieve when predicting that all objects belong to the dominating class. When we reduce the number of features, performance of Venn machines based on random forests does not considerably change, but forced accuracy of VM-SVM2 improves and in most cases reaches the accuracy of algorithms based on random forests.

These observations have two important implications. First, they confirm that Venn machines based on random forests are robust to noise, which is presented in non-informative features. Second, the results conform well with our hypothesis that Venn machines based on SVMs perform well when the number of features is comparable with the half of the total number.

3.3 Summary

In this chapter new implementations of confidence and Venn machines were proposed: confidence and Venn machines based on random forests as well as Venn machines with the taxonomy derived from SVMs.

Experiments demonstrate that proposed confidence machines based on random forests are more efficient than known confidence machines on mass spectrometry data (and at least as efficient on other type of data) while maintaining the property of validity: they output fewer multiple predictions, and the ratio of mistakes does not exceed the preset level. Experimental results also confirm validity of Venn machines based on random forests.

When forced to produce singleton predictions, confidence machines and Venn machines based on random forests result in accuracy similar to random forest accuracy. This implies that frameworks of confidence machines and Venn machines could be applied to random forests and produce algorithms with the same accuracy, but in addition they complement each individual prediction with the measure of its reliability. That is, although confidence and Venn machines are designed to produce valid region and multiprobability prediction, they can also be a useful tool when making singleton predictions.

In comparison with other algorithms with online validity, forced accuracy of all methods derived from random forests is at least as high as accuracy produced by other algorithms of the same class (i.e., confidence machines or Venn machines) and sometimes is considerably higher. In addition, all methods based on random forests proved to be robust to noise, robust to parameters of random forest construction and, in the case of Venn machines, comparatively robust to the type of Venn taxonomy and the number of categories. All these characteristics make Venn machines based on random forests an attractive analytical tool.

Venn machines based on SVMs also proved to be valid. Their performance substantially depends on the type of the Venn taxonomy, kernel selection, a kernel parameter and the number of categories. They may produce high accuracy, but when using taxonomy based on SVMs, one should be cautious with the choice of taxonomies and the number of categories. Thus, these methods may be not very consistent and may require tuning in order to find good parameters. Venn machines based on SVM often produce wide prediction intervals, which can make predictions uninformative.

However, experiments with Venn machines derived from SVMs conformed with the hypothesis that these methods perform well on the data with the number of informative peaks comparable to the half of the total number. Therefore, we could advise that methods be applied when this requirement is expected to be satisfied.

Chapter 4

Algorithms with Online Validity for Proteomics

Prediction of ovarian cancer, breast cancer and heart disease is a critical task. For some of these diseases (e.g., ovarian cancer), it is especially crucial to make a diagnosis in their early stages, when the disease has no clinical symptoms. Proteomics and its mass spectrometry techniques can be used to address these problems.

When making a diagnosis based on mass spectrometry data, the classical machine learning approach is to predict the diagnosis without any measure of how accurate this prediction is. For this purpose, a wide range of machine learning algorithms can be applied, e.g., SVMs or decision trees.

However, it would be useful if prediction of diagnosis could be made with a level of confidence so that practitioners could have the assessment of risk error. In addition, we would like the information on prediction reliability to compliment each individual patient rather than a whole set of patients. This would allow clinicians to distinguish more confident predictions of diagnosis from uncertain ones.

Another issue we would like to address is prediction of diagnosis well in advance of the moment of clinical diagnosis or the moment of death. For example, for ovarian cancer it is crucial to identify the disease as soon as possible: if ovarian cancer is diagnosed at the early stage, it may be possible to remove the single affected ovary and fallopian tube. Thus, we are aiming at designing the methodology which would allow us to determine how well in advance of the moment of diagnosis/death we can make reliable diagnosis predictions.

Thus, we would like to introduce the methodology of mass spectrometry data analysis which focuses on these two particular problems:

- Can we develop a methodology which provides a measure of reliability tailored for each individual patient?
- Can we develop a methodology which would determine how early in advance of the moment of clinical diagnosis / the moment of death we can make reliable predictions?

In this chapter we apply frameworks of confidence and Venn machines in order to develop such methodologies. These frameworks will allow us to provide additional information on prediction reliability for each patient, and this information will be valid. Here, we develop algorithms for the analysis of mass spectrometry data rather than general algorithms with online validity designed in Chapter 3. Algorithms developed in this chapter are adjusted to address the second problem listed above, to determine how early in advance one can make reliable predictions, and allow us to pinpoint mass spectrometry profile peaks which are important for disease diagnosis and therefore could be potential biomarkers. These algorithms also take into account the nature of mass spectrometry experiments and special features of the data we analysed: serial samples and triplet setting.

4.1 Proteomics and Mass Spectrometry

In this section we give description of proteomics, mass spectrometry experiments as a tool in proteomics research and the format of output mass spectrometry data.

4.1.1 Proteomics

Proteomics is a study of proteins. Understanding the proteome, structure and functions of different proteins is crucial for development of effective diagnostic techniques and treatments of diseases.

One of applications of proteomics is identification of biomarkers, proteins which can be used as an indicator of a particular disease, disease state or another physiological state of the organism. This approach relies on the information in genome and proteome to identify proteins that can reflect a disease. For example, if the level of a particular protein in serum samples of diseased patients is higher than in serum samples of healthy patients, this protein could be a potential biomarker, and higher level of this protein in a new serum sample could point at the risk that the patient is diseased.

Thus, the most intuitive use of biomarkers is disease diagnosis. Their another application is identification of potential new drugs for disease treatment: identified biomarkers associated with the disease can be used as targets for new drugs.

One of the main diseases we focus on in this thesis is ovarian cancer. Several biomarkers for ovarian cancer have been identified [18; 21; 33; 55]. The most extensively assessed biomarker is cancer antigen 125 (CA125), that is typically elevated in the blood of ovarian cancer patients [5; 47].

4.1.2 Mass Spectrometry Experiments and Data

A number of various techniques allow us to test the level of proteins in serum, blood, urine or tissue samples. Among these techniques are Western blot, enzyme linked immunosorbent assay (ELISA) and mass spectrometry. Mass spectrometry [53] is an attractive analytical tool because it enables researchers to simultaneously analyse hundreds of biomolecules [12; 35; 68; 69].

There are several types of this technique, and the one which is used when processing data used throughout this thesis is matrix-assisted laser desorption and ionisation—time-of-flight (MALDI-TOF) [31; 32; 58]. The general description of this technology, according to [68], is presented below: "The basic operating principle . . . consists of ionising prepared biologic samples (such as blood serum or plasma) through application of a laser beam. The ionisation process creates a vaporous mixture of charged particles which is accelerated down a so-called flight tube through application of an electric field. As the time-of-flight of any particle within the mixture will depend on its mass-to-charge ratio (m/z), we may determine the m/z-distribution of the constituent particles in the ionised mixture by recording the flight times." The unit of m/z-ratio measurement is Dalton (Da).

"The mass spectrometer produces a sequence of intensity readings for each sample on a pre-defined, fixed and ordered set of contiguous bins within a given m/z-range, which discretises the signal. The recorded intensity for any bin thus corresponds to the total number of particles detected within the m/z-range spanned by that bin . . . The mass spectrum may thus be thought of as an extremely high-dimensional histogram, as the number of bins will typically be in the thousands, recording the distribution of ionised particles within a serum sample. Bins are usually chosen to be of equal length at the time scale, which implies bin widths will be exponentially increasing with m/z-value at the transformed scale [68]."

Thus, a mass spectrum can be visually represented as a set of narrow peaks of various intensity. These peaks may overlap, which makes separate peak

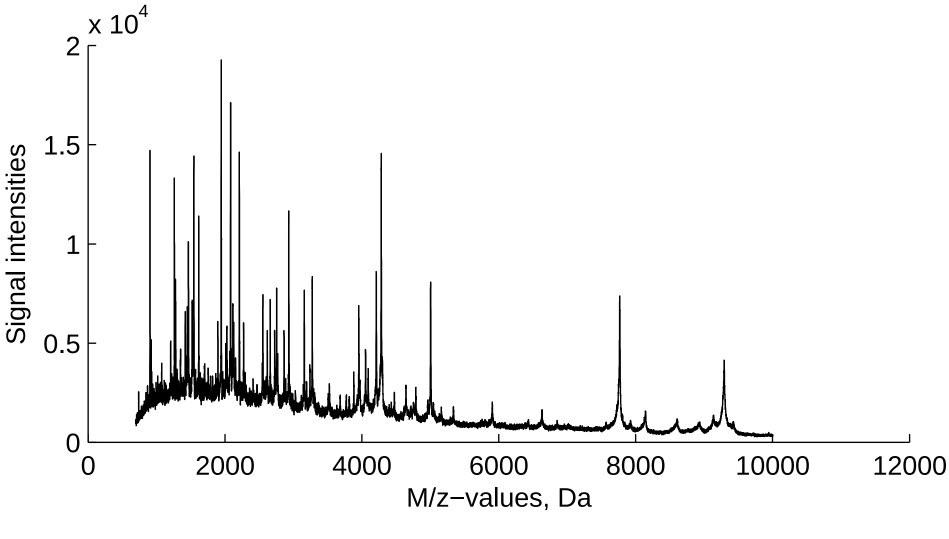


Figure 4.1: Example of a mass spectrometry plot (a UKCTOCS OC sample)

detection difficult. Figure 4.1 gives a graphical representation of one of mass spectra analysed in this thesis. Each mass spectrum consists of three elements: signal itself, baseline and noise. In the figure, one can see the baseline, a hump at the range of m/z-values closer to 0, which peaks at 1500–2000 Da and then gradually decreases back to zero towards the larger m/z-values. On top of this baseline, one can see a dense mixture of narrow peaks. Thus, to identify the peaks we need to subtract the baseline. In addition, mass spectra plots can be noisy because of physical, electrical or chemical artefacts. Pre-processing is applied to mass spectra to get rid of this systematic noise. Auxiliary goals of pre-processing are to normalise the spectra from different samples and reduce the dimensionality of the data. Pre-processing can include the following steps: smoothing, baseline subtraction, normalisation to make sure that the total amounts of ions across different samples are the same. After the true signal is extracted from mass spectra, peaks are identified in each spectrum and then aligned, that is, peaks from different spectra get related to each other and are considered as one peak. Finally, the intensities of identified peaks are calculated. The pre-processing steps applied in this work can be found in Section 4.2.

4.1.3 Limitations of Proteomics Application

Even though current proteomics methods are powerful and widely used for disease diagnosis and drug development, most current diagnostic tests focus on the same biomarkers [70]. This may happen due to limitations that proteomics approach has. Proteomic technologies are able to detect proteins in high concentrations but cannot detect single molecules. The concentration sensitivity limit of currently used approaches varies: for example, SELDI-TOF mass spectrometry has detection limit of 10^{-6}-10^{-8}M [64], and ELISA — 10^{-9}–10^{-12}M [1]. Therefore, proteomic methods enable detection of up to 20% of the protein species that are in plasma. The rest of protein species are beyond the detection ability of proteomics. This limitation is a serious obstacle of future development of proteomics technologies. The fact that the most of the proteome is not accessible by means of proteomics may be the reason why proteomics application were not very efficient recently: the quantity of new diagnostic tests declined in recent years [1]. Some important biomarkers may be at ultra-low concentrations and therefore among those proteins that are not detected by proteomics methods.

There are new methods, such as atomic force microscopy molecular detector [2], which enable detection of single molecules. Prototypes of devices that could detect single molecules are developed. These detectors will enable identification and measurements of proteins at the level of concentration close to the reverse Avogadro number, i.e., 10^{-24}M, and thus will be highly sensitive to presence of proteins in plasma.

4.2 The UKCTOCS Data

In this chapter, we develop methodologies for mass spectrometry data analysis and apply them to subsets of the UKCTOCS [1] biobank. The UKCTOCS is an extensive study, and its biobank contains serum samples and data of 202,638 women participating in this trial [40]. The women were recruited between 2001–2005 by the use of random selection from the age/sex registers of health authorities geographically related to the 13 regional collaborating centres [71]. To participate in the trial women had to be aged 50–74 years and be postmenopausal, that is, with more than twelve months amenorrhoea following a natural menopause or hysterectomy or more than twelve months of hormone replacement therapy commenced for menopausal symptoms [39; 71]. The exclusion criteria were the following [71]:

- women with history of bilateral oophorectomy;
- women with currently active non-ovarian malignancy (excluding skin cancer);
- women who have had an ovarian malignancy;
- women with high risk of familial ovarian cancer;

[1] For more information see `www.cancerresearchuk.org`.

- woman who participate in other ovarian cancer screening trials.

The women included in the trial were randomly divided into two groups of equal size (100,000 patients each): a study group and a control group. The study group was further randomly split into an ultrasound group and multimodal group (50,000 patients each). The ultrasound group experienced annual screening with transvaginal ultrasound of the ovaries. The multimodal group received annual screening with the Risk of Ovarian Cancer algorithm for serum CA125 as a primary test and ultrasound as a secondary test. No screening was applied to the control group. More details can be found in [71].

Women in the study group experienced annual screening with repeat samples collected if an abnormality was detected [38]. The serum samples underwent prefractionation using a reversed-phase batch extraction protocol prior to MALDI-TOF mass spectrometry data acquisition. The unique feature of this trial was that the women were screened for up to 5 years.

Two studies of samples from women diagnosed to have ovarian and breast cancer respectively and healthy (no cancer at follow up) patients was undertaken. A third sample set consisted of women who had died of myocardial infarction (heart disease) and healthy patients. In this study, we analyse available ovarian cancer and breast cancer data sets and to a lesser extent the data related to heart disease. The mass spectrometry data of ovarian cancer and heart disease are provided by the University of Reading; breast cancer data were processed by University College London.

The information regarding which patients were followed-up and which were missing cannot be disclosed. Hence, the interpretation of results obtained in this chapter is subject to the missingness pattern being random or the percentage of missing data being small.

The data pertain to serum samples collected from patients diagnosed with the disease (we will call them *cases*) and healthy patients (they will be referred to as *controls*). Originally, each case was accompanied by two controls matched on patient age, sample collection location and sample collection date/time to minimise differences in sample processing [57]. A case accompanied with two matched controls is called a *triplet*. In the ovarian cancer data, several measurements could have been taken from the same case patient. In such case, a set of triplets corresponding to the same case patient is called a *triplet group*.

Due to the triplet setting, the number of controls in each data set is twice as greater than the number of cases:

- 208 controls and 104 cases in the ovarian cancer data set (312 samples in total);

- 108 controls and 54 cases in the breast cancer data set (162 samples in total).

As it was mentioned before, ovarian cancer has a widely used biomarker CA125 [5; 47]. Its level is usually elevated in the blood of ovarian cancer patients. However, the potential role of this protein for the early detection of ovarian cancer is unproved and still subject to clinical trials. One of the main problems related to the use of CA125 is its low predictive ability at early-stages of the disease. Another problem is that CA125 can be produced by other mesothelium-derived tissues [57] and therefore may also be elevated in women with benign gynaecological conditions and other types of cancer (such as breast, bladder, pancreatic, liver, lung) [9]. Thus, CA125 deployment lacks sensitivity [19; 51]: if the level of CA125 is elevated, an operation is needed to confirm the disease, and this operation may result in death of the patient. Thus, it is thought that CA125 alone may not be accurate enough for detection of early-stage ovarian cancer [4].

When carrying out the analysis of the UKCTOCS OC data, we attempt to identify certain mass spectrometry profile peaks which could, in combination with CA125, result in accurate ovarian cancer diagnosis well in advance of the moment of clinical diagnosis. Thus, each mass spectrum of the ovarian cancer data set is assigned a level of CA125, and we will make predictions of the diagnosis based not only on MALDI-TOF mass spectrometry data but also on CA125 levels.

Each sample is assigned a non-negative value T — time to diagnosis confirmed by histology/cytology for ovarian and breast cancer patients and time to death for heart disease patients. Controls are assigned the same value T as the case they match. We will refer to this value as *time to diagnosis*. Since we have the information regarding when each sample was taken, we can consider sets of samples taken in different time slots before the time of diagnosis.

The UKCTOCS data sets were applied to the statistical analysis (see Appendix B and [10; 11; 15]) which was introduced in [26] using the so-called *triplet setting*. We will refer to this analysis as the *triplet analysis* throughout the thesis. This analysis of the UKCTOCS data demonstrated that there are certain time slots when mass spectrometry profile peaks carry statistically significant information for discrimination between controls and cases for the analysed diseases, i.e., we can reject the null hypothesis that the diagnosis is independent of the information contained in peak intensities at significance level of 5% well in advance of the moment of diagnosis. For example, for ovarian cancer, mass spectrometry data allow us to reject the null hypothesis for detection up to 15 months in advance of the moment of diagnosis.

In the triplet analysis, we used the information that there is exactly one diseased patient in each triplet and assumed that measurements can be compared only if they are matched. In this chapter, we mostly develop algorithms which merge all samples together and check whether samples from different triplets can be compared to each other although they are not matched by sample collection location and time. However, the algorithm we develop in Section 4.3.4 applies confidence machines in the triplet setting.

The triplet analysis also allowed us to determine statistically significant peaks which could be potential candidates for biomarkers. For example, we identified certain mass spectrometry profile peaks that in combination with CA125 level can provide reliable long term prediction of ovarian cancer. The peaks with m/z-values 7772 Da and 9297 Da were the most informative in extending the period of significant discrimination, and in combination with CA125 they proved to be able to predict the disease up to 6 and 4 months earlier than CA125 alone, respectively.

4.2.1 Applied Pre-processing

This section briefly describes the pre-processing applied to the UKCTOCS data. The data are provided in a two-column format: the first column is a list of m/z-values, the second column is a list of corresponding intensities. Calibration had been performed prior to the data being distributed; therefore, all our further pre-processing steps were applied only to the intensities, m/z-values remained unchanged. We applied the pre-processing deployed in [26]. The details of each steps are provided below.

1. **Down-sampling** was performed in order to decrease the number of m/z-values for computational optimisation purposes.

2. For noise elimination, we performed **smoothing** by averaging the intensities within a moving window.

3. **Baseline subtraction** ensured that the spectra sit on the intensity $= 0$ axis. The algorithm applied is based on finding the lowest points between some dominant local maxima (*troughs*). We define a dominant local maximum as the point with the highest intensity within some range, the width of this range is a parameter of the algorithm. We then apply Piecewise Cubic Hermite Interpolating Polynomial to all the points marked as troughs in order to construct a baseline. Further steps are applied to correct the baseline for any points where the baseline is above the spectra.

4. We performed **normalisation** to make sure that the total amount of ions across different samples were the same. The algorithm involved dividing the intensity of each point in a spectra by the sum of all intensities.

5. The goal of the **peak identification** step was to generate a list of peaks for each sample. This was achieved by identifying all local maxima in the mass spectra above an intensity threshold and above a certain signal-to-noise ratio threshold. As a result of this algorithm we had a table for each sample with the following columns:

   (a) Unique sample ID — for any single sample this column has the same number for each entry, the reason for this column will become apparent in the next step

   (b) Number of peak in the initial array of m/z-values

(c) M/z-value of a local maximum

(d) Signal-to-noise ratio within a window of the certain width

(e) Corresponding intensity

6. Peaks obtained in the previous step do not strictly coincide because of the noise and possible presence of different isotopes. Therefore, the next step (called **peak alignment**) is to find common peaks (that is, peaks with m/z-values close to each other) among the samples. We combined all peak lists constructed for individual samples into one list and sorted in descending order relative to column 5 — peak intensity. We then worked down the list taking one of the following actions for each peak:

   (a) If the peak is within some predefined range of an existing peak group and there is no other peak from the same sample in that group, then the current peak can be added to the group; else the peak is ignored.

   (b) If there are no groups close to the current peak, then a new peak group is created containing the single peak.

7. The result of the previous step is a set of peak groups, each of which can potentially contain between 1 peak and N_{ms} peaks, where N_{ms} is the number of samples in the data set. Now we have to compute peak intensities for each sample for each peak group. For those peak groups that have less than N_{ms} peaks, we need to estimate the intensities for the remaining samples. For this purpose, we set a mass separation parameter which we use to define a range in the m/z-vector given a single m/z-value — the maximum m/z-ratio of all the peaks in the group. Then the intensity of each missing sample is defined as the maximum intensity within this range in the spectrum of the sample.

So the data we apply to our methodology are represented as intensities of identified peaks. The peaks are usually sorted by their frequency: the greater the number of mass spectra containing a peak, the higher the rank of that peak. We usually consider a certain number of the most frequent peaks only. The number of most frequent peaks was determined by the properties of specific data sets. These are 5 peaks for ovarian cancer, 20 peaks for breast cancer and 41 peaks for heart disease.

Thus, every x_i is a vector of features — intensities of the most frequent peaks. Later, peak numbers are used; the lower the peak number, the more common the peak is. Please note that sets of peaks vary for different data sets; therefore, peaks with the same number from various data sets have different m/z-values.

4.3 Algorithms for Proteomic Analysis

Here we describe algorithms with online validity designed for mass spectrometry data and the UKCTOCS data in particular. On the one hand, these algorithms hedge predictions by complementing them with additional information on their reliability; on the other hand, they are specially developed for mass spectrometry data and take into account peculiarities of the UKCTOCS data.

4.3.1 Category-Based Confidence Machines Constructed on Linear Rules

Here we develop a category-based confidence machine with the strangeness measure based on linear combinations of a small number of peaks (and CA125 for ovarian cancer). The framework of category-based confidence machines allows us to assign confidence to each individual prediction and have a theoretical guarantee on the asymptotical error rate of region predictions. The taxonomy used is label-conditioned and allows us to control regional specificity and regional sensitivity by presetting significance levels for categories of healthy and diseased samples.

Practically any known machine learning algorithm can be plugged into the framework of category-based confidence machines and thus result in a new algorithm of prediction with confidence. In this section we developed the strangeness measure which produces results both comprehensible and useful for medical researchers. It also allows to pinpoint mass spectrometry profile peaks that correlate with a disease and therefore could be potential biomarkers, while other strangeness measures designed or considered in this thesis do not give such an opportunity. Thus, we describe the application of linear rules within the framework of category-based confidence machines used for discrimination between mass spectrometry samples taken from healthy and diseased patients.

When designing a new strangeness measure, we will use simple linear rules of the following type:

$$\sum_{k=1}^{h} w_k \log I(n_k) > \theta \,, \tag{4.1}$$

where $I(n_k)$ is the intensity of peak n_k, $w_k \in \mathbb{R}$, $k = 1, \ldots, h$ are weights, $\theta \in \mathbb{R}$ is a threshold. A rule classifies a patient as diseased if it returns the value *true*, healthy otherwise. If a rule of this type can discriminate healthy samples from diseased, the peaks which are included in the linear rule could be potential biomarkers. In addition, weights w_k, $k = 1, \ldots, h$, can reflect the importance of each peak (the larger the absolute value of a peak's weight, the more important the peak is) and whether the higher or lower protein level is a risk factor (depending on the sign of the corresponding weight).

In order to design a category-based strangeness measure, we first need to define taxonomy of the category-based confidence machine. We will consider a category-based taxonomy $\kappa(n, (x_n, y_n)) = y_n$, i.e., the taxonomy which consists of two categories that correspond to two different diagnoses: a category of healthy patients and a category of diseased patients. Such taxonomy will allow us to guarantee regional sensitivity and regional specificity. Hence, a p-value for the hypothesis $y_n = y$ is calculated as follows:

$$p_n(y) = \frac{|\{i = 1, \ldots, n : y_i = y \;\&\; \alpha_i \geq \alpha_n\}|}{|\{i = 1, \ldots, n : y_i = y\}|} \,, \tag{4.2}$$

that is, the p-value is calculated as the ratio of healthy (diseased) patients that are at least as strange as the new healthy (diseased) patient to the total number of healthy (diseased) patients.

We apply category-based confidence machine with this taxonomy rather that confidence machine or another category-based confidence machine because this taxonomy seems to be most natural to use in combination with the strangeness measure we are proposing.

The strangeness measure was calculated using the classification method described below. We fix the number h of peaks used in a rule. We then consider a set of possible linear rules of type (4.1) where parameters of the rules can possess the following values: $w_k \in W_k \subseteq \mathbb{R}$, $\theta \in \mathbb{R}$, $\{n_1, \dots, n_h\} \in \mathbb{P} \subseteq \{1, \dots, N\}^h$. Out of these rules, we select the following one:

$$\begin{aligned}\{\tilde{w}_1, \dots, \tilde{w}_h, \tilde{\theta}, \tilde{n}_1, \dots, \tilde{n}_h\} \\ = \arg \max_{\substack{w_k \in W_k, \theta \in \mathbb{R}, \\ \{n_1, \dots, n_h\} \in \mathbb{P}}} &(\min(\mathrm{TPR}(w_1, \dots, w_h, \theta, n_1, \dots, n_h), \\ &\mathrm{TNR}(w_1, \dots, w_h, \theta, n_1, \dots, n_h))) \,, \quad (4.3)\end{aligned}$$

where $\mathrm{TPR}(w_1, \dots, w_h, \theta, n_1, \dots, n_h)$ and $\mathrm{TNR}(w_1, \dots, w_h, \theta, n_1, \dots, n_h)$ are the true positive rate (sensitivity) and the true negative rate (specificity) of the rule (4.1) with parameters $(w_1, \dots, w_h, \theta, n_1, \dots, n_h)$, respectively, on the set of patients including a new patient with a new hypothetical diagnosis. If there are more than one set of parameters which provide maximum in arg max expression, we choose the one with the smallest absolute values of parameters giving priorities in the following order: $w_1, \dots, w_h, n_1, \dots, n_h, \theta$.

We can then define the strangeness score of a new patient with a diagnosis on the basis of the chosen rule. The value of the chosen linear combination $\sum_{k=1}^{h} \tilde{w}_k \log I(\tilde{n}_k)$ is used as a strangeness score for healthy patients or as a value negative to a strangeness score for diseased patients. Thus, when calculating a p-value, we compare the value of the chosen linear combination for the new patient with the value of the same combination for patients with the same diagnosis. If the new patient was healthy, the larger the value of the linear combination, the more non-conformal the patient is, and the other way around if the patient is diseased.

In our experiments, the significance level was the same for the classes of healthy and diseased patients. However, if we wanted to put regional sensitivity or specificity first, we could consider different significance levels for healthy and diseased samples.

Cross-validation of the method using a leave-one-out approach was performed: each patient (x_i, y_i) is considered as if it was a new test sample and all the remaining patients in the data are treated as the training set.

4.3.2 Logistic Venn Machines

The category-based confidence machine designed in the previous section outputs sets of potential labels. Now we would like to focus on probability that the patient is diseased. This is known as risk of a disease. For this purpose, we apply a framework of Venn machines which allows us to output multiprobability predictions that are valid in terms of agreement with the observed frequencies.

In order to process mass spectrometry data, we design a novel Venn taxonomy which will allow us to obtain ranking of all mass spectrometry profile peaks in terms of their importance for disease diagnosis. We can then identify peaks with the highest ranks and investigate them as potential biomarkers. Other Venn machines designed in this thesis do not have such an ability to pinpoint the most important features.

The proposed Venn taxonomy is based on a logistic regression algorithm. Since logistic regression outputs probability distributions, this Venn taxonomy will also allow us to compare performance of a Venn machine (with multiprobability outputs) and its underlying algorithm (with probabilistic outputs). This way we can get an insight into whether multiprobability predictions have any advantages over probabilistic predictions.

This Venn machine based on logistic regression was designed in collaboration with Ilia Nouretdinov.

4.3.2.1 Logistic Regression

When implementing logistic regression, we followed [54]. This algorithm outputs probability distribution of a new label as follows.

Suppose each object out of the training set $x_1, \ldots, x_{n-1}$ is an m-dimensional vector, each with corresponding labels $y_1, \ldots, y_{n-1} \in \mathbf{Y} = \{0, 1\}$. The statistical model of logistic regression is based on the assumption that y_i is 1 with probability P_i and 0 with probability $1 - P_i$, where

$$P_i = \frac{1}{1 + e^{-<x_i,b>}} \tag{4.4}$$

and

$$1 - P_i = \frac{1}{1 + e^{<x_i,b>}} \, . \tag{4.5}$$

The m-dimensional vector b is an unknown parameter of the model and can be interpreted as signed weights of different attributes. An additional value of '1' may be appended to each x_i to allow a free additive term to $< x_i, b >$.

The optimisation goal for logistic regression is:

$$\sum_{i=1}^{n-1} \log \left(1 + e^{(-1)^{y_i} \langle x_i, b \rangle}\right) + a \, \langle b, b \rangle \to \min_b \, . \tag{4.6}$$

This formula is based on the maximum likelihood estimation for logistic regression with an added regularisation term $a \, \langle b, b \rangle$ to ensure that a minimum always exists and to avoid overfitting. In this work we always set $a = 0.1$. The above minimisation problem can be solved by the gradient descent method. Denote by $\hat{b}$ the solution of the optimisation problem above.

For a new object x_n, the probabilistic prediction based on logistic regression will be

$$P_n = P_{\text{new}} = \frac{1}{1 + e^{-\langle x_n, \hat{b} \rangle}} \, , \tag{4.7}$$

which estimates the maximum likelihood probability that $y_n = 1$ if the data are generated by a distribution from a logistic model. We will call P_{new} a *direct* prediction to distinguish it from multiprobability predictions produced by Venn machines.

4.3.2.2 Logistic Venn Taxonomy

Now we can describe how logistic regression can be embedded in Venn machine as an underlying algorithm. As earlier, the aim is the predict labels y_i, which are equal to 0 for controls and 1 for cases, by objects x_i — vectors of features, which are intensities of the most frequent peaks in the logarithm scale.

The probabilistic method of logistic regression allows us to create a new type of taxonomy: the *logistic taxonomy* $\tau_i, i = 1, \dots, |\mathbf{Y}|$, which is defined as follows. The solution $\hat{b}$ of the optimisation problem (4.6) is calculated for the whole set $(x_1, y_1), \dots, (x_{n-1}, y_{n-1}), (x_n, y)$ as a training set and is used to make direct predictions $P_1, \dots, P_n$ on the same (training) examples. These predictions are not fair leave-one-out predictions, but it is correct to use them for taxonomy construction.

Let P'_i, $i = 1, \dots, n$ be direct predictions P_i sorted in ascending order. Set a number of taxonomy categories K. Let $L_0 = 0$ and $L_1, L_2, \dots, L_K$ be the integers closest to $n/K, 2n/K, 3n/K, \dots, n$. The category τ_i of the example (x_i, y_i) is then defined as the number of the interval formed by division points $P'_{L_0}, P'_{L_1}, \dots, P'_{L_K}$ where value P_i falls: $\tau_i = \{j = 1, \dots, K : P'_{j-1} < P_i \leq P'_j\}$.

Thus, we divide the examples into several groups of approximately equal size being grouped by the similarity of their direct predictions. The division is carried out by K-quantiles of a set of direct predictions. We construct categories of equal size because the small size of categories will result in overfitting and will be punished by the large diameter of a probability intervals. On the other hand, large categories will result in underfitting.

When the direct prediction is obtained, the solution of the optimisation problem $\hat{b} = \{\hat{b}_k\}$ can be interpreted as implicit peak ranking. Since the solution is used as weighting of peak intensities in the logistic regression model $\left\langle x_i, \hat{b} \right\rangle$, the larger the absolute value of a corresponding weight $\hat{b}_k$, the more important peak k is. This way Venn predictions can pinpoint peaks which could potentially reflect the absence or presence of the disease.

4.3.3 Time Dependency

Identification of diseases in their early stages can be crucial for successful treatment. For this reason, one of the important questions we would like to answer when analysing mass spectrometry data of diseased and healthy samples is the following: How long in advance of the moment of diagnosis can we provide reliable predictions? In this section we describe the methodology which could answer this question. The methodology was previously applied in [26]; we are going to use it in combination with the developed algorithms with online validity (category-based confidence machine based on linear rules and logistic Venn machine).

Since the samples were collected over a long period of time (up to five years) and we have the information regarding how long in advance of the moment of diagnosis these samples were taken, we can take advantage of this fact. We consider different time slots of a fixed width; for example, 6 or 12 months. The higher the number of available samples, the narrower window we can consider. The time slots finish $t = 0, 1, 2, \ldots$ months in advance of the moment of diagnosis. After fixing the time slot, we pick all the patients whose measurements were taken in this time slot, together with matched controls. The feature of the ovarian cancer data is that several measurements were taken from the same patient at different moments. If several measurements of the same patient from the ovarian cancer data fall in the time slot, we consider only the one closest to the moment of diagnosis, eliminating the others together with corresponding controls.

We then can apply designed algorithms with online validity to patient measurements in time slots moving away from the moment of the diagnosis and observe how validity and efficiency of designed algorithms change over time. We expect output predictions to maintain the property of validity (as long as the number of patients fallen in each time slot is large enough). However, the efficiency of predictions is expected to deteriorate as the time slot is moving away since we assume that, further from the moment of diagnosis, mass spectra contain less information useful for discrimination between cases and controls

4.3.4 Confidence Machines in the Triplet Setting

The UKCTOCS data sets were originally provided in a triplet setting: each case is accompanied by two controls. In this section we design confidence machines which can take the advantage of the triplet setting. The approach laid out here can be naturally generalised for the situation when each case is complement with a fixed number of matched controls. For example, we could consider quadruples out of which only one sample is diseased and the others are healthy. However, we will apply our approach only to a triplet setting because the UKCTOCS data were structured this way.

Each object in this setting is a triplet of mass spectra rather than a single spectrum. Mass spectra are arbitrarily ordered within objects-triplets. Each label is then the number of the sample in a triplet which is a case. There are therefore three possible labels: $\{1, 2, 3\}$. The proposed strangeness measure is also based on linear combinations of peak intensities and possibly CA125. This strangeness measure designed for a triplet setting will allow us to pinpoint informative mass spectrometry profile peaks, which could be potential biomarkers.

Let us assume we have objects $x_i = \{x_i^1, x_i^2, x_i^3\}$, $i = 1, \ldots, n$, where x_i^j, $i = 1, \ldots, n$, $j = 1, 2, 3$, are samples represented by mass spectra. Label $y_i \in \mathbf{Y} = \{1, 2, 3\}$, $i = 1, \ldots, n$, is a number of a case in a triplet x_i.

In order to design a strangeness measure, we consider linear rules of the following type

$$r(x_i^j; w_1, \ldots, w_h, n_1, \ldots, n_h) = \sum_{k=1}^{h} w_k \log I(n_k)\,, \tag{4.8}$$

where $I(n_k)$ is the intensity of peak n_k; $w_k \in \mathbb{R}$; $k = 1, \ldots, h$ are weights. These are the same rules as (4.1) only without a threshold. Such a rule identifies a diseased sample as the one with the highest value of the rule: $y_i(r) = \arg\max_j r(x_i^j)$.

The strangeness measure was calculated using the classification method described below. We fix the number h of peaks used in a rule. We then consider a set of possible linear rules of type (4.8) where parameters of the rules can have the following values: $w_i \in W_i \subseteq \mathbb{R}$, $\{n_1, \ldots, n_h\} \in \mathbb{P} \subseteq \{1, \ldots, N\}^h$.

Out of these rules we select the one with the maximum value of accuracy on the set of all objects-triplets including a new one with a new hypothetical label. We denote it by $\hat{r}$. We can then define the strangeness score for triplet x_i:

$$\alpha_i = -(\hat{r}(x_i^{\text{case}}) - \max(\hat{r}(x_i^{\text{control1}}), \hat{r}(x_i^{\text{control2}})))\ ,$$

where $x_i^{\text{case}} = x_i^{y_i}$ is a case in triplet x_i, and x_i^{control1} and x_i^{control2} are controls. Thus, the better the rule $\hat{r}$ performs on a triplet x_i, the less strange the triplet is. The rule was derived from the whole set of triplets; therefore, the strangeness measure defined above could reflect how strange a triplet is in relation to other samples.

The rule $\hat{r}$ which is chosen when strangeness scores are calculated is the best in terms of discriminating between healthy and diseased samples in a triplet setting. Hence the peaks that are included in $\hat{r}$ are the most informative ones and could be potential biomarkers. In addition, weights w_i, $i = 1, \ldots, h$ can reflect the importance of each peak (the larger the absolute value of a peak's weight, the more important the peak is) and whether the higher or lower protein level is a risk factor (depending on the sign of the corresponding weight).

The application of confidence machines in this setting seems to be more intuitive and more useful when the number of samples in a group is big. In this case we need to identify a diseased sample out of a group of matched patients and can state that some of them are more likely to be diseased than others.

Confidence machines in a triplet setting can also be useful for the following reason. When we are given a diseased patient, and she is accompanied by several controls, in reality we can never be sure that the controls are not in an early stage of the disease. Application of confidence machines in a triplet-like setting could help identify such situations. For example, if credibility or confidence is low, this may mean that some of controls could be potential cases.

4.4 Experimental Results

In this section, we present the results of application of designed methods to the UKCTOCS data sets. Results of application of the same algorithms to the Salmonella mass spectrometry data provided by VLA are shown in Appendix C.

Here we first demonstrate results of application of category-based confidence machines based on linear rules; then results on logistic Venn machines; and finally, confidence machines in a triplet setting. All experiments are run within the time dependency analysis framework.

It should be noted that, in Chapter 3, we applied designed algorithms to the whole UKCTOCS data sets, while in this chapter we apply algorithms developed for proteomic data only to subsets of the same data sets (which fall in time slots of 6 or 12 months). Therefore, the results of application of different algorithms to the UKCTOCS cannot be compared.

4.4.1 Category-Based Confidence Machines

In order to demonstrate how the proposed methodology works in practice, we applied the designed category-based confidence machine to MALDI-TOF mass spectrometry data: the ovarian cancer, breast cancer and heart disease data sets.

For the ovarian cancer data we are looking for mass spectrometry profile peaks that could in combination with the biomarker CA125 provide accurate predictions. Hence for ovarian cancer we consider the simplest possible combinations (4.1) of CA125 and one peak ($h = 2$); n_1 corresponds to CA125 level, n_2 is any peak, $w_1 \in W_1 = \{0, 0.5, 1, 2\}$ is a CA125 weight, $w_2 \in W_2 = \{-1, 0, 1\}$ is a peak weight. Because of the small number of samples, any additional terms in the rules (4.1) considered would bring overfitting.

For the breast cancer and heart disease data sets we did not have any known biomarkers and considered cut-off rules (4.1) with one peak involved ($h = 1$) with $w_1 \in W_1 = \{-1, 1\}$, a weight that determines whether the peak has higher or lower intensities for cases.

No normalisation/standardisation was applied; however, designed category-based confidence machines have embedded logarithm transformation.

We will first aim to provide predictions with confidence just before the moment of the diagnosis, and then we will investigate how long in advance of the moment of diagnosis we can provide reliable predictions.

4.4.1.1 Prediction before the Moment of Diagnosis

Thus, we now consider only patients whose measurements were taken not long in advance of the moment of diagnosis: not earlier than 6 months in advance for ovarian cancer patients and 12 months in advance for breast cancer patients. We are using these periods to be consistent with the triplet analysis of these data [10; 11; 15], where we considered time slots of the same width.

We accumulate all cases which were taken not earlier than 6 (12) months in advance, together with controls which match these cases (and hence have the same time to diagnosis values). In case of the ovarian cancer data, if several measurements of the same patients were taken in this time window, we considered only the latest one (i.e., with the minimum value of T), all the other case measurements and their matched controls were eliminated.

When analysing the heart disease data, we will consider all patients together regardless how early these measurements were taken for two reasons. First, for heart disease it is not essential to produce reliable predictions as early in advance as possible since once the disease is identified, a successful operation may be performed on a patient. Second, when we did analyse heart disease patients in shorter time slots, we could not achieve performance better than on the whole data set.

4.4.1.1.1 Region Predictions. At first, we will demonstrate how prediction with confidence works. The implemented category-based confidence machine provides two p-values for each patient: one for the 'healthy' diagnosis, another for the 'diseased' diagnosis. On the basis of these p-values, we calculate confidence and credibility for each patient as described in Section 2.1.3. After assigning each patient with two p-values, we make forced prediction, that is, predict the diagnosis with the highest p-value.

Table 4.1 represents p-values, confidence and credibility for ovarian cancer measurements taken not earlier than 6 months in advance of the moment of diagnosis; several examples are given for illustrative purposes.

Recall that p-values reflect how well the hypothetical label conforms with the rest of the sequence; therefore, confidence is close enough to 1 and credibility is not close to 0, the prediction is considered to be reliable. We will demonstrate this in detail on several examples from Table 4.1. For instance, measurement 141100 in Table 4.1 has two p-values one of which is high (0.99), another — low (0.01). This results in high confidence of 0.99 and high credibil-

Table 4.1: Examples of the output of category-based confidence machines applied to the ovarian cancer data in a 0–6 month time slot: true and predicted diagnoses, p-values for both diagnoses (0 — healthy, 1 — diseased), confidence and credibility for several patients

Measurement ID	True diagnosis	Predicted diagnosis	p-value for 0	p-value for 1	Confidence	Credibility
141100	0	0	0.99	0.01	0.99	0.99
146384	0	1	0.12	0.13	0.88	0.13
232604	1	0	0.51	0.28	0.72	0.51
245401	1	1	0.01	0.97	0.99	0.97

ity of 0.99 and identifies the prediction as reliable: only one diagnosis conforms well with the rest of the set. If this patient were a case (diagnosis value of 1) and the underlying model were i.i.d., this would mean that an event of probability $\leq 1\%$ occurred. Here by 'probability' we mean any probability distribution $\mathcal{P}$ on $\mathbf{Z}^\infty$ which is i.i.d. This is the consequence of confidence machine p-value property:

$$\mathcal{P}(\text{p-value} \leq 0.01) \leq 0.01$$

for any i.i.d. probability distribution $\mathcal{P}$ on $\mathbf{Z}^\infty$. Thus, this statement is not related to the estimation of the error on the whole population: we just calculate the probability of the event that *individual* p-value for a particular object and a particular label does not exceed 0.01, and this probability is calculated according to any i.i.d. distribution on $\mathbf{Z}^\infty$.

For this reason, we expect the patient to be healthy, which she is. In contrast, measurement 232604 has p-values, neither of which is close to zero. These p-values do not produce high confidence (0.72) or high credibility (0.51), which means that neither of the diagnoses is likely to be correct and, hence, there is not enough information to confidently classify the patient. As a result, the output prediction for measurement 232604 is indeed incorrect.

We further observe the accuracy of forced predictions. In this section, we would like to demonstrate the advantage of category-based confidence machines as region predictors: their output is always conditionally valid. We implemented region predictors, which, for each significance level $\epsilon > 0$, predict a set of diagnoses with p-values greater than ϵ. Significance levels ϵ we considered are 1%, 5%, 10% and 20%.

The category-based confidence machines implemented in our work proved to be conditionally valid for all data sets, that is, the rate of erroneous predictions among healthy patients and among diseased patients does not exceed the significance level in a leave-one-out procedure. (Recall that conditional validity is proved under i.i.d. assumption only for the online mode.) The property of conditional validity means that we could guarantee a certain level not only for accuracy, but also for accuracy within healthy and diseased samples. This is demonstrated in Table 4.2, which shows the error rate for the ovarian cancer data in the time slot of 0–6 months for different significance levels.

We could also consider different significance levels for classes of healthy and diseased patients. This would lead to different guaranteed values of regional sensitivity or specificity if we need to put one of these characteristics first.

Table 4.2: Validity and efficiency of category-based confidence machines applied in the offline mode to the ovarian cancer data in months 0–6: accuracy, regional sensitivity and regional specificity are not less than the preset confidence level $(1 - \epsilon)$; rates of empty, certain and multiple predictions reflect the efficiency of the region predictor

Confidence level $(1 - \epsilon)$	99%	95%	90%	80%
Regional accuracy	99.5%	95.6%	90.7%	80.4%
Regional sensitivity	100.0%	95.6%	91.2%	80.9%
Regional specificity	99.3%	95.6%	90.4%	80.2%
Empty predictions	0.0%	0.0%	3.9%	18.1%
Certain predictions	25.5%	91.7%	94.1%	81.4%
Multiple predictions	74.5%	8.3%	2.0%	0.5%

To demonstrate the property of conditional validity in dynamics, we ran the analogous experiments in the online mode: considering one patient after another as a test set and adding all processed patients to a training set. Figure 5.1 shows the erroneous prediction dynamics for the ovarian cancer data in the time slot of 0–6 months processed in the online mode for different significance levels. The horizontal axis represents the number of patients, the vertical axis is the number of erroneous region predictions. The figure demonstrates conditional validity of implemented predictors: solid lines, which represent the actual number of errors, correspond to dotted lines, which demonstrate the expected number of errors (equal to a significance level multiplied by the number of patients).

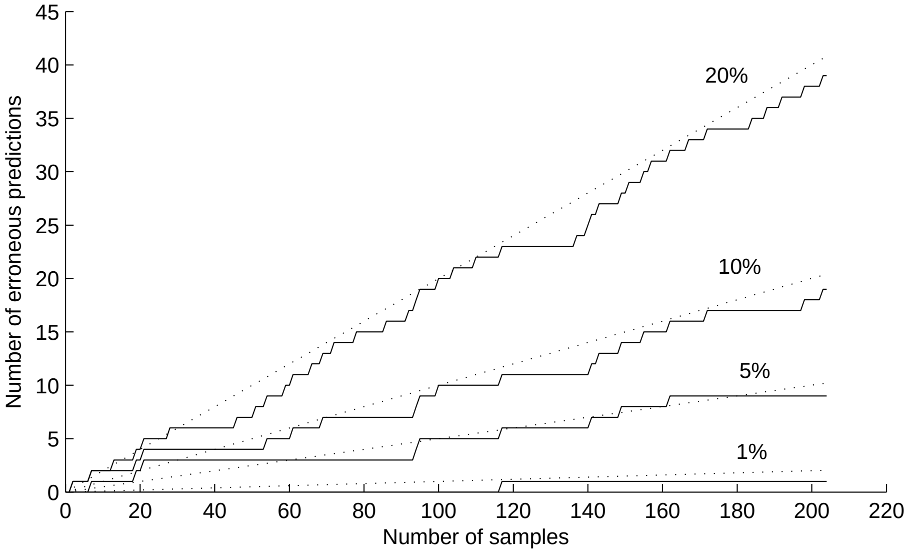


Figure 4.2: Validity dynamics in the online mode for the ovarian cancer data in the time slot of 0–6 months

Having the guaranteed error rate, we may still obtain several diagnoses as a prediction. This is shown in the bottom half of Table 4.2. The table provides the number of multiple predictions, certain predictions and empty predictions for the ovarian cancer data in the time slot of 0–6 months in advance. These characteristics describe efficiency of the predictor. As we can see from Table 4.2, the higher the confidence level, the more multiple predictions appear in experiments to maintain validity. It is also shown in the table that, at confidence levels of 95% and 90%, more than 90% of all region predictions comprise exactly one label, that is, are similar to the output of simple predictors.

4.4.1.1.2 Accuracy of Forced Predictions. So far we output region predictions by presetting the error rate we would like to obtain. Now we change the significance level for each patient so that each prediction is singleton and examine the accuracy of forced predictions.

We can compare the accuracy of forced predictions with the accuracy of the underlying algorithm, which we will refer to as *bare threshold predictions*. The underlying algorithm is the following. We consider the same sets of cut-off rules (4.1) we considered in category-based confidence machines. Such rules classify a patient as diseased if the *true* value is returned, healthy otherwise. Out of the set of rules we also select the one (4.3) which maximises the minimum value of sensitivity and specificity on the training set. The selected rule is then applied to the test set (a left-out sample).

Table 4.3 demonstrates the accuracy of forced predictions output by category-based confidence machines and bare threshold predictions for the three data sets. It can be seen from the table that forced prediction accuracy of category-based confidence machines is comparable to the accuracy of its underlying algorithms for all data sets.

The highest accuracy was obtained for the ovarian cancer data: 92.2%. The breast cancer data resulted in the accuracy of 71.9%. The accuracy of the application of category-based confidence machines to the heart disease data is much lower (59.7%). Nevertheless, this accuracy is comparable with the accuracy of its underlying algorithm — 59.4%. One could speculate that the accuracy of heart disease experiments is not high enough, because we consider all measurements in the data set taken as early as three years before the diagnosis. But as it was mentioned before, when we analysed heart disease patients in shorter time slots, we could not make highly accurate predictions either.

Thus, designed predictors, when forced to make singleton predictions, result in accuracy similar to the accuracy of their underlying algorithms, but, when interpreted as region predictions, they can guarantee the preset error rate asymptotically.

4.4.1.2 Prediction in Advance of the Moment of Diagnosis

Here we attempt to answer the second question stated at the beginning of this chapter: How long in advance of the moment of diagnosis can we make reli-

Table 4.3: Forced point predictions output by category-based confidence machine and bare predictions output by the threshold method for measurements taken not long in advance of the moment of diagnosis

	Forced predictions			Bare threshold predictions		
Data set	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
Ovarian cancer	92.2%	91.2%	92.7%	92.7%	89.7%	94.1%
Breast cancer	71.9%	73.7%	71.1%	71.9%	73.7%	71.1%
Heart disease	59.7%	59.9%	59.6%	59.4%	59.4%	59.4%

able predictions? It was shown in [10; 15] and in Appendix B that, for ovarian and breast cancers, there are certain time slots when mass spectrometry profile peaks carry statistically significant information for discrimination between controls and cases, i.e., we can reject the null hypothesis that the diagnosis is independent of the information contained in peak intensities at significance level of 5% well in advance of the moment of diagnosis. In this section, we will be able to confirm this tendency and also to check whether measurements from different triplets are comparable. In the triplet analysis, we assumed that we can compare only measurement which were combined in triplets, that is, were matched by the location and time of sample collection. Here we are merging all samples together and check whether we can compare them with each other and output highly accurate predictions.

In order to do this, we will investigate dynamics of output predictions over time using the time dependency analysis laid out in Section 4.3.3. We will consider different time slots of the fixed length (6 months for ovarian cancer and 12 months for breast cancer) shifting away from the moment of diagnosis. These time slots finish 1, 2, 3, ... months in advance of the moment of the diagnosis.

After fixing the time slot, we pick all the patients whose measurements were taken in this time slot together with matched controls. For the ovarian cancer data, if several measurements of the same patients fall in this time slot, we consider only the one closest to the moment of the diagnosis, eliminating the others together with corresponding controls. We then apply designed category-

Table 4.4: The rate of certain predictions output by category-based confidence machines at significance levels $\epsilon = 5\%$, 10%, 20% in different time slots for the ovarian cancer and breast cancer data sets

Time slot	Ovarian cancer			Breast cancer		
end	$\epsilon = 5\%$	$\epsilon = 10\%$	$\epsilon = 20\%$	$\epsilon = 5\%$	$\epsilon = 10\%$	$\epsilon = 20\%$
0	91.7%	94.1%	81.4%	5.3%	15.8%	50.9%
1	78.6%	94.6%	84.5%	8.3%	23.6%	62.5%
2	58.2%	80.1%	90.1%	7.7%	24.4%	76.9%
3	25.9%	48.2%	88.9%	7.7%	24.4%	76.9%
4	27.2%	56.8%	91.4%	8.3%	19.4%	93.1%
5	21.7%	44.9%	71.0%	8.3%	29.2%	88.9%
6	18.3%	41.7%	58.3%	10.0%	36.7%	90.0%
7	9.8%	21.6%	62.8%	5.3%	14.0%	59.7%
8	2.0%	29.4%	60.8%	3.9%	17.7%	62.8%
9	18.3%	33.3%	73.3%	1.9%	14.8%	64.8%
10	11.9%	29.8%	67.9%	1.9%	14.8%	35.2%
11	9.5%	22.6%	58.3%	1.9%	16.7%	50.0%

based confidence machines to patient measurements in time slots moving away from the moment of the diagnosis.

4.4.1.2.1 Efficiency Dynamics. All region predictions output by designed category-based confidence machines proved to be valid in all time slots: the number of erroneous predictions within healthy patients and diseased patients corresponded to the preset value up to statistical fluctuations.

We therefore need to investigate the dynamics of region prediction efficiency. Table 4.4 shows the ratio of certain predictions output by category-based confidence machine in different time slots at significance levels of $\epsilon = 5\%$, 10%, 20% for the ovarian cancer and breast cancer data sets. The table demonstrates that for the ovarian cancer data efficiency deteriorates when we move away from the moment of diagnosis: the number of certain predictions decreases. As for the breast cancer data, efficiency improves first, achieves the maximum of certain prediction rate in months 4–6 and then also deteriorates.

Table 4.5: Accuracy dynamics of forced point predictions output by category-based confidence machine and bare predictions output by the threshold method on the ovarian cancer data set

		Forced predictions			Bare threshold predictions		
Time slot	No of samples	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
0–6	204	92.2%	91.2%	92.7%	92.7%	89.7%	94.1%
1–7	168	89.9%	89.3%	90.2%	89.3%	89.3%	89.3%
2–8	141	83.7%	83.0%	84.0%	85.1%	80.9%	87.2%
3–9	108	78.7%	80.6%	77.8%	76.9%	80.6%	75.0%
4–10	81	79.0%	74.1%	81.5%	84.0%	85.2%	83.3%
5–11	69	73.9%	73.9%	73.9%	76.8%	73.9%	78.3%
6–12	60	66.7%	65.0%	67.5%	65.0%	65.0%	65.0%
7–13	51	68.6%	64.7%	70.6%	68.6%	82.4%	61.8%
8–14	51	66.7%	70.6%	64.7%	72.6%	64.7%	76.5%
9–15	60	73.3%	75.0%	72.5%	73.3%	70.0%	75.0%
10–16	84	70.2%	71.4%	69.6%	71.4%	71.4%	71.4%
11–17	84	66.7%	67.9%	66.1%	66.7%	64.3%	67.9%

4.4.1.2.2 Accuracy of Forced Predictions over Time. When applied in different time slots, category-based confidence machines can also be forced to output singleton predictions and compared with the corresponding bare threshold prediction.

Tables 4.5 and 4.6 demonstrate the accuracy of forced predictions and bare predictions in the moving time window. It can be seen from the tables that forced prediction accuracy of category-based confidence machines is again comparable to the accuracy of its underlying algorithms. At the same time, in their natural setting category-based confidence machines can output region predictions with the preset regional sensitivity and specificity.

On the whole, the tables demonstrate that forced predictions produced by category-based confidence machines are reasonably accurate well in advance of the moment of the moment of diagnosis. For example, the accuracy on the ovarian cancer data set in the time slot 10–16 (the latest time slot when CA125 on its own does not carry statistically significant information for disease discrimination, see Appendix B and [57]) is 70.2%. This is quite good given

Table 4.6: Accuracy dynamics of forced point predictions output by category-based confidence machines and bare predictions output by the threshold method on the breast cancer data set

		Forced predictions			Bare threshold predictions		
Time slot	No of samples	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
0 – 12	57	71.9%	73.7%	71.1%	71.9%	73.7%	71.1%
1 – 13	72	77.8%	79.2%	77.1%	77.8%	79.2%	77.1%
2 – 14	78	76.9%	76.9%	76.9%	76.9%	80.8%	75.0%
3 – 15	78	76.9%	76.9%	76.9%	76.9%	80.8%	75.0%
4 – 16	72	77.8%	79.2%	77.1%	75.0%	75.0%	75.0%
5 – 17	72	75.0%	75.0%	75.0%	75.0%	75.0%	75.0%
6 – 18	60	73.3%	75.0%	72.5%	76.7%	75.0%	77.5%
7 – 19	57	71.9%	73.7%	71.1%	68.4%	68.4%	68.4%
8 – 20	51	70.6%	70.6%	70.6%	70.6%	70.6%	70.6%
9 – 21	54	70.4%	72.2%	69.4%	70.4%	72.2%	69.4%
10 – 22	54	48.2%	55.6%	44.4%	72.2%	66.7%	75.0%
11 – 23	54	70.4%	72.2%	69.4%	68.5%	72.2%	66.7%

that the diagnosis is made not earlier than 10 months in advance before the diagnosis. For comparison, when we make predictions with the same method for measurements taken just before the moment of diagnosis (in a 0–6 time slot), the accuracy is equal to 92.2%. As we move away from the moment of diagnosis, accuracy of predictions decreases. Low accuracy 6, 7 and 8 months in advance may be explained by a small number of samples in this period (below 70 samples for any time slot). In general, category-based confidence machines produce predictions with accuracy higher than 66% up to 11 months in advance of the moment of ovarian cancer diagnosis.

Similarly, category-based confidence machines achieve accuracy higher than 70% for the breast cancer data up to 9 months in advance of the moment of diagnosis. However, there is no apparent decreasing trend in accuracy; it fluctuates in the range of 70.4–77.8%.

4.4.1.2.3 Prediction Dynamics over Time. As mentioned before, the feature of the ovarian cancer data set is that ovarian cancer cases can have several measurements taken at different moments. For this reason, we can observe the change in the output of category-based confidence machines for this data set. As an illustration, we will consider several ovarian cancer cases that have measurements taken over a long period of time and will show how confidence and credibility are changing when the patient is approaching the moment of diagnosis.

We select patients with at least three measurements. For each measurement, we train the category-based confidence machine on the samples in the earliest 6-month time slot containing the measurement leaving out the measurement itself. For example, if a measurement was taken 6.5 months in advance, we consider the time slot from month 12 to month 6. If an ovarian cancer case has several measurements fallen in this time slots, all except for the one closest to the moment of diagnosis were eliminated from the training set. We then apply the category-based confidence machine to the left-out measurement and output a forced prediction, its confidence and credibility. All other parameters were the same as in previous experiments.

Dynamics of confidence and credibility for measurements of several patients is shown in Table 4.7.

Table 4.7: Dynamics of confidence and credibility for measurements taken from two ovarian cancer cases

Case ID	Months in advance	Prediction	Confidence	Credibility
39	10	1	89.5%	67.9%
	4	1	90.9%	44.4%
	2	1	99.0%	66.0%
	1	1	99.1%	76.8%
42	24	1	69.0%	71.4%
	15	0	45.0%	78.1%
	3	1	98.6%	100.0%

Recall that we can trust a prediction if its confidence is close to 1 (that is, all p-values for alternative diagnoses are close to 0) and its credibility is not close to 0 (that is, the maximum p-value is not close to 0). This implies that if a category-based confidence machine makes correct predictions about the case, we expect confidence to be approaching 100% when measurements are getting closer to the moment of diagnosis. Meanwhile, credibility is expected not to be getting close to 0%. Table 4.7 demonstrates that patient 39 confirms our expectations.

Patient 42 represents a more interesting example: we make an erroneous prediction 15 months in advance. However, its confidence is not close to 100%, which reflects that we cannot be sure in this prediction. When we make a final prediction for this patient 3 months in advance, both confidence and credibility are close to 100%.

4.4.1.3 Summary

When applied to the UKCTOCS data sets, designed category-based confidence machines prove to be conditionally valid: the number of erroneous prediction within healthy and diseased patients corresponds to the preset significance level.

Moreover, when making predictions for the ovarian cancer data just before the moment of diagnosis, more than 90% of output region predictions contain exactly one label at confidence levels of 95% and 90%. This makes them similar to simple predictors; however, category-based confidence machines can also guarantee the error rate within groups of healthy and diseased patients.

Even though category-based confidence machines produce region predictions, their output can be interpreted as singleton predictions. It was demonstrated that when forced to make single predictions, our methodology provides accuracy close to the accuracy of its underlying algorithm (threshold rule).

As a result, accuracy on the ovarian cancer data rises from 70.2% 10 months in advance of the moment of diagnosis to up to 92.2% just before the moment of diagnosis. When applied to the breast cancer data, the methods allowed us to achieve accuracy of 70.4–77.8% for up to 9 months in advance of diagnosis. The same approach has been applies to the heart disease data without time dependency although the achieved accuracy was not high.

4.4.2 Logistic Venn Machines

In this section, we apply logistic Venn machines to the UKCTOCS data sets: ovarian cancer, breast cancer and heart disease. Again, for ovarian cancer and breast cancer we consider different time slots of the fixed length (6 months for ovarian cancer and 12 months for breast cancer) shifting away from the moment of diagnosis. After fixing the time slot, we select all the patients whose measurements were taken in this time slot together with matched controls. We then apply logistic Venn machine to selected patient measurements.

For the heart disease data, we analyse all samples without considering time slots for the same reasons as for category-based confidence machines: there was no incentive from the practical point of view and there were no better results achieved on the data in time slots.

In all experiments, we use leave-one-out mode: each example (x_i, y_i) is considered as if it were a new test example and all the remaining examples in the data are treated as the training set. We applied a logistic Venn taxonomy with 5 categories to avoid a small number of categories and a small number of samples in each category. Before logistic Venn machine is launched, logarithm transformation is applied to the data. No standardisation was applied. Hence each object x_i is a vector comprising the following features: intensities of the most frequent peaks on the logarithmic scale, value '1' for possible absolute term in logistic regression model and logarithm of the CA125 value for the ovarian cancer data set.

4.4.2.1 Multiprobability Predictions

Since the underlying algorithm — logistic regression — also produces probability distributions, we can compare the results of the application of the Venn machine based on logistic regression and the probabilistic predictor of logistic regression itself.

Results of experiments for several controls and cases of the heart disease data are shown in Table C.5 for illustrative purposes. For each example, the table contains the true label $y = y_{\text{new}}$, and a Venn prediction — the interval $[P^-_{\text{new}}, P^+_{\text{new}}]$ of probability that $y = 1$. To avoid confusion, we would like

Table 4.8: Leave-one-out Venn predictions for the heart disease data

No.	True label	Venn prediction	Direct prediction
1	0	0.313–0.321	0.508
2	1	0.616–0.616	0.689
3	0	0.321–0.330	0.510
4	0	0.143–0.259	0.371
5	0	0.616–0.634	0.622
6	0	0.321–0.321	0.484
7	1	0.313–0.321	0.516
8	0	0.616–0.634	0.558
9	0	0.143–0.161	0.333
10	1	0.616–0.625	0.703

to point out that in Chapter 3, where some of the classes are multilabel, we used intervals (2.10) of probability that the predicted label is correct. They are different from the ones we show in this chapter: the latter are probability intervals $[P^-_{\text{new}}, P^+_{\text{new}}]$ (2.11) for label 1, regardless of whether it is correct or not. We can use these intervals since all UKCTOCS data sets comprise two classes.

Let us look at some of the example in Table C.5. One can see that logistic Venn machine outputs prediction intervals [0.321, 0.508] and [0.616, 0.689] for probabilities that examples 1 and 2 are cases ($y = 1$). As prediction intervals indicate, the correct labels for example 1 and 2 are 0 and 1, respectively. The table also includes predictions P_{new} output by logistic regression for each example. Recall that we refer to these predictions as *direct* predictions as opposed to *Venn* predictions output by Venn machines. The table demonstrates that both direct and Venn predictions can be correct or erroneous. The performance of such predictions will be analysed in Section 4.4.2.3.

First of all, we attempt to demonstrate implications of logistic Venn machine validity. We aim to show that true probabilities of label distribution are covered or almost covered by the interval between lower Venn prediction and upper Venn prediction. Since we do not know true probabilities of label distribution, we compare empirical probabilities, that is, mean true labels, with mean direct and Venn predictions.

Figure 4.3 is a graphical representation of corresponding cumulative results. The horizontal axis shows the number of observed examples. The vertical axis shows the cumulative values of: (1) true labels y_{new} (a solid line); (2) lower and upper Venn predictions $P^-_{\text{new}}, P^+_{\text{new}}$ (two dot-dashed lines) and (3) cumulative direct predictions P_{new} (a dashed line). The examples are sorted according to direct predictions.

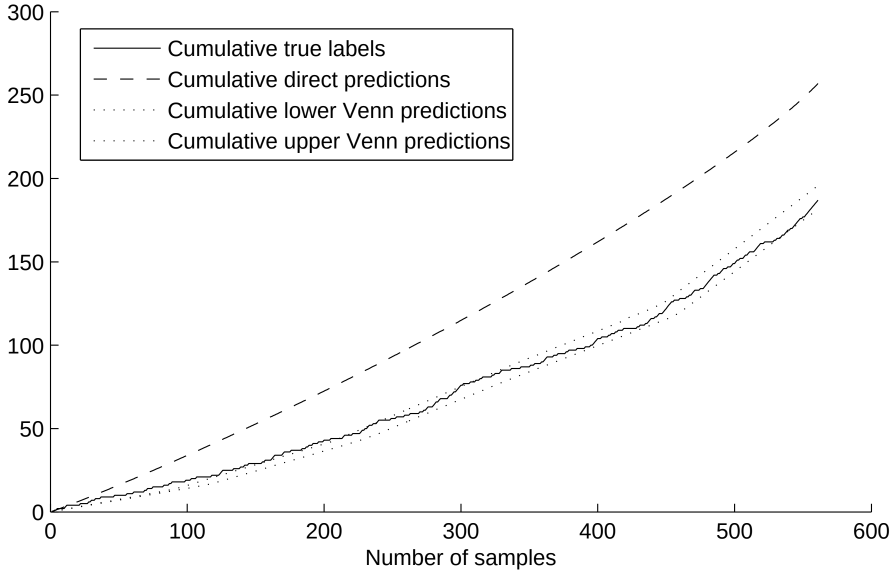


Figure 4.3: Cumulative Venn and direct predictions for the heart disease data (all samples)

Firstly, the plot conforms with validity of Venn machine outputs. Secondly, we can see that probability intervals output by Venn machines are narrow (0.025 on average for the heart disease data); therefore, they are almost as precise as single probabilities.

Finally, Figure 4.3 demonstrates that probability intervals can be more accurate than single probabilities produced by logistic regression. It can be seen from the figure that the true labels are very different from the direct predictions but are only slightly above the upper Venn prediction up to approximately 210 examples and within the upper and lower Venn predictions for the remaining examples after this point. Thus, direct predictions can be misleading, while Venn predictions always cover or almost cover true labels.

Similar plots for the ovarian cancer and breast cancer data sets can be found in Appendix A (Figures A.1 and A.2). They also confirm the property of validity of Venn machines: the line corresponding to cumulative true labels is covered or almost covered by the space between lines for cumulative Venn predictions. For ovarian cancer the area between cumulative Venn prediction lines is also narrow (with average probability interval width of 0.026) and almost covers the cumulative true label line, while the line representing cumulative direct predictions diverges from the true label line for up to 180 first examples. For breast cancer, the average interval width is much larger (0.332), hence, probability intervals are not as precise and informative.

It can be said that both algorithms relied on the assumption of the mechanism generating the data — logistic regression statistical model. However, probability predictions used this mechanism directly, and Venn machines deployed the mechanism when defining the taxonomy. As a result, since the statistical model does not hold true (the opposite can be guaranteed only for artificially generated data), probabilities output by logistic regression are different from empirical probabilities. In contrast, Venn machine's validity was not affected by the fact that the model is not correct. Hence, Venn machine predictions appeared to be more accurate than singleton probability predictions.

4.4.2.2 Prediction Dynamics over Time

Now, for illustrative purposes, we will consider several ovarian cancer cases that have measurements taken over a long period of time and will show how probability intervals output by Venn machines are changing when the patient is approaching the moment of diagnosis.

The setting is the same as with designed category-based confidence machines. We select patients with at least three measurements. For each measurement, we train the machine on the samples in the earliest 6-month time slot containing the measurement leaving out the measurement itself. If an ovarian cancer case has several measurements fallen in this time slots, all except for the one closest to the moment of diagnosis were eliminated from the training set. We then apply the machine to the left-out measurement and output a Venn prediction.

Table 4.9: Dynamics of prediction intervals output by Venn machines for measurements taken from the same ovarian cancer case

Personal ID	Months in advance	Prediction interval
29	13	0.22–0.39
	10	0.59–0.71
	4	0.88–0.94
39	10	0.53–0.71
	4	0.44–0.94
	2	0.96–1.00
	1	0.97–1.00

Table 4.9 shows the dynamics of prediction intervals output by Venn machines for samples 29 and 39. Each row corresponds to a single measurement. Column 2 demonstrates how early in advance this measurement was taken. These samples with multiple measurements illustrate two trends in probability interval change. First, the interval is getting narrower when the moment of diagnosis is approaching, which means that two probability distributions produced by Venn machines are getting closer to each other, and as a result, the overall prediction is getting more precise. Second, the interval is moving towards 1 (the prediction that the sample is diseased). This implies that we have more trust in our prediction and the prediction is indeed correct (because all the samples considered were cases).

4.4.2.3 Accuracy of Forced Predictions

It was shown in [10; 15] that in a triplet setting for ovarian cancer and breast cancer, there are certain time slots when mass spectrometry profile peaks carry statistically significant information for discrimination between controls and cases. Here we would like to check that samples from different triplets can be compared to each other and be merged while providing high accuracy of prediction.

Even though Venn machines and logistic regression produce multiprobability and probability predictions, respectively, their outputs can be interpreted as singleton predictions (forced predictions). In this section we examine the accuracy of forced predictions when we are not able to use advantages of multiprobability predictions.

We can extract forced predictions out of Venn machines and logistic regression the most intuitive way: we classify a new sample as 1 (case) if and only if $P_{\text{new}} > 0.5$ for direct prediction or $P^{+}_{\text{new}} + P^{-}_{\text{new}} > 1$ for Venn prediction. This will also allow us to compare accuracy of Venn predictions with direct predictions.

For the ovarian and breast cancer data sets, we again consider the dynamics of predictive ability of mass spectrometry profile peaks across the timeline. Table 4.10 allows us to compare accuracy of forced predictions produced on the ovarian cancer data by logistic Venn machine and its underlying algorithm, logistic regression, in different time slots. The table demonstrates that Venn machines are comparable with logistic regression in terms of forced prediction accuracy: in time slots close to the moment of diagnosis Venn machine is slightly outperformed by logistic regression, then in months 5–7 they have equal accuracy, and in months 8–11 (time slots we are mostly interested in) Venn machine beats logistic regression.

Table 4.10 demonstrates that forced predictions produced by Venn predictions are reasonably accurate well in advance of the moment of diagnosis. For example, the accuracy on the ovarian cancer data set just before the moment of diagnosis (a 0–6 time slot)is 90.2% whereas the accuracy in the time slot 10–16 (the latest time slot when CA125 on its own does not carry statistically significant information for disease discrimination) is 73.8%. In general, Venn machines produce predictions with accuracy higher than 73% up to 10 months in advance of the moment of diagnosis.

Venn machines do not produce high accuracy on the breast cancer data. We can speculate that this fact can be explained the following way: we showed earlier [10] and will show in this thesis that the breast cancer data set contains only one peak which carries statistically significant information (peak 19). When we consider more features, we include more peaks that carry more noise,

Table 4.10: Dynamics of Venn machine and logistic regression performance on the ovarian cancer data set

	Venn machine			Logistic regression		
Time slot	Accuracy	Sensiti- vity	Specifi- city	Accuracy vity	Sensiti- city	Specifi-
0–6	90.2%	95.6%	87.5%	93.6%	85.3%	97.8%
1–7	88.1%	91.1%	86.6%	92.9%	83.9%	97.3%
2–8	76.6%	59.6%	85.1%	87.9%	78.7%	92.6%
3–9	83.3%	58.3%	95.8%	83.3%	69.4%	90.3%
4–10	75.3%	59.3%	83.3%	82.7%	66.7%	90.7%
5–11	79.7%	52.2%	93.5%	79.7%	56.5%	91.3%
6–12	81.7%	55.0%	95.0%	81.7%	55.0%	95.0%
7–13	70.6%	35.3%	88.2%	70.6%	35.3%	88.2%
8–14	82.4%	52.9%	97.1%	78.4%	47.1%	94.1%
9–15	75.0%	45.0%	90.0%	71.7%	35.0%	90.0%
10–16	73.8%	67.9%	76.8%	67.9%	25.0%	89.3%
11–17	66.7%	50.0%	75.0%	64.3%	17.9%	87.5%

which results in poor performance of Venn machines. The early diagnosis of breast cancer also gets more complicated because of the following feature of the breast cancer data set: all breast cancer samples were taken at least three months in advance of the moment of diagnosis.

Finally, for heart disease we consider the whole data set rather than dynamics across the timeline since it is sufficient to predict this disease at any moment to prevent the consequences. The accuracy of the application of Venn machines to the heart disease data is 69.9%. The accuracy is again comparable with the accuracy of underlying algorithms: 67.9%.

Thus, designed Venn machines when forced to make one single prediction result in accuracy similar to the accuracy of their underlying algorithm. Meanwhile, when interpreted as multiprobability predictions, Venn machines have the theoretically guaranteed property of validity.

4.4.2.4 Summary

The experiments showed that probability intervals constructed for ovarian cancer and heart disease are narrow, that is, the output of the multiprobability predictor is similar to a single probability distribution. Meanwhile, plots confirmed that property of validity holds true even in the offline mode. In addition, probability intervals produced for ovarian cancer and heart disease were more accurate than the output of corresponding probability predictor of logistic regression: probability predictions can be misleading while multiprobability predictor outputs a narrow probability interval which almost always covers the empirical label probability.

When Venn machines are forced to make point predictions, the accuracy of such predictions is comparable with the accuracy of the underlying algorithm of logistic regression. As a result, the accuracy of the proposed method on the ovarian cancer data rises from 73.8% 10 months in advance of the moment of diagnosis to up to 90.2% before the moment of diagnosis. However, Venn machines do not produce high accuracy on the breast cancer or heart disease data. On the heart disease data taken as a whole data set, we achieved forced accuracy of 69.9%.

4.4.3 Confidence Machines in a Triplet Setting

In this section, we apply confidence machines in a triplet setting to the UKCTOCS ovarian cancer and breast cancer data sets. We consider the same time slots — 6 months for ovarian cancer and 12 months for breast cancer — in order to be consistent with the previous analysis. However, this time slot width leads to a small number of triplets falling in these windows as the number of objects gets three times as smaller. We mostly consider this launch of experiments as a trial since the number of triplets is not large enough to produce reliable predictions. The larger number of patients per group would also make the outcome of confidence predictor more useful.

For each time slot, we launch a confidence machine in a triplet setting on the set of triplets within this time window. We then calculate two characteristics: forced accuracy and mean confidence. Forced accuracy describes performance of a confidence machine as a simple predictor, and mean confidence is a characteristic of region predictions. These values are summarised for different time slots of the ovarian cancer and breast cancer data sets in Tables 4.12 and 4.13, respectively.

No normalisation/standardisation was applied; however, designed confidence machines have embedded logarithm transformation.

4.4.3.1 Predictions for Individual Triplets over Time

First, we will illustrate how predictions with assigned confidence work in a triplet setting and how this confidence changes over time for measurements taken from the same sample. Several examples of p-values, confidence and credibility are provided in Table 4.11. The table represents predictions for UKCTOCS OC triplets with a case patient whose ID can be found in the first column; the prediction is made on the basis of the training set within the six-month time slot whose end is shown in the second column of the table.

Table 4.11: Confidence machines in a triplet setting: dynamics of confidence and credibility for triplets with measurements taken for the same ovarian cancer case

Case	Months	Prediction		p-values for		Confi-	Credi-
ID	in advance		1	2	3	dence	bility
39	10	1	0.82	0.21	0.11	0.79	0.82
	4	2	0.70	0.74	0.04	0.30	0.74
	2	1	0.85	0.02	0.06	0.98	0.85
	1	1	0.71	0.02	0.05	0.98	0.71
42	24	1	0.50	0.29	0.43	0.71	0.50
	15	3	0.10	0.10	0.65	0.90	0.10
	3	1	1.00	0.47	0.03	0.53	1.00

All samples are sorted within triplets so that the first sample in a triplet is a case; therefore, the correct label for each triplet (which is a case number) equals 1. Columns 'p-values' provide p-values for three possible labels (hypothetical case numbers). Then follow the columns with confidence and credibility calculated the way it was described in 2.1.2. Column 'Prediction'

Table 4.12: Confidence machines in a triplet setting applied to the ovarian cancer data

Time slot	No of triplets	Accuracy	Mean confidence
0–6	68	97.1%	0.93
1–7	56	92.9%	0.92
2–8	47	89.4%	0.89
3–9	36	80.6%	0.82
4–10	27	77.8%	0.80
5–11	23	73.9%	0.73
6–12	20	75.0%	0.72
7–13	17	76.5%	0.71
8–14	17	82.4%	0.72
9–15	20	80.0%	0.68
10–16	28	64.3%	0.63
11–17	28	50.0%	0.56
12–18	28	42.9%	0.55
13–19	30	43.3%	0.58
14–20	25	40.0%	0.53
15–21	20	40.0%	0.55
16–22	10	50.0%	0.58

represents the predicted number of a case in a triplet, which was determined as the one with the highest p-value.

Table 4.11 also provides examples of triplets with case measurements taken from the same ovarian cancer samples: cases 39 and 42 (dynamics for measurements of the same cases was provided in Table 4.7 when category-based confidence machines based on linear rules were applied). As before, for each measurement, we considered the earliest 6-month time slot containing the measurement and then used all measurements taken in this time slot as a training set.

For example, one can see from Table 4.11 that for triplets with case 39 we can achieve high confidence only 1 and 2 months in advance, whereas 4 months in advance we are making an incorrect prediction. For a triplet taken 4 months in advance, label 3 is rejected because it has a small p-value of 0.04; however, p-values for labels 1 and 2 are close to each other (0.70 and 0.74, respectively), which results in low confidence (0.30) of the prediction. And the prediction is indeed incorrect.

4.4.3.2 Forced Accuracy

Table 4.12 demonstrates than when making predictions for ovarian cancer just before the moment of diagnosis, we can output predictions with accuracy of 97.1% and most predictions are highly confident: mean confidence equals 0.93. When we move away from the moment of diagnosis, both accuracy and mean confidence deteriorate. However, we can still make predictions with accuracy of at least 73.9% as early as 9 months in advance.

Table 4.13: Confidence machines in a triplet setting applied to the breast cancer data

Time slot	No of triplets	Accuracy	Mean confidence
0–12	19	63.2%	0.55
1–13	24	83.3%	0.65
2–14	26	80.8%	0.63
3–15	26	80.8%	0.63
4–16	24	83.3%	0.63
5–17	24	79.2%	0.64
6–18	20	35.0%	0.50
7–19	19	47.4%	0.45
8–20	17	47.1%	0.44
9–21	18	44.4%	0.42
10–22	18	44.4%	0.39
11–23	18	33.3%	0.41
12–24	20	35.0%	0.39
13–25	17	47.1%	0.44
14–26	18	55.6%	0.54
15–27	20	55.0%	0.57
16–28	20	55.0%	0.57

Figure 4.4 shows that the dynamics of forced accuracy in a triplet setting corresponds to the trend in forced accuracy in a usual setting. A dashed line in the figure demonstrates values of forced accuracy for a category-based confidence machine based on linear rules applied to individual patients; a solid

line represents forced accuracy produced by a confidence machine in a triplet setting. However, one should keep in mind that we cannot compare these two types of forced accuracy directly: if we make predictions at random, the accuracy in an individual patient setting is 50%, while in a triplet setting it equals 33.3%. Also, the difference is that, in a triplet setting, we have additional information that exactly one sample in each triplet is a case and the others are controls.

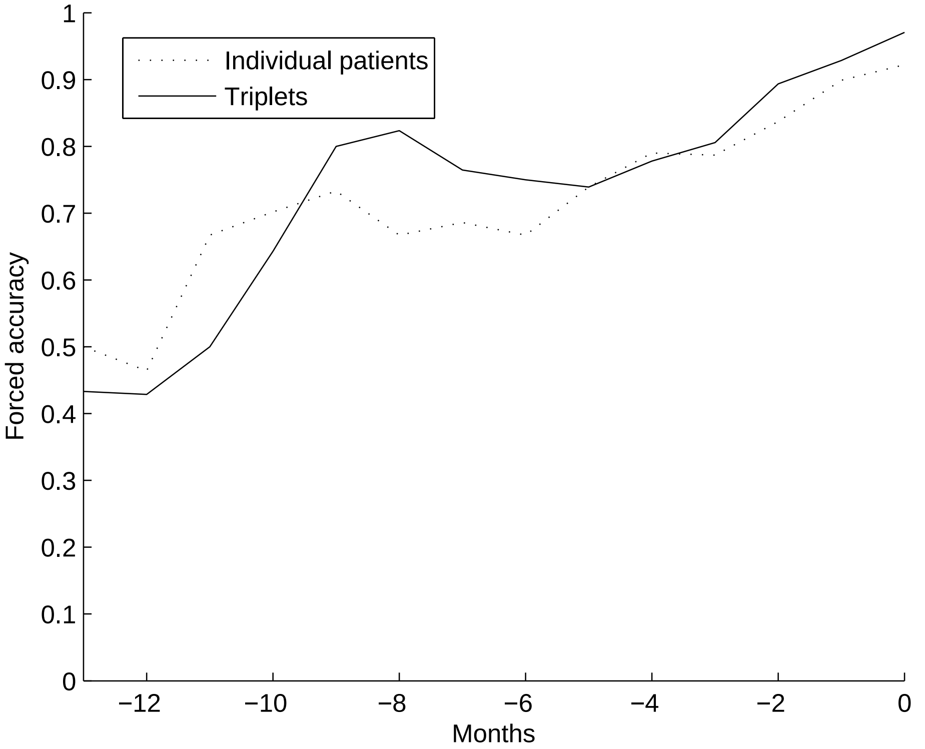


Figure 4.4: Dynamics of forced accuracy in a triplet setting and in an individual patient setting for the ovarian cancer data

Forced accuracy achieved on the breast cancer data (Table 4.13) is lower: it fluctuates around 80% from month 1 to month 5 in advance of diagnosis and is considerably lower in other months. However, achieved forced accuracy is still higher than the accuracy of 33.3%, which would be achieved on random data.

When we compare forced accuracy in a triplet setting and in an individual patient setting on the breast cancer data (see Figure 4.5), it can be easily observed that forced accuracy on individual patients remain high much longer (at least 9 months in advance), while in a triplet setting high forced accuracy is achieved much later, about 5 months in advance. By 'high' accuracy we mean values around or above 70%. This threshold is chosen because of clear visual separation in Figure 4.5.

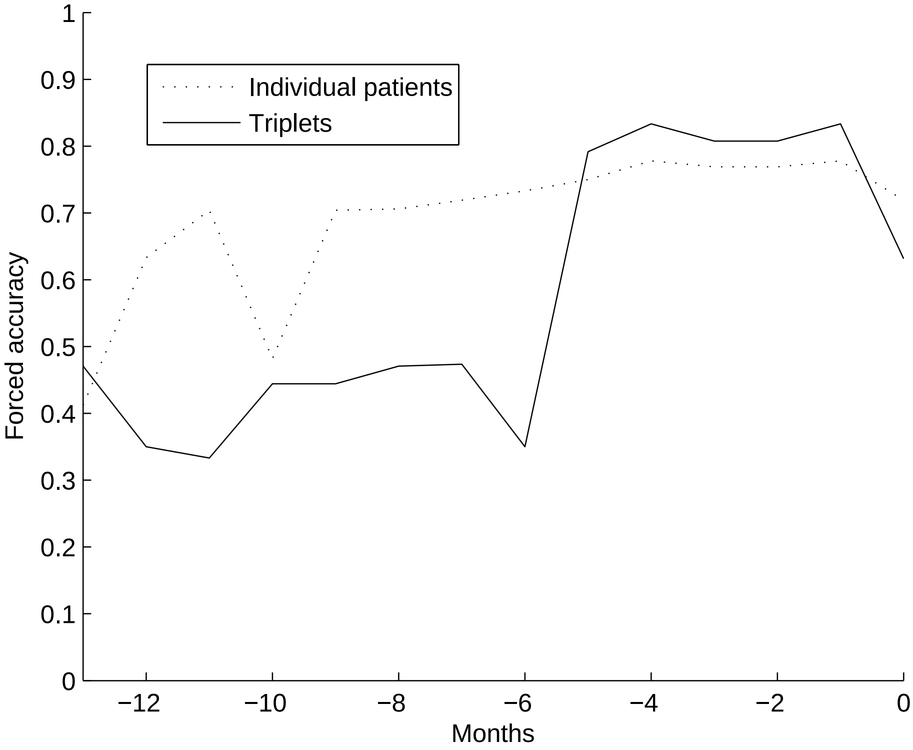


Figure 4.5: Dynamics of forced accuracy in a triplet setting and in an individual patient setting for the breast cancer data

In general, the results achieved by confidence machines in a triplet setting may be not reliable because the number of triplets in each time slot is not high enough. This confidence machine requires further investigation on mass spectrometry data sets with a a larger number of examples.

In addition, these confidence machines would be more useful if the number of patients per group were larger than three: in this case we would have to output patients (maybe, several of them) within groups which are more likely to be diseased with a preset long-run error rate.

4.5 Contributions to Proteomics

The principal goal of the whole chapter is to develop machine learning methodology for analysis of mass spectrometry data. However, implementation and application of such methods to real world data allowed us to make preliminary conclusions of medical nature.

Firstly, we achieved good classification results on real world proteomic data of ovarian cancer and breast cancer. For example, when making predictions for the ovarian cancer data set just before the moment of diagnosis, both category-based confidence machines based on linear rules and logistic Venn machine achieve accuracy higher than 90%. This confirmed the hypothesis that samples can be compared to each other even if they are not matched, that is, triplets can be merged and still provide high accuracy of prediction.

Secondly, proposed methodologies allowed us to speculate how long in advance we can output accurate predictions for these diseases. For the ovarian cancer data, we can predict with output predictions with accuracy of 91.7% just before the moment of diagnosis and at least 66.7% up to 11 months in advance of the moment of diagnosis; for the breast cancer data, we can achieve accuracy of 70.4–77.8% for up to 9 months in advance of diagnosis.

Thirdly, algorithms allowed us to confirm mass spectrometry profile peaks previously identified as carrying statistically significant information for discrimination between controls and cases.

4.5.1 Selection of Peaks

Our previous research of the UKCTOCS data [10; 11; 15] devoted to triplet analysis allowed us to determine potential candidates for biomarkers. We identified certain mass spectrometry profile peaks that carry statistically significant information for the diagnosis of the diseases. Both category-based confidence machines and Venn machines can also indirectly pinpoint potential biomarkers. Despite the different nature of these three methods, category-based confidence machines and Venn machines confirm mass spectrometry profile peaks that carry statistically significant information for discrimination between controls and cases.

The triplet analysis was carried out for the same data sets but in a different experimental setting: the data were normalised against such factors as age, sample collection time and location, storage and transportation conditions. All samples were grouped in triplets comprising one case and two controls matched by these factors. Thus, when making predictions, we had additional information about label distribution: we knew that exactly one sample is diseased in a triplet.

We will consider the time slots when Venn machines and category-based confidence machines produced high accuracy on the data sets: 6-month windows finishing 0–11 months in advance for ovarian cancer and 12-month windows finishing 0–9 months in a advance for breast cancer. For ovarian cancer we are especially interested in time slots at least as early as month 10 because this is the first time slot when CA125 on its own does not provide statistically significant discrimination between cases and controls. We will again consider one time slot including all samples for heart disease.

Category-based confidence machines help us identify potential biomarkers in the following way. When we run the leave-one-out procedure, for each possible label we choose the best rule $w_1 \log(C) + w_2 \log I(n_1) > \theta$ (for ovarian cancer) or $w_1 \log I(n_1) > \theta$ (for breast cancer and heart disease), which contains peak n_1. The selected peak may not be the same for every possible label and every possible left out sample, but in the time slots we are mostly interested in, the same peak was selected as a part of the best rule, that is, we chose the same weights and peak number when leaving out an example. These are peak 6 for heart disease (the whole data set); peak 19 for breast cancer in time slots finishing with months 0–9, 11, 12; peak 3 for ovarian cancer in time slots finishing with months 9–11. The detailed results for ovarian cancer and breast cancer samples taken in different time slots are represented in Table 4.14. The table shows the peak which was selected most often (‘Top peak’ column) and

Table 4.14: Top peaks pinpointed by category-based confidence machines ('CCM') and Venn machines ('VM') in different time slots for the ovarian cancer and breast cancer data sets

	Ovarian cancer			Breast cancer		
	CCM		VM	CCM		VM
Time slot end	Top peak	Peak frequency	Top peak	Top peak	Peak frequency	Top peak
0	1	96.1%	4	19	100.0%	19
1	1	83.0%	2	19	100.0%	19
2	1	72.7%	2	19	100.0%	19
3	2	56.0%	2	19	100.0%	19
4	2	98.2%	2	19	100.0%	19
5	1	95.7%	1	19	100.0%	19
6	1	69.2%	2	19	100.0%	19
7	4	94.1%	2	19	100.0%	19
8	3	73.5%	2	19	100.0%	19
9	3	100.0%	3	19	100.0%	19
10	3	100.0%	3	19	87.0%	19
11	3	100.0%	3	19	100.0%	19
12	2	85.7%	5	19	100.0%	2
13	3	95.0%	5	6	78.4%	1
14	3	85.3%	2	15	100.0%	15
15	2	89.2%	2	14	67.5%	15
16	5	63.3%	3	14	67.5%	15

how often it was selected ('Peak frequency' column).

Logistic Venn machine also produces an explicit ranking of peaks. It can be extracted from the coefficients of the optimal value for parameter b calculated in Venn predictions. The optimal parameter $\hat{b}$ is recalculated for each example and each of two hypotheses; to summarise the coefficients for all runs, we calculate the mean value of coefficients. The most important features are those with the highest absolute value of their coefficients.

Table 4.14 shows the peak with the highest ranking produced in different time slots by Venn machines for ovarian cancer and breast cancer. The table demonstrates that peak 3 is the top ovarian cancer peak in months 9–11 and peak 19 is the highest ranked breast cancer peak in months 0–11.

Table 4.15: Numbers of the most important peaks selected with different methods for the heart disease, breast cancer and ovarian cancer data sets (corresponding m/z-values are shown in Table A.16 in Appendix A)

Method	Ovarian cancer	Breast cancer	Heart disease
Triplet analysis	2, 3	19	7, 6, 4
Category-based confidence machine	3	19	6
Venn machine	3	19	4, 6, 7

Table 4.15 summarises all peaks selected by three different approaches: the triplet analysis in a triplet setting, category-based confidence machines and Venn machines. Those peaks are shown that were selected in time slots of high interest: slots finishing with months 10–11 for ovarian cancer, 0–9 for breast cancer and the whole data set for heart disease. For heart disease, the three peaks with the highest ranking produced by Venn machines are provided. The m/z-values of the peaks shown in the table are given in Table A.16 in Appendix A. Table 4.15 demonstrates that algorithms with online validity confirm the peaks identified as carrying statistically significant information in the triplet setting.

For the ovarian cancer data in time slots finishing with month 10 or 11 all three methods select peak 3 (9297.8 Da). These are the time slots when CA125 on its own does not carry statistically significant information as shown in previous research (Appendix B and [57]). Ovarian cancer peak 3 was also pinpointed in research on other data sets. It is similar to the peak CTAPIII with m/z-value 9288 Da, which was validated in another study using clinical samples from ovarian cancer women and controls [56]. In this study too lower intensities were noted for ovarian cancer samples. In addition, peak 3 coincides with peak 7 (m/z-value range 9294.7–9319.7 Da) previously found in the analysis of similar serial ovarian cancer samples and controls in the pilot [26; 40] trial which preceded UKCTOCS.

The predictive ability of CA125 on its own and in combination with ovarian cancer peak 3 is represented in Figure 4.6 [57]. The figure shows the median dynamics of values $\log C$ versus $\log C - \log I(3)$ for case measurements. For

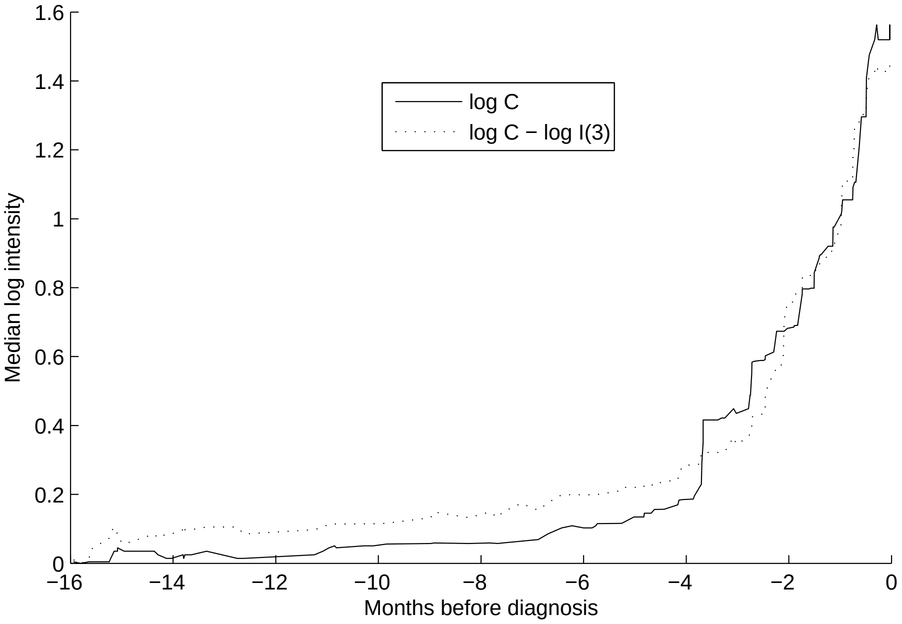


Figure 4.6: Median dynamics of rules $\log C$ and $\log C - \log I(3)$ (for ovarian cancer cases only) [57]

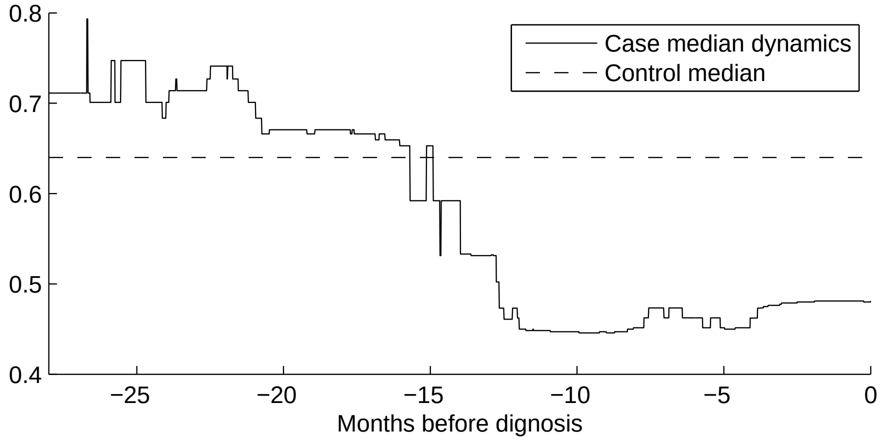


Figure 4.7: Median dynamics of peak 19 for cases and the median of peak 19 for controls in the breast cancer data [16]

each time moment, the latest available case measurement for each triplet group is taken into account. These measurements are averaged by median through all samples. The figure illustrates that the combination of CA125 with peak 3 starts to grow earlier than $\log C$. However, the CA125 growth at the moments close to diagnosis is quicker due to the exponential growth of CA125.

For the breast cancer data, we will observe the dynamics of peak 19 identified as a potential biomarker, whose intensities are supposed to be lower for cases rather than for controls according to our research. In Figure 4.7 [16], a solid line represents the median dynamics of peak 19 for breast cancer cases, a dashed line represents the peak 19 median calculated for all breast cancer controls. The values in the figure were calculated for samples within 9-month windows ending with the month shown on the horizontal axis.

One can see from Figure 4.7 that peak 19 median intensity drops about 15 months in advance of the moment of the diagnosis, which confirms our hypothesis about predictive ability of peak 19 and explains the results we obtained using this peak when discriminating between breast cancer cases and controls.

4.6 Summary

This chapter introduced algorithms with online validity for the analysis of mass spectrometry data. We designed algorithms which have guaranteed validity and are adapted to the needs of proteomics research. The algorithms are the category-based confidence machine based on linear rules, the logistic Venn machine and the confidence machine in a triplet setting. All of them are applied in this chapter within the framework of time dependency analysis.

We applied the designed methods to real-world MALDI-TOF mass spectrometry data sets or the UKCTOCS and demonstrated how they work. First of all, all of these algorithms allowed us to complement each individual prediction with additional information on its reliability (confidence or a probability interval). Second, the property of validity was proved to hold true on all data sets and all algorithms. At the same time in certain time slots, algorithms provided high efficiency, especially on the ovarian cancer data set. Close to the moment of diagnosis most region predictions produced by the category-based confidence machine contained exactly one label, similarly to the output of simple predictors. As for logistic Venn machines, probability intervals produced by them for the heart disease and ovarian cancer data were more accurate than the output of corresponding probability predictor of logistic regression: probability predictions can be misleading while multiprobability predictors output a narrow interval which almost always covered true label probability.

Even though category-based confidence machines and Venn machines are designed to output region and multiprobability predictions, they can be forced to output singleton predictions. Their forced accuracy was approximately the same as the accuracy of underlying algorithms. Confidence machines in a triplet setting output lower forced accuracy. Since the number of triplets was, in general, small, the method requires further investigation on larger data sets.

Forced predictions allow us to speculate how long in advance we can output accurate predictions for these diseases. For example, for the ovarian cancer data, we could achieve accuracy of 91.7% (or 90.2%) just before the moment of diagnosis and at least 70.6% up to 10 months in advance of the moment of diagnosis; for the breast cancer data, we could output predictions with accuracy of 70.4–77.8% for up to 9 months in advance of diagnosis. This demonstrated that samples can be compared to each other even if they are not matched by sample collections location and time, that is, we can merge triplets and still make accurate predictions.

Finally, the designed methods confirmed mass spectrometry profile peaks previously identified as carrying statistically significant information for discrimination between controls and cases. Ovarian cancer peak 3 was pinpointed in the triplet analysis and confirmed by category-based confidence machines and Venn machines. The analysis of median dynamics showed that the combination of CA125 with peak 3 starts to grow earlier in advance of the moment of diagnosis than CA125 on its own. Breast cancer peak 19 was confirmed as a potential biomarker whose intensities are lower for cases rather than for controls. Further analysis demonstrated that peak 19 median intensity of breast cancer cases drops about 15 months in advance of the moment of diagnosis.

Chapter 5

An Algorithm with Online Validity in the Linear Regression Model

All algorithms with online validity investigated in this thesis are based on the i.i.d. assumption. In this chapter, we aim to extend the class of algorithms which output valid predictions beyond this assumption. Our first attempt is related to the linear regression model with i.i.d. errors with a known distribution. We introduce a new interval predictor which has the property of *exact validity* under this statistical model.

Exact validity is stronger that validity of confidence machines considered before. Confidence machines make predictions with the error rate which does not exceed the preset significance level asymptotically while the property of exact validity implies that the error rate converges to the significance level. This property does not hold true for confidence machines but does hold true for their modification — smoothed confidence machines — under the standard i.i.d. assumption.

This theoretical research was carried out in collaboration with Peter McCullugh, Vladimir Vovk, Ilia Nouretdinov and Alexander Gammerman. My contribution comprises a proof of Lemma 5.1, which made the construction of prediction intervals consistent, and computational experiments described in Section 5.7.

5.1 Exact Validity of Smoothed Confidence Machines

First, we will state the property of exact validity. It was proved in [65] for smoothed confidence machines.

A confidence predictor is said to be *exactly valid* with respect to the statistical model on Z^∞ if for any distribution P from this statistical model, $\mathrm{err}_1^\epsilon(\Gamma, P)$, $\mathrm{err}_2^\epsilon(\Gamma, P)$, ... are a sequence of independent Bernoulli random variables with parameter ϵ, i.e., are equal to 0 with probability $1-\epsilon$ and 1 with probability ϵ. In other words, exact validity implies that the confidence predictor makes errors independently with probability ϵ at each step.

The immediate consequence of exact validity and the law of large numbers is the property of asymptotic validity. The confidence predictor Γ is *asymptotically exact* with respect to a statistical model if for any probability distribution P from this model generating examples and any significance level ϵ,

$$\limsup_{n\to\infty} \frac{\mathrm{Err}_n^\epsilon(\Gamma, P)}{n} = \epsilon$$

with probability one. In words, the error rate asymptotically converges to the preset significance level in the online mode.

It was shown that no confidence machine is exactly valid, however, their modification, smoothed confidence machines, have this property. A smoothed confidence machine is also a confidence predictor, and the general framework is the same as for confidence machine: we use the same problem setting, the same i.i.d. assumption, define the strangeness measure A_n and calculate strangeness scores α_i according to 2.3.

A new element in this framework is a sequence of random variables τ_1, τ_2, ... which are independent and uniformly distributed in $[0,1]$, i.e., they are outcomes of a classical random number generator. They are involved in calculating p-values:

$$p_n(y, \tau_n) = \frac{|\{i = 1, \dots, n : \alpha_i > \alpha_n\}| + \tau_n |\{i = 1, \dots, n : \alpha_i = \alpha_n\}|}{n}.$$

Smoothed confidence machine determined by the strangeness measure A_n is then defined as a function

$$\Gamma : (\mathbf{X} \times [0,1] \times \mathbf{Y})^* \times (\mathbf{X} \times [0,1]) \times (0,1) \to 2^{\mathbf{Y}}$$

which outputs the following region prediction:

$$\Gamma^\epsilon(x_1, \tau_1, y_1, \ldots, x_{n-1}, \tau_{n-1}, y_{y-1}, x_n, \tau_n) = \{y \in \mathbf{Y} | p(y, \tau_n) > \epsilon\}.$$

We define $\mathrm{err}_n^\epsilon(\Gamma, \omega)$ and $\mathrm{Err}_n^\epsilon(\Gamma, \omega)$ the same way as for confidence machines (Section 2.1.2), only now these values depend on τ_i, $i = 1, \ldots, n$. Finally, if Γ is a smoothed confidence machine, P is an exchangeable distribution on $\mathbf{Z}^\infty$, $n \in \mathbb{N}$, random variables

$$\mathrm{err}_n^\epsilon(\Gamma, (x_1, \tau_1, y_1, x_2, \tau_2, y_2, \ldots))$$

and

$$\mathrm{Err}_n^\epsilon(\Gamma, (x_1, \tau_1, y_1, x_2, \tau_2, y_2, \ldots)) ,$$

where (x_1, y_1), $(x_2, y_2), \ldots$ are drawn from P, and τ_i are independent and uniformly distributed on $[0,1]$, will be denoted by $\mathrm{err}_n^\epsilon(\Gamma, P)$ and $\mathrm{Err}_n^\epsilon(\Gamma, P)$, respectively.

Smoothed confidence machines are *exactly valid* in respect to the i.i.d. assumption: for any exchangeable distribution P a smoothed confidence machine makes errors independently with probability ϵ at each step.

5.2 Statistical Model and Fundamental σ-algebras

Now we consider a new statistical model different from the standard i.i.d. assumption, and we attempt to construct region predictions which have the same property of exact validity with respect to the new statistical model.

The new model we are considering is the linear regression model with i.i.d. errors with a known distribution, not necessarily Gaussian.

Let $\mathbf{x}_1, \mathbf{x}_2, \ldots$ be a given sequence of vectors in $\mathbb{R}^m$; their elements (explanatory variables) will be denoted $x_{i,j}$, $i = 1, 2, \ldots$, $j = 1, \ldots, m$. Our statistical model (P_θ), parameterised by $\theta = (\beta, \sigma) \in \mathbb{R}^m \times (0, \infty)$ is that the sequence of observations $y_1, y_2, \ldots$ is generated by

$$y_i = \beta' \mathbf{x}_i + \sigma \xi_i \,, \tag{5.1}$$

where $\xi_1, \xi_2, \ldots$ is a sequence of i.i.d. noise variables with a known distribution P. Each $P_{\beta,\sigma}$ is a probability measure on $\mathbb{R}^\infty$ (the distribution of the sequence of labels, with the instances fixed). P is a continuous probability measure on $\mathbb{R}$ with density p.

This model may appear narrower than the i.i.d. model, standard in algorithms with online validity, but its advantage is that the instances $\mathbf{x}_i$ can be controlled rather than chosen independently from the same distribution.

Let $\mathcal{G}$ be the group of all transformations

$$g_{a,b} : (y_1, y_2, \ldots) \mapsto (a'\mathbf{x}_1 + by_1, a'\mathbf{x}_2 + by_2, \ldots) \,,$$

where $a \in \mathbb{R}^m$ and $b > 0$, acting on $\mathbb{R}^\infty$. We consider two fundamental σ-algebras on the set $\mathbb{R}^\infty$ of all infinite sequences $(y_1, y_2, \ldots)$ of real numbers. The σ-algebra $\mathcal{F}$ consists of all Borel sets in $\mathbb{R}^\infty$ (it will be our default σ-algebra on $\mathbb{R}^\infty$); by *events* we mean elements of $\mathcal{F}$. For $i = 1, 2, \ldots$, let $Y_i : \mathbb{R}^\infty \to \mathbb{R}$ be the projection onto the ith component: $Y_i(y_1, y_2, \ldots) := y_i$. For each $n = 0, 1, \ldots$, the σ-algebra $\mathcal{F}_n$ on $\mathbb{R}^\infty$ is defined as $\mathcal{F}_n := \sigma(Y_1, \ldots, Y_n)$. The σ-algebra $\mathcal{K}$ on $\mathbb{R}^\infty$ consists of all *invariant* events in $\mathbb{R}^\infty$, i.e., all events that are invariant with respect to the group $\mathcal{G}$. The σ-algebra $\mathcal{K}_n$ on $\mathbb{R}^\infty$ is defined as $\mathcal{K}_n := \mathcal{F}_n \cap \mathcal{K}$.

5.3 Normalisation

Let $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{n-1}, y_{n-1})$ be the training set of the fixed size $(n-1)$. We will consider the linear transformation $\nu : Y \to Y$ on a label of the next object (called *normalising transformation*):

$$z_n = \nu(y_n) = \frac{y_n - \tilde{\beta}\mathbf{x}_n}{\tilde{\sigma}}\,, \tag{5.2}$$

where the pair $(\tilde{\beta}, \tilde{\sigma})$ is the solution of the the following maximisation problem:

$$\prod_{i=1}^{n-1} \left(\frac{1}{\tilde{\sigma}}\right) q \left(\frac{y_i - \tilde{\beta}'\mathbf{x}_i}{\tilde{\sigma}}\right) \to \max\,, \tag{5.3}$$

where q is a probability density function that corresponds to a continuous probability measure, but not necessarily to distribution P. This optimisation problem is equivalent to likelihood maximisation problem for the same model as described above, but only when $\xi_1, \xi_2, \dots$ is a sequence of i.i.d. noise variables with a distribution Q and a probability density function q.

Lemma 5.1 *The distribution of z_n does not depend on true (β, σ).*

Proof

$$\prod_{i=1}^{n-1} \left(\frac{1}{\tilde{\sigma}}\right) q \left(\frac{y_i - \tilde{\beta}'\mathbf{x}_i}{\tilde{\sigma}}\right) = \prod_{i=1}^{n-1} \left(\frac{1}{\tilde{\sigma}}\right) q \left(\frac{\beta' - \tilde{\beta}'}{\tilde{\sigma}}\mathbf{x}_i + \frac{\sigma}{\tilde{\sigma}}\xi_i\right)\,, \tag{5.4}$$

where (β, σ) are true values of model parameters.

Transformation ν can be then presented in the following way:

$$z_n = \nu(y_n) = \frac{y_n - \tilde{\beta}\mathbf{x}_n}{\tilde{\sigma}} = \frac{\beta' - \tilde{\beta}'}{\tilde{\sigma}}\mathbf{x}_n + \frac{\sigma}{\tilde{\sigma}}\xi_n\,. \tag{5.5}$$

It is sufficient to prove the two following facts:

1. The distribution of z_n is the same for $(\beta, \sigma) = (\beta_1, \sigma_0)$ and $(\beta, \sigma) = (\beta_2, \sigma_0)$.

2. The distribution of z_n is the same for $(\beta, \sigma) = (\vec{0}, \sigma_1)$ and $(\beta, \sigma) = (\vec{0}, \sigma_2)$.

To prove the first statement, we will notice that there is a bijective mapping between $(\tilde{\beta}, \tilde{\sigma})$ and $\left(\frac{\beta' - \tilde{\beta}'}{\tilde{\sigma}}, \tilde{\sigma}\right)$ when β and σ are fixed. Thus to solve the optimisation problem (5.3), we need to find a pair of values $\left(\frac{\beta'-\tilde{\beta}'}{\tilde{\sigma}}, \tilde{\sigma}\right)$ that maximises (5.4).

This means that for pairs (β_1, σ_0) and (β_2, σ_0) with the same $\sigma = \sigma_0$, solutions of the corresponding maximisation problems $(\tilde{\beta}_1, \tilde{\sigma}_1)$ and $(\tilde{\beta}_2, \tilde{\sigma}_2)$ will satisfy $\frac{\beta_1'-\tilde{\beta}_1'}{\tilde{\sigma}_1} = \frac{\beta_2'-\tilde{\beta}_2'}{\tilde{\sigma}_2}$ and $\tilde{\sigma}_1 = \tilde{\sigma}_2$; therefore, the distribution of z_n will be the same for $(\beta, \sigma) = (\beta_1, \sigma_0)$ and $(\beta, \sigma) = (\beta_2, \sigma_0)$.

Fact 2 is proved as follows. Let $(\tilde{\beta}_1, \tilde{\sigma}_1)$ and $(\tilde{\beta}_2, \tilde{\sigma}_2)$ be the solutions of the maximisation problem (5.3) with $(\beta, \sigma) = (\vec{0}, \sigma_1)$ and $(\beta, \sigma) = (\vec{0}, \sigma_2)$, respectively, and $\sigma_2 = a\sigma_1,\ a > 0$.

Since $(\tilde{\beta}_1, \tilde{\sigma}_1)$ maximises (5.3), then $\forall \hat{\sigma} > 0, \hat{\beta}$

$$\prod_{i=1}^{n-1} \left(\frac{1}{\tilde{\sigma}_1}\right) q\left(-\frac{\tilde{\beta}_1'}{\tilde{\sigma}_1}\mathbf{x}_i + \frac{\sigma_1}{\tilde{\sigma}_1}\xi_i\right) \geq \prod_{i=1}^{n-1} \left(\frac{1}{\hat{\sigma}}\right) q\left(-\frac{\hat{\beta}'}{\hat{\sigma}}\mathbf{x}_i + \frac{\sigma_1}{\hat{\sigma}}\xi_i\right). \tag{5.6}$$

It is then easy to see that $\tilde{\beta}_2 = a\tilde{\beta}_1$ and $\tilde{\sigma}_2 = a\tilde{\sigma}_1$ because, for arbitrary $\hat{\sigma} > 0$ and $\hat{\beta}$ and $\tilde{\beta}_2 = a\tilde{\beta}_1$ and $\tilde{\sigma}_2 = a\tilde{\sigma}_1$,

$$\begin{aligned}
\prod_{i=1}^{n-1} \left(\frac{1}{\tilde{\sigma}_2}\right) q\left(-\frac{\tilde{\beta}_2'}{\tilde{\sigma}_2}\mathbf{x}_i + \frac{\sigma_2}{\tilde{\sigma}_2}\xi_i\right) &= \left(\frac{1}{a}\right)^{n-1} \prod_{i=1}^{n-1} \left(\frac{1}{\tilde{\sigma}_1}\right) q\left(-\frac{\tilde{\beta}_1'}{\tilde{\sigma}_1}\mathbf{x}_i + \frac{\sigma_1}{\tilde{\sigma}_1}\xi_i\right) \\
&\geq \left(\frac{1}{a}\right)^{n-1} \prod_{i=1}^{n-1} \left(\frac{1}{\hat{\sigma}/a}\right) q\left(-\frac{(\hat{\beta}/a)'}{\hat{\sigma}/a}\mathbf{x}_i + \frac{\sigma_1}{\hat{\sigma}/a}\xi_i\right) \\
&= \prod_{i=1}^{n-1} \left(\frac{1}{\hat{\sigma}}\right) q\left(-\frac{\hat{\beta}'}{\hat{\sigma}}\mathbf{x}_i + \frac{\sigma_2}{\hat{\sigma}}\xi_i\right),
\end{aligned}$$

that is, $(a\tilde{\beta}_1, a\tilde{\sigma}_1)$ is a solution of a maximisation problem for $(\beta, \sigma) = (\vec{0}, \sigma_2)$.

Hence $\tilde{\beta}_2 = a\tilde{\beta}_1$, $\tilde{\sigma}_2 = a\tilde{\sigma}_1$ and $z_n = -\frac{\tilde{\beta}_1'}{\tilde{\sigma}_1}\mathbf{x}_n + \frac{\sigma_1}{\tilde{\sigma}_1}\xi_n$ for both $(\beta, \sigma) = (\vec{0}, \sigma_1)$ and $(\beta, \sigma) = (\vec{0}, \sigma_2)$. ■

Considering different distributions Q, we can obtain different normalisations:

1. $Q = P$.

2. If Q is the Gaussian distribution $N(0, 1)$, $\tilde{\beta}$ is a solution for the min-

imisation problem $\sum_{i=1}^{n-1}(y_i - \tilde{\beta}'\mathbf{x}_i)^2 \to \min$ (the least square estimate); $\tilde{\sigma} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n-1}(y_i - \tilde{\beta}'\mathbf{x}_i)^2}$.

3. If Q is the Laplace distribution $(0, 1)$, $\tilde{\beta}$ is a solution for the minimisation problem $\sum_{i=1}^{n-1}|y_i - \tilde{\beta}'\mathbf{x}_i| \to \min$; $\tilde{\sigma} = \frac{1}{n-1}\sum_{i=1}^{n-1}|y_i - \tilde{\beta}'\mathbf{x}_i|$.

We are going to consider normalisation 2 based on the Gaussian distribution.

5.4 Prediction Intervals

By the *Gosset measure* G (with respect to the density p, transformation ν defined above and sample size $(n-1)$) we will mean the image $P_{\beta,\sigma}\nu^{-1}$ of any measure $P_{\beta,\sigma}$ under the normalising transformation ν. Lemma 5.1 says that it does not matter which β and σ we take.

In this section a training set $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{n-1}, y_{n-1})$ of the fixed size $n-1$ will usually be represented as the $(n-1) \times m$ matrix $\mathbf{X}$ whose rows are the vectors $\mathbf{x}_i'$, $i = 1, \ldots, n-1$, and the $(n-1) \times 1$ vector $\mathbf{y}$ of all y_is. We will always assume that $n - 1 > m + 1$ and that $\mathbf{X}$ is a full rank matrix. Our goal is to predict the label of a new instance $\mathbf{x}_n$.

Fix a significance level $\epsilon \in (0, 1)$. An *interval predictor* is a pair of measurable functions $L : \mathbb{R}^{n-1} \to \mathbb{R}$ and $U : \mathbb{R}^{n-1} \to \mathbb{R}$ such that $L \leq U$. Another representation of the interval predictor is as the function

$$\Gamma(\mathbf{y}) := [L(\mathbf{y}), U(\mathbf{y})]$$

mapping the labels to the corresponding prediction interval. The interval predictor is called *unconditionally valid* (for a given statistical model) if its coverage probability is $1-\epsilon$: $\mathbb{P}(\mathrm{err}_n^\epsilon) = \epsilon$ under each probability measure in the given model, where $\mathrm{err}_n^\epsilon = \mathrm{err}_n^\epsilon(\Gamma; Y_1, Y_2, \ldots)$ is the event $\{Y_n \notin \Gamma(Y_1, \ldots, Y_{n-1})\}$ (which is called an *error*).

An interval predictor is called $\mathcal{K}_{n-1}$*-valid* if $\mathbb{P}(\mathrm{err}_n^\epsilon \mid \mathcal{K}_{n-1}) = \epsilon$ a.s.

There exist many interval predictors Γ such that

$$G(\mathrm{err}_n^\epsilon \mid \mathcal{F}_{n-1}) = \epsilon \quad \text{a.s.} \tag{5.7}$$

(assuming that the conditional distribution of Y_n given $\mathcal{F}_{n-1}$ with respect to the Gosset measure is continuous; this assumption is satisfied for the noise distributions used in our empirical studies in Section 5.7).

Given an interval predictor Γ for the Gosset measure G, we can define an interval predictor Γ' for the original linear regression model $(P_{\beta,\sigma})$ as

$$\Gamma'(y_1,\dots,y_{n-1}) := \tilde{\sigma}_y \Gamma\left(\frac{y_1 - \tilde{\beta}_y' \mathbf{x}_1}{\tilde{\sigma}_y},\dots,\frac{y_{n-1} - \tilde{\beta}_y' \mathbf{x}_{n-1}}{\tilde{\sigma}_y}\right) + \tilde{\beta}_y' \mathbf{x}_n\,, \tag{5.8}$$

where we again use the notation $\tilde{\beta}_y := \tilde{\beta}(y_1,\dots,y_{n-1})$ and $\tilde{\sigma}_y := \tilde{\sigma}(y_1,\dots,$ $y_{n-1})$. In words, to obtain $\Gamma'(y_1,\dots,y_{n-1})$ we first normalise $(y_1,\dots,y_{n-1})$, then apply Γ to obtain a prediction interval and finally apply the inverse transformation to that prediction interval.

Proposition 5.1 *If* Γ *is an interval predictor satisfying (5.7), the interval predictor* Γ' *defined by (5.8) is* $\mathcal{K}_{n-1}$*-valid.* □

Proof can be found in [36].

Now we can define our interval predictor. Among the interval predictors Γ satisfying (5.7) we choose the *symmetric* one, i.e., the interval predictor $\Gamma = [L, U]$ such that $G(Y_n < L \mid \mathcal{F}_{n-1}) = G(Y_n > U \mid \mathcal{F}_{n-1}) = \epsilon/2$ a.s. Such an interval predictor is essentially unique. The predictor Γ' defined by (5.8) will be called the *symmetric pivotal interval predictor* (abbreviated to *SPIP*). Proposition 5.1 says that it is $\mathcal{K}_{n-1}$-valid. Hence it is unconditionally valid.

5.5 Validity in the Online Mode

In this section we will consider the online mode: we start from the empty training set and add each new example $(\mathbf{x}_n, y_n)$, $n = 1, 2, \dots$, to the training set after predicting its label y_n.

The SPIP defined above can be considered as a confidence predictor — a predictor which for any given finite sequence of labelled objects (x_1, y_1), (x_2, y_2), …, a new object x_n without a label and a significance level ϵ outputs a subset of the label space:

$$\Gamma^{\epsilon}(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n),$$

so that

$$\Gamma^{\epsilon_1}(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n) \subseteq \Gamma^{\epsilon_2}(x_1, y_1, \dots, x_{n-1}, y_{n-1}, x_n)$$

for any $\epsilon_1 \geq \epsilon_2$. This means that prediction regions for different ϵ represent nested subsets of $\mathbf{Y}$, and by changing significance level ϵ we can regulate the size of the output prediction. Thus confidence machines and designed SPIP belong to the same class of confidence predictors.

It follows from Proposition 5.1 that for SPIP $\mathbb{P}(\mathrm{err}_n^{\epsilon} \mid \mathrm{err}_1^{\epsilon}, \dots, \mathrm{err}_{n-1}^{\epsilon}) = \epsilon$ a.s. for each $n = 1, 2, \dots$. This implies that interval predictors are *exactly valid with respect to the model* $(P_{\beta,\sigma})$: for any distribution from the statistical model $(P_{\beta,\sigma})$, $\mathrm{err}_1^{\epsilon}, \mathrm{err}_2^{\epsilon}, \dots$ are a sequence of independent Bernoulli random variables with parameter ϵ, that is, output erroneous prediction intervals independently with probability ϵ at different steps. This is the same property as the property of exact validity of smoothed confidence machines. Similarly, SPIPs are asymptotically valid with respect to model $(P_{\beta,\sigma})$: for any probability distribution P from this model generating examples and any significance level ϵ,

$$\limsup_{n \to \infty} \frac{\sum_{i=1}^{n} \mathrm{err}_i^{\epsilon}}{n} = \epsilon$$

with probability one. In our empirical studies (Section 5.7), we will demonstrate that SPIPs are asymptotically exact in the offline mode.

5.6 MCMC Implementation of the Algorithm

Suppose $(Y_1, Y_2, \dots)$ are distributed according to $P_{0,1} = P^{\infty}$. Let $Z_i := \nu(Y_i)$, $i \in \mathbb{N}$, and let B and Σ be the random vector $\tilde{\beta}(Y_1, \dots, Y_{n-1})$ and random variable $\tilde{\sigma}(Y_1, \dots, Y_{n-1})$, respectively. If A and B are random elements, we let $f_{A|B}(a \mid b)$ stand for the conditional density of A at point a given $B = b$. We will also use similar notation for unconditional distributions: $f_A(a)$ stands for the density of A at a. We will be interested in continuous versions of conditional distributions, and so will omit the qualification "a.s."

The following lemma plays the main part in our prediction algorithm implementation.

Lemma 5.2 *Suppose* $\beta = 0$ *and* $\sigma = 1$. *The conditional density of* (B, Σ) *given* $Z_1 = z_1, \ldots, Z_{n-1} = z_{n-1}$ *is proportional to* $\tilde{\sigma}^{n-m-2} p(\tilde{\beta}'\mathbf{x}_1 + \tilde{\sigma} z_1) \cdots p(\tilde{\beta}'\mathbf{x}_{n-1} + \tilde{\sigma} z_{n-1})$ *(with the coefficient of proportionality a function of* $z_1, \ldots,$ z_{n-1}*).* □

Proof can be found in [36].

Our prediction algorithm, given as Algorithm 1, uses Markov chain Monte Carlo (MCMC) sampling [17] from the conditional distribution of $(\tilde{\beta}, \tilde{\sigma})$ given $\nu(y_1)$, …, $\nu(y_{n-1})$. Its inputs are the training set $\mathbf{x}_1, y_1, \ldots, \mathbf{x}_{n-1}, y_{n-1}$ and a new object $\mathbf{x}_n$. The duration of the burn-in period is B, and $B + M$ is the overall duration of the random walk. Let s_β and s_σ be small positive constants (the standard deviations of the Gaussian proposal distributions for B and $\log \Sigma$, respectively) and I be the identity matrix (whose size will be clear from the context). Fix a significance level $\epsilon \in (0, 1/2)$; for simplicity we assume that $M\epsilon/2$ is an integer.

The correctness of the algorithm is in detailed justified in [36]. In our empirical studies reported in the next section we use $B = M = 100{,}000$ and $s_\beta = s_\sigma = 0.1$.

5.7 Empirical Studies

In our empirical studies we apply Algorithm 1 to the following noise distributions: the Gaussian distribution with density $p(y) \propto e^{-y^2/2}$, the Laplace distribution with density $p(y) \propto e^{-|y|}$, and Student's t-distribution of 4 degrees of freedom with density $p(y) \propto (1 + y^2)^{-5/2}$ (up to scaling).

First we investigate the behaviour of our prediction intervals on artificially generated data sets. Figure 5.1 shows four validity plots, each of which is a graph of the error rate against the significance level for data generated by p_1 and the algorithm using p_2 (where p_1 and p_2 are to be defined later). For each

Algorithm 1 MCMC SPIP based on noise distribution with density p

```
Compute $\tilde{\beta}$, $\tilde{\sigma}$ from the training set;
for $i = 1, \dots, n-1$ do
  $z_i := (y_i - \tilde{\beta}'\mathbf{x}_i)/\tilde{\sigma}$;
end for
set $\beta_0 := 0$, $\sigma_0 := 1$, and $p_0 := p(z_1) \cdots p(z_{n-1})$;
for $j = 1, \dots, B+M$ do
  set $\beta_j := \beta_{j-1} + N(0, s_\beta^2 I)$ and $\log \sigma_j := \log \sigma_{j-1} + N(0, s_\sigma^2)$;
  set $p_j := \sigma_j^{n-m-2} p(\beta_j' \mathbf{x}_1 + \sigma_j z_1) \cdots p(\beta_j' \mathbf{x}_{n-1} + \sigma_j z_{n-1})$;
  if $p_j < p_{j-1}$ then
    with probability $1 - p_j/p_{j-1}$,
    redefine $\beta_j := \beta_{j-1}$, $\sigma_j := \sigma_{j-1}$,
  end if
  sample $\xi_j$ from $p$ and set $\zeta_j := (\xi_j - \beta_j' \mathbf{x}_n)/\sigma_j$;
end for
order $\zeta_{B+1}, \dots, \zeta_{B+M}$ into an increasing sequence
output the prediction interval $[\tilde{\beta}'\mathbf{x}_n + \tilde{\sigma}\zeta_{(M\epsilon/2)}, \tilde{\beta}'\mathbf{x}_n + \tilde{\sigma}\zeta_{(M(1-\epsilon/2))}]$.
```

plot we generated 5,500 examples $(\mathbf{x}_i, y_i)$ from the model (5.1) with $m = 1$, $\beta = \sigma = 2$, $\mathbf{x}_i$ generated independently from the uniform distribution on $[0, 1]$, and ξ_i generated independently (among themselves and $\mathbf{x}_1, \dots, \mathbf{x}_{5500}$) from a distribution with density p_1. The first 500 examples were used as the training set ($n = 501$) and the remaining 5,000 examples as the test set. We ran Algorithm 1 (with $n+k-1$ in place of n) based on a noise distribution with density p_2 to predict the label y_{n+k-1} of each test instance $\mathbf{x}_{n+k-1}$, $k = 1, \dots, 5000$, for a fine grid of significance levels $\epsilon \in (0, 0.2]$. Each of the four plots shows the percentage of the test examples $(\mathbf{x}_{n+k-1}, y_{n+k-1})$, $k = 1, \dots, 5000$, for which y_{n+k-1} was not covered by the prediction interval produced for $\mathbf{x}_{n+k-1}$ by Algorithm 1 as function of ϵ. We call them validity plots since the function being close to the bisector of the first quadrant means that the predictor is asymptotically valid: the prediction algorithm's frequency of error is close to the nominal significance level. We concentrate on the most interesting range of small ϵ, which includes, in particular, the standard values of 5% and 1%. Since the results of our experiments are random, each validity plot is shown for five different initial states of the MATLAB random number generator.

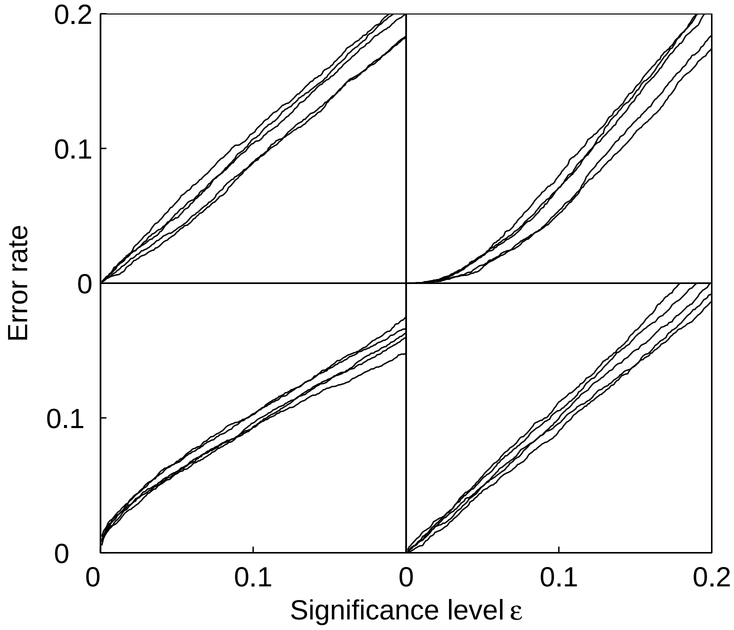


Figure 5.1: Validity plots for the Gaussian prediction intervals on Gaussian (top left) and Laplace (bottom left) data and for the Laplace prediction intervals on Gaussian (top right) and Laplace (bottom right) data. The horizontal axis is $\epsilon \in (0, 0.2]$, and the range of the vertical axis is also $[0, 0.2]$.

The top left plot has both p_1 and p_2 equal to the Gaussian distribution, and the bottom right plot has both p_1 and p_2 equal to the Laplace distribution. These two plots demonstrate empirically the validity of our prediction algorithm: when it is provided with the correct model, its predictions are valid. The top right plot describes an application of a robust prediction algorithm (based on the Laplace distribution) to benign (Gaussian) data. The algorithm is rather conservative: at significance level 5% it typically makes between 1% and 2% of errors, while at 1% the percentage of errors is typically below 0.05%. The bottom left plot describes an application of an optimistic prediction algorithm (based on the Gaussian distribution) to somewhat unruly (Laplace) data. For interesting values of the significance level, the predictions are not valid: at significance level 5%, the percentage of wrong predictions is around

6–7%, and at 1% it is around 2–3%.

In Figure 5.2 we give the median widths of the prediction intervals at significance levels $\epsilon \in (0, 0.2]$, with p_1 being the Gaussian distribution for the two top plots and the Laplace distribution for the two bottom plots, and with p_2 being the Gaussian distribution for the two left-hand plots and the Laplace distribution for the two right-hand plots.

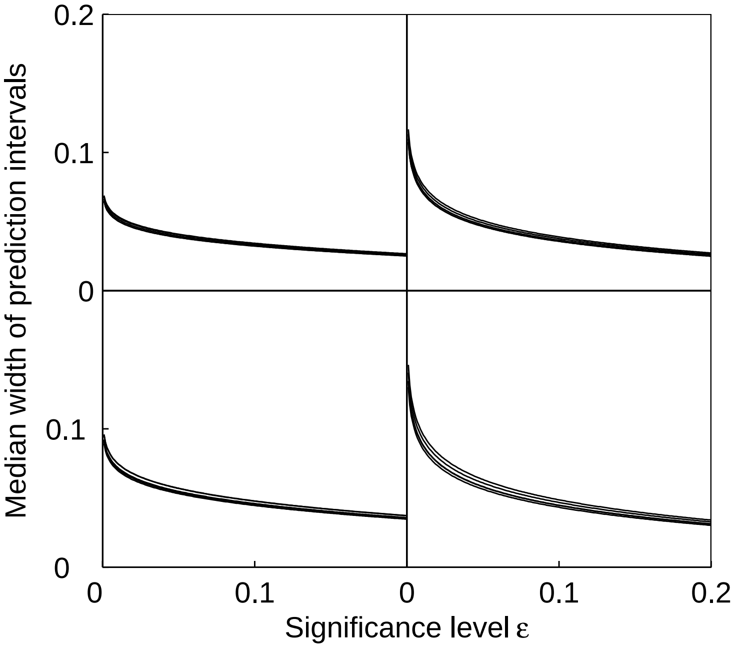


Figure 5.2: The median widths of prediction intervals for various $\epsilon \in (0, 0.2]$, with the same layout as Figure 5.1. The range of the vertical axis is always $[0, 40]$

We know that the unconditional, and conditional on $\mathcal{K}_{n-1}$, coverage probability of our prediction intervals is equal to the confidence level $1 - \epsilon$; this is illustrated by the top left and bottom right plots of Figure 5.1. An interesting question is how stable the fully conditional, i.e., conditional on $\mathcal{F}_{n-1}$, coverage probabilities are. The results for our experimental setup are shown in Figure 5.3. We generated five training sets, each of size 500; box plots 1 to 5 describe the results for the first training set, 6 to 10 for the second training set, etc.

For each training set we generated five test sets of size 5,000 following the same distribution. For each of the test examples $(\mathbf{x}, y)$ we computed the fully conditional coverage probability of the corresponding prediction interval (computed from the instance $\mathbf{x}$ and the corresponding training set, with the label y ignored). Box plot 1 gives some statistics for the first test set generated for the first training set, box plot 2 gives statistics for the second test set generated for the first training set, etc. Namely, each box plot gives the median coverage probability, the quartile coverage probabilities, and the maximum and minimum coverage probabilities for the prediction intervals generated for the test instances. We can see that for the same training set the box plots are very similar (because of the large size of the test sets), but the variation of coverage probabilities between the training sets is substantial.

We have also applied three kinds of prediction intervals to the ChickWeight data set ([14], Example 5.3; [30], Table A.2; part of the standard R distribution, package `datasets`). The data set gives weight versus age of chicks on different diets. The body weights of the chicks were measured at birth, every second day thereafter until day 20 and on day 21. The range of the body weights is 35 to 373 grams. There are four groups of chicks on different protein diets. Our task was to predict a chick's weight given its age. We used the chicks on diets 1 and 2 as the training set (of size 340) and the chicks on diets 3 and 4 as the test set (of size 238).

It is clear that all three models that we have discussed are wrong for this data set, for a multitude of reasons, and our question is which is more useful for predicting the weights of the chicks in the test set. Figure 5.4 shows that the Laplace prediction intervals (i.e., those produced by Algorithm 1 based on the Laplace distribution) are fairly asymptotically valid over our range of significance levels, and that the t4 prediction intervals (i.e., those produced by Algorithm 1 based on the t distribution on 4 degrees of freedom) are not very different. Figure 5.5 gives the median widths of the prediction intervals at significance levels $\epsilon \in (0, 0.2]$, as in Figure 5.2.

In general, we have found that the best validity was usually achieved by the Gaussian prediction intervals (with the Laplace and t4 ones somewhat conservative) or by the Laplace and t4 prediction intervals (with the Gaussian ones somewhat invalid). The performance of Laplace and t4 prediction intervals was broadly similar, despite the different nature of the tails of the corresponding noise distributions (decaying exponentially fast in the case of Laplace and polynomially fast in the case of t4).

To be on the safe side, our recommendation would be to use the Laplace or t4 prediction intervals when in doubt. Alternatively, a safe prediction algorithm could output the union of the Gaussian, Laplace, and t4 prediction intervals.

5.8 Summary

The research covered in this section extended the class of algorithms which produce valid predictions. The main difference of SPIPs from the other algorithms with online validity investigated in this thesis is that SPIPs are based on the assumption different from the standard i.i.d. assumption used in confidence machines, category-based confidence machines and Venn machines. Nevertheless, SPIPs are exactly valid and asymptotically valid.

The experimental studies confirmed the property of validity on artificially generated data. We also investigated application of SPIPs to real world data and came up with a general recommendation to use prediction intervals based on Laplace or Student's t-distribution noise.

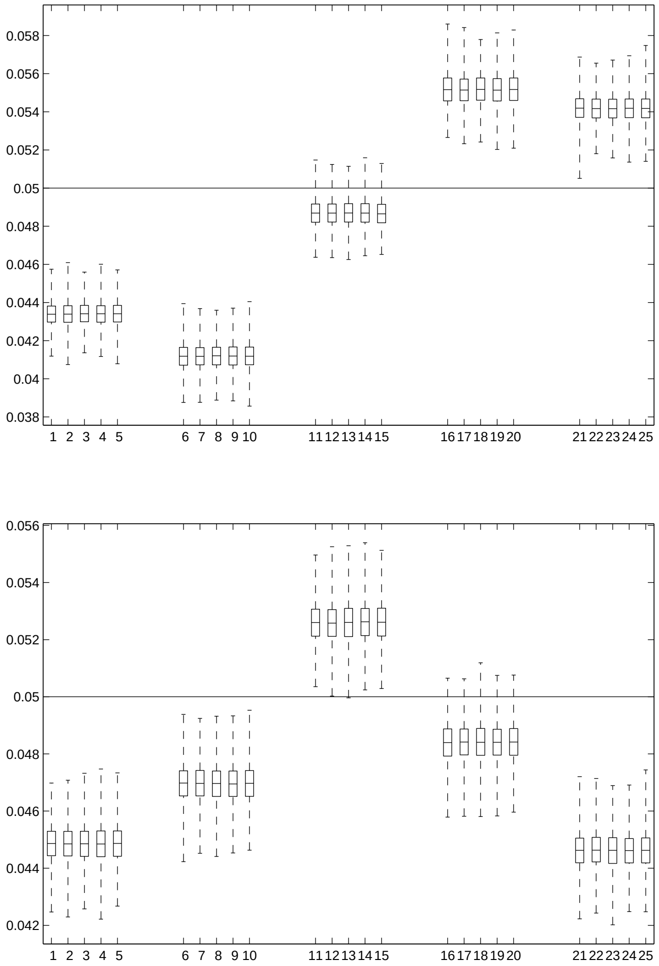


Figure 5.3: The fully conditional coverage probabilities of Gaussian (top) and Laplace (bottom) prediction intervals for $\epsilon = 5\%$

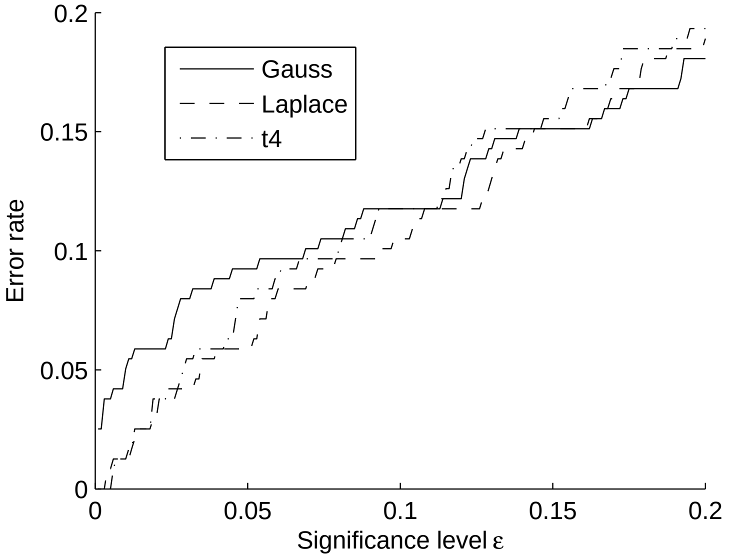


Figure 5.4: The validity plots for the ChickWeight data set, with $\epsilon \in (0, 0.2]$

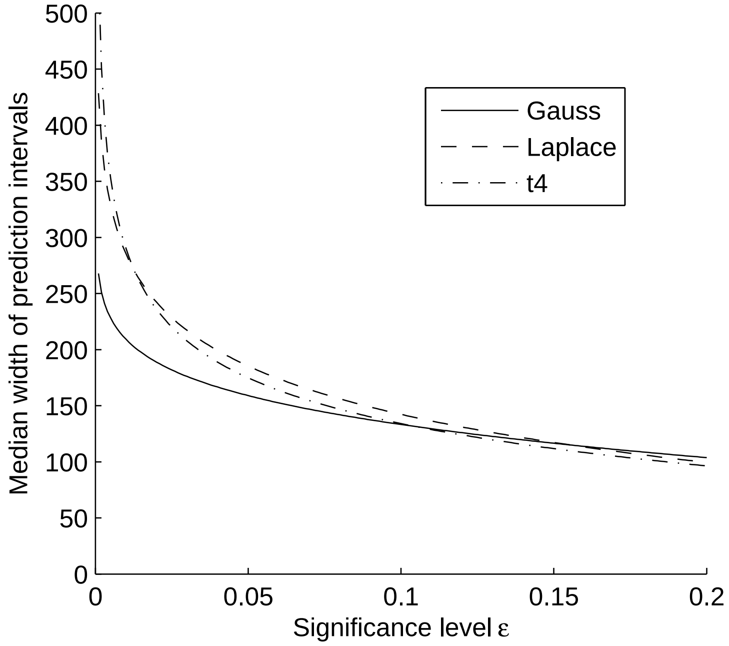


Figure 5.5: Median widths of the prediction intervals for the ChickWeight data set

Chapter 6

Conclusions and Future Work

This final chapter concludes the thesis and provides possible directions for future work.

6.1 Conclusions

This thesis was devoted to algorithms with online validity. Such algorithms allow us to complement each individual prediction with its reliability measure, and these predictions have a guarantee on the overall outcome. The majority of the thesis focuses on frameworks of confidence machines, category-based confidence machines and Venn machines; however, the last part is devoted to development of a new algorithm with online validity.

The following new developments and results were presented in this thesis.

- New implementations of confidence and Venn machines were proposed: confidence machines based on random forests, Venn machines based on random forests and Venn machines with the taxonomy derived from an SVM.

- Experimental testing of designed confidence and Venn machines was carried out. It demonstrated that proposed confidence machines based on random forests are more efficient than the known ones on mass spectrometry data (and at least as efficient on other types of data) while maintaining the property of validity. The designed Venn machines also empirically proved to be valid in the offline mode. When forced to produce singleton predictions, confidence and Venn machines based on random forests result in accuracy similar to random forest accuracy. In comparison with other algorithms with online validity, forced accuracy of all methods derived from random forests is at least as high as accuracy produced by other algorithms of the same class (i.e., confidence machines or Venn machines) and sometimes is considerably higher. In addition, all methods based on random forests proved to be robust to noise, robust to parameters of random forest construction and in the case of Venn machines — comparatively robust to the type of Venn taxonomy and the number of categories. All these characteristics make Venn machines based on random forests an attractive analytical tool.

- The designed Venn machines based on SVMs also empirically proved to be valid in the offline mode. Their performance proved to substantially depend on the type of the Venn taxonomy, kernel selection, kernel parameter and the number of categories. They may produce high accuracy, but when using taxonomy based on SVMs, one should be cautious with the choice of taxonomies and the number of categories. Thus, these methods may be not very consistent and may require tuning in order to find good parameters. Venn machines based on SVMs often produce wide prediction intervals, which can make predictions uninformative.

  However, experiments with Venn machines derived from SVMs conformed with the hypothesis that these methods perform well on the data with the number of informative peaks comparable to the half of the total number. Therefore, we could advise that these methods be applied when this requirement is expected to be satisfied.

- Algorithms with online validity for the analysis of mass spectrometry data were designed. The algorithms are a category-based confidence machine based on linear rules, a logistic Venn machine and a confidence machine in a triplet setting. These algorithms have guaranteed validity and are adapted to the needs of proteomics research.

- The designed methods were applied to experimental MALDI-TOF mass spectrometry data sets of the UKCTOCS. First, all of these algorithms allowed us to complement each individual prediction with additional information on its reliability (confidence or a probability interval).

  Second, the property of validity was empirically proved to hold true on all data sets and all algorithms. Meanwhile, in certain time slots, algorithms provided high efficiency, especially on the ovarian cancer data set. Close to the moment of diagnosis most region predictions produced by the category-based confidence machine contained one label, similarly to the output of simple predictors. As for logistic Venn machines, probability intervals produced for ovarian cancer and heart disease were more accurate than the output of a corresponding probability predictor of logistic regression. Forced accuracy of the designed algorithms was approximately the same as the accuracy of the underlying algorithms.

- Forced predictions on the UKCTOCS data sets allow us to speculate how long in advance we can output accurate predictions for ovarian and breast cancers. For the ovarian cancer data, we could make predictions with accuracy of 91.7% (or 90.2%) just before the moment of diagnosis and at least 70.6% up to 10 months in advance of the moment of diagnosis; for the breast cancer data, we could achieve accuracy of 70.4–77.8% for up to 9 months in advance of diagnosis.

- Experiments on the UKCTOCS data demonstrated that samples can be compared to each other even if they are not matched by sample collection location and time. This implies that triplets can be merged and still provide accurate predictions.

- Mass spectrometry profile peaks previously identified as carrying statistically significant information for diagnosis of ovarian and breast cancers were also pinpointed by our methods.

- A new algorithm with online validity (SPIP) was designed for linear regression statistical model. This is a model different from the standard i.i.d. assumption; however, SPIPs are exactly valid and asymptotically valid.

- The experimental studies of SPIP confirmed the property of validity on artificially generated data. We also investigated application of SPIPs to real-world data and came up with a general recommendation to use prediction intervals based on Laplace or Student's t-distribution noise.

6.2 Future Work

This research has left some questions open and raised some new questions. Here we describe possible directions of further research. We divide them into three areas corresponding to different chapters of the thesis.

6.2.1 Design of Algorithms with Online Validity

1. The designed Venn machines based on SVMs were developed for a binary classification problem only. To extend the area of application of such Venn machines, we should also define Venn taxonomy based on SVMs for multilabel classification without significantly raising computational complexity.

2. The performance of confidence and Venn machines is usually in line with the accuracy of the underlying algorithm. Meanwhile, accuracy of different simple predictors varies across different types of data, and there is no perfect simple predictor that outputs the highest accuracy on any data. Therefore, it would be beneficial to deploy new underlying algorithms in order to design new strangeness measures and Venn taxonomies to inherit their ability to perform well on certain types of data.

3. To estimate performance of designed confidence and Venn machines, we can compare them with methods which hedge individual predictions (for example, Platt's calibration [48]) when such comparison is possible.

4. Methods based on random forests and SVMs are less computationally efficient than the ones based on k-nearest neighbour method. To mitigate this disadvantage, we can attempt to improve their computational efficiency. It could be achieved by both modifying strangeness measures / Venn taxonomies (an example could be the use of out-of-bag predictions instead of leave-one-out predictions for methods based on random forests) and proposing their more efficient implementations.

6.2.2 Algorithms with Online Validity for Proteomics

1. There are general concerns about effectiveness of proteomics research. As it was mentioned in Section 4.1.3, proteomics technologies are capable of detecting only up to 20% of the protein species presented in plasma. Hence lots of information useful for early diagnosis of diseases may be at ultra-low concentrations and therefore not accessible for proteomics methods. However, there are newly developed methods based on nanotechnologies which allow users to detect single molecules of proteins. Should we have data produced by these methods, we could adjust our algorithms to the output of these methods and test their accuracy and efficiency. Since we managed to produce informative results on proteomic data, the application of designed algorithms to more complete data may allow us to achieve even better results.

2. More experimental work is required to verify applicability of designed algorithms. Application of the methods to other mass spectrometry data sets with a larger number of samples would be beneficial. In addition, for confidence machines in a triplet-like setting, we need data sets with a larger number of patients in each group (for example, groups of at least five matched patients only one of which is diseased).

6.2.3 An Algorithm with Online Validity in the Linear Regression Model

1. In this part of the thesis, we started extending the range of statistical models where we can develop algorithms with online validity, in particular, algorithms which in the online mode make errors at each step independently and with the preset probability. We considered a linear regression model rather than the standard i.i.d. assumption. Further research can be carried out in order to develop algorithms with the property of validity under other statistical assumptions.

2. We proved that SPIPs have the property of validity, however did not investigate efficiency of these predictors, which is another useful characteristic of their performance. Further research should be carried out on the efficiency of output intervals (that is, their width): we should attempt to discover theoretical bounds and obtain experimental results. These theoretical bounds could be dependent on the number of samples or the whole training set.

3. In order to estimate the risk of miscalibration when the model used in the SPIP algorithm does not correspond to the data (for example, when the noise is Guassian, but the model used in the SPIP is Laplace), ideal distributions for validity plots (Figure 5.1) can be generated and compared with experimental results. That is, we can attempt to explicitly compute the probability of an error made on data generated by probability distribution p_1 by the SPIP algorithm using probability distribution p_2 for significance level ϵ, where p_1 and p_2 are Gaussian and Laplace distribution, respectively, or vice versa. Similar ideal distributions for validity plots would be beneficial for the case when one of distributions is the Student's t-distribution since this is one of distributions we recommend to use when SPIPs are applied to real-world data.

4. When the noise distribution P is the Student's t-distribution, we should attempt to design normalisation that leads to the distribution of z_n being invariant not only of β and σ but also the number of degrees of

freedom for Student's t-distribution. If Proposition 5.1 holds true for this normalisation, it would allow us to construct prediction intervals with the property of online validity for statistical model 5.1 with Student's t-distribution noise without knowing the exact number of degrees of freedom.

5. In our work we investigated only three types of noise: Gaussian, Laplace and Student's t-distribution. However, there exist other distributions which could be potentially useful when the SPIPs are applied to real-world data, e.g.:

   - the uniform distribution with the probability distribution function $p(y) = \mathbb{I}_{[-1,1]}$;
   - the triangular distribution with the probability distribution function $p(y) = (1 - |y|)^+$;
   - the logistic distribution with the cumulative distribution function $(1 + e^{-y})^{-1}$ and the probability distribution function $p(y) = (2 + e^y + e^{-y})^{-1}$.

   Further empirical studies should be carried out using these noise distributions, in order to investigate their applicability.

Appendix A

Additional Experimental Results

In this appendix, we provide additional plots and tables for experimental results presented in the thesis.

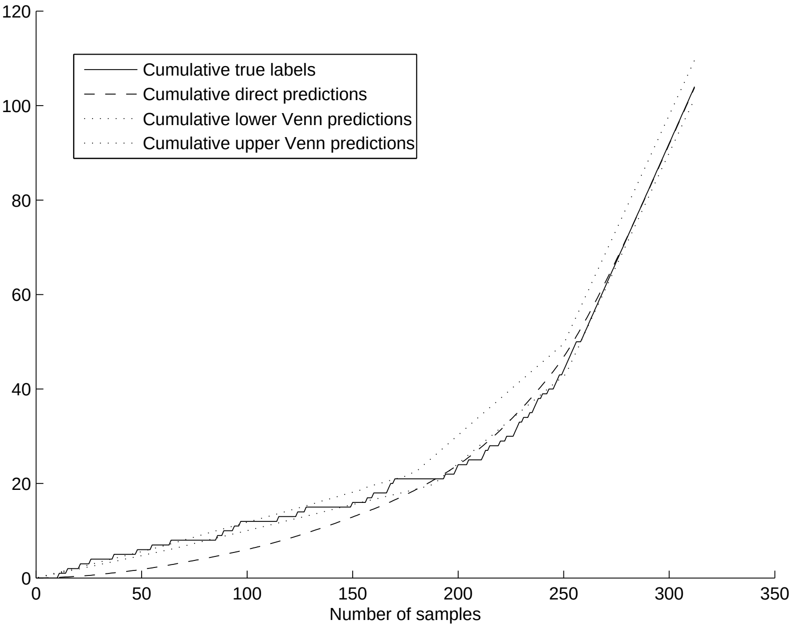


Figure A.1: Cumulative Venn and direct predictions for the ovarian cancer data (all samples)

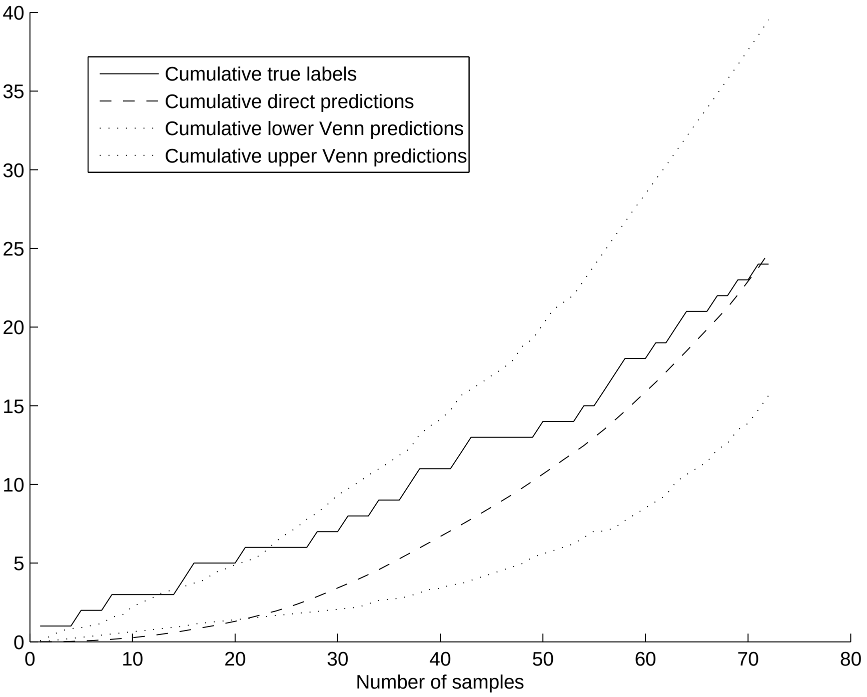


Figure A.2: Cumulative Venn and direct predictions for the breast cancer data (samples in the 5–17 month time slot)

Table A.1: Dependence of confidence machine performance on the number of features to split on at random forest nodes: forced accuracy of confidence machines CM-RF, CM-RF-1NN and CM-RF-5NN applied to the Sonar data (the number of features used in experiments in Chapter 3 is 7)

Number of features	CM-RF	CM-RF-1NN	CM-RF-5NN
1	84.6%	89.9%	86.1%
2	86.1%	88.9%	84.6%
3	84.6%	90.4%	86.1%
4	84.6%	90.4%	84.6%
5	85.1%	90.4%	86.1%
6	84.6%	89.9%	83.7%
7	84.6%	89.9%	84.6%
8	85.1%	88.0%	83.7%
9	85.1%	89.4%	84.1%
10	83.2%	87.0%	82.7%
11	83.2%	86.1%	84.1%
12	84.1%	86.1%	82.7%
13	84.1%	86.5%	84.1%
14	84.1%	85.1%	83.7%
15	82.2%	88.5%	83.2%
16	84.1%	84.6%	83.2%
17	84.1%	87.0%	83.2%
18	83.7%	84.1%	81.7%
19	84.6%	86.5%	83.2%
20	83.7%	83.7%	81.7%

Table A.2: Noise robustness testing of confidence machines: difference in accuracy, erroneous prediction rate, empty prediction rate, certain prediction rate, correct certain prediction rate and multiple prediction rate at significance level of 10% caused by injected 10% noise

	Decrease in					
Data set / Machine	accuracy	error	empty	certain	correct certain	multiple
UKOPS						
CM-1NN	0.036	0.010	0.039	-0.039	-0.010	0.000
CM-5NN	0.024	0.011	0.021	-0.021	-0.011	0.000
CM-RF	0.025	0.050	0.000	0.199	0.150	-0.199
CM-SVM (poly, 5)	-0.010	0.008	0.011	-0.011	-0.008	0.000
CM-SVM (RBF, 5)	-0.022	0.009	0.016	-0.016	-0.009	0.000
UKCTOCS OC						
CM-1NN	0.023	0.003	0.010	-0.010	-0.003	0.000
CM-5NN	0.013	0.006	0.012	-0.012	-0.006	0.000
CM-RF	0.017	0.040	0.000	0.297	0.257	-0.297
CM-SVM (poly, 5)	0.026	-0.054	-0.039	0.039	0.054	0.000
CM-SVM (RBF, 5)	0.041	0.004	0.005	-0.005	-0.004	0.000
Competition						
CM-1NN	0.029	0.007	0.013	-0.013	-0.007	0.000
CM-5NN	0.007	0.011	0.011	-0.011	-0.011	0.000
CM-RF	-0.014	0.041	0.000	0.246	0.205	-0.246
CM-SVM (poly, 5)	0.031	-0.058	-0.048	0.048	0.058	0.000
CM-SVM (RBF, 5)	0.030	0.003	0.003	-0.003	-0.003	0.000
Abdominal pain						
CM-1NN	0.083	0.011	0.019	-0.019	-0.011	0.000
CM-5NN	0.005	0.011	0.011	-0.011	-0.011	0.000
CM-RF	-0.001	0.080	0.047	0.247	0.206	-0.294
CM-SVM (poly, 5)	0.011	-0.008	-0.008	0.008	0.008	0.000
CM-SVM (RBF, 5)	-0.003	0.005	0.005	-0.005	-0.005	0.000

Table A.3: Noise robustness testing of confidence machines: difference in accuracy, erroneous prediction rate, empty prediction rate, certain prediction rate, correct certain prediction rate and multiple prediction rate at significance level of 10% caused by injected 10% noise (continued)

	Decrease in					
Data set / Machine	accuracy	error	empty	certain	correct certain	multiple
Microarray						
CM-1NN	0.040	0.003	0.010	-0.010	-0.003	0.000
CM-5NN	0.005	0.003	0.003	-0.003	-0.003	0.000
CM-RF	0.004	0.078	0.055	0.247	0.221	-0.302
CM-SVM (poly, 5)	0.024	-0.152	-0.159	0.159	0.152	0.000
CM-SVM (RBF, 5)	-0.007	0.004	0.004	-0.004	-0.004	0.000
Sonar						
CM-1NN	0.043	0.009	0.009	-0.009	-0.009	0.000
CM-5NN	0.020	0.007	0.007	-0.007	-0.007	0.000
CM-RF	0.002	0.038	0.000	0.159	0.120	-0.159
CM-SVM (poly, 5)	0.045	0.006	0.036	-0.036	-0.006	0.000
CM-SVM (RBF, 5)	0.051	0.004	0.005	-0.005	-0.004	0.000
Iris						
CM-1NN	0.073	0.009	0.011	-0.011	-0.009	0.000
CM-5NN	-0.001	0.004	0.004	-0.004	-0.004	0.000
CM-RF	0.023	0.075	0.080	0.241	0.243	-0.321
CM-SVM (poly, 5)	-0.001	0.001	-0.004	0.004	-0.001	0.000
CM-SVM (RBF, 5)	0.035	0.007	0.011	-0.011	-0.007	0.000

Table A.4: Dependence of Venn machine performance on the number of features to split on at random forest nodes: results of application of Venn machine VM-RF2A with 5 categories to the Sonar data (the number of features used in experiments in Chapter 3 is 7)

Number features	Forced accuracy	Average lower probability	Average upper probability	Average interval length
1	86.5%	0.810	0.887	0.076
2	87.0%	0.813	0.888	0.075
3	85.6%	0.814	0.875	0.061
4	82.7%	0.818	0.878	0.060
5	85.6%	0.799	0.870	0.071
6	84.1%	0.804	0.868	0.064
7	85.1%	0.807	0.870	0.062
8	86.5%	0.803	0.862	0.059
9	84.1%	0.807	0.861	0.055
10	87.0%	0.803	0.865	0.062
11	85.1%	0.807	0.857	0.051
12	86.5%	0.809	0.864	0.055
13	84.6%	0.800	0.856	0.056
14	83.2%	0.803	0.856	0.054
15	83.2%	0.799	0.850	0.052
16	83.7%	0.804	0.857	0.053
17	85.1%	0.799	0.852	0.053
18	82.7%	0.802	0.852	0.050
19	83.7%	0.800	0.850	0.051
20	82.7%	0.798	0.851	0.053

Table A.5: Performance of Venn machines on the Sonar data in the leave-one-out mode: forced accuracy, average probability interval start, end and length

					Average interval		
Venn machine	Kernel	Parameter	K'	Accuracy	start	end	length
VM-1NN				87.5%	0.87	0.88	0.01
VM-RF1				85.1%	0.81	0.85	0.04
VM-RF2A			2	86.5%	0.81	0.86	0.06
VM-RF2A			5	85.1%	0.81	0.87	0.06
VM-RF2A			10	85.6%	0.80	0.88	0.08
VM-RF2B2			1	86.1%	0.82	0.85	0.03
VM-RF2B2			3	86.5%	0.81	0.88	0.07
VM-RF2B2			5	85.1%	0.80	0.87	0.07
VM-RF2B2			7	84.1%	0.80	0.88	0.08
VM-RF3				86.5%	0.81	0.88	0.07
VM-SVM1	linear			84.6%	0.55	0.95	0.40
VM-SVM1	poly	5		55.3%	0.58	0.82	0.25
VM-SVM1	poly	10		53.4%	0.53	0.53	0.00
VM-SVM1	RBF	0.2		100.0%	0.01	1.00	0.99
VM-SVM1	RBF	1		100.0%	0.01	1.00	0.99
VM-SVM1	RBF	5		100.0%	0.41	1.00	0.59
VM-SVM2	linear		2	73.6%	0.56	0.95	0.40
VM-SVM2	linear		5	76.4%	0.60	0.97	0.38
VM-SVM2	linear		10	75.5%	0.56	0.98	0.43
VM-SVM2	poly	5	2	69.7%	0.60	0.80	0.20
VM-SVM2	poly		5	56.3%	0.51	0.82	0.31
VM-SVM2	poly		10	59.1%	0.53	0.86	0.34
VM-SVM2	poly	10	2	53.4%	0.53	0.54	0.00
VM-SVM2	poly		5	53.4%	0.53	0.54	0.00
VM-SVM2	poly		10	53.4%	0.53	0.54	0.00
VM-SVM2	RBF	0.2	2	54.3%	0.13	1.00	0.87
VM-SVM2	RBF		5	59.1%	0.15	0.96	0.81
VM-SVM2	RBF		10	54.3%	0.07	0.98	0.91
VM-SVM2	RBF	1	2	84.1%	0.38	0.98	0.59
VM-SVM2	RBF		5	78.9%	0.46	1.00	0.54
VM-SVM2	RBF		10	84.1%	0.37	1.00	0.63
VM-SVM2	RBF	5	2	82.2%	0.45	0.98	0.52
VM-SVM2	RBF		5	88.5%	0.54	0.95	0.41
VM-SVM2	RBF		10	82.2%	0.46	0.99	0.53

Table A.6: Performance of Venn machines on the Sonar data in the leave-one-out mode: forced accuracy, average probability interval start, end and length (continued)

					Average interval		
Venn machine	Kernel	Parameter	K'	Accuracy	start	end	length
VM-SVM3	linear		1	72.6%	0.22	0.98	0.75
VM-SVM3	linear		2	72.6%	0.22	0.98	0.75
VM-SVM3	linear		4	72.6%	0.22	0.98	0.75
VM-SVM3	poly	5	1	55.3%	0.57	0.84	0.27
VM-SVM3	poly		2	55.3%	0.57	0.84	0.27
VM-SVM3	poly		4	55.3%	0.57	0.84	0.27
VM-SVM3	poly	10	1	53.4%	0.53	0.54	0.00
VM-SVM3	poly		2	53.4%	0.53	0.54	0.00
VM-SVM3	poly		4	53.4%	0.53	0.54	0.00
VM-SVM3	RBF	0.2	1	53.4%	0.00	1.00	1.00
VM-SVM3	RBF		2	53.4%	0.00	1.00	1.00
VM-SVM3	RBF		4	53.4%	0.00	1.00	1.00
VM-SVM3	RBF	1	1	53.4%	0.00	1.00	1.00
VM-SVM3	RBF		2	53.4%	0.00	1.00	1.00
VM-SVM3	RBF		4	53.4%	0.00	1.00	1.00
VM-SVM3	RBF	5	1	71.6%	0.13	1.00	0.87
VM-SVM3	RBF		2	71.6%	0.13	1.00	0.87
VM-SVM3	RBF		4	71.6%	0.13	1.00	0.87

Table A.7: Accuracy of simple predictors (random forests and SVMs) on the Sonar data in the leave-one-out mode

Algorithm	Kernel	Parameter	Accuracy
Random forest			85.1%
SVM	linear		76.9%
SVM	poly	5	70.2%
SVM	poly	10	48.6%
SVM	RBF	0.2	53.4%
SVM	RBF	1	54.3%
SVM	RBF	5	85.6%

Table A.8: Performance of Venn machines on the UKCTOCS OC data in the leave-one-out mode: forced accuracy, average probability interval start, end and length. Accuracy of corresponding simple predictors (random forests and SVMs) is provided for comparison.

				Average interval		
Algorithm	Kernel, parameter	K'	Accuracy	start	end	length
VM-1NN			66.7%	0.66	0.67	0.01
Random forest			*84.9%*			
VM-RF1			84.9%	0.83	0.85	0.02
VM-RF2A		2	76.0%	0.69	0.73	0.04
VM-RF2A		5	83.7%	0.80	0.85	0.05
VM-RF2A		10	83.0%	0.80	0.86	0.07
SVM	*linear*		*77.6%*			
VM-SVM2	linear	2	73.1%	0.56	0.83	0.26
VM-SVM2	linear	5	78.2%	0.64	0.93	0.29
VM-SVM2	linear	10	78.2%	0.63	0.94	0.31
SVM	*poly, 5*		*59.0%*			
VM-SVM2	poly, 5	2	66.7%	0.62	0.69	0.07
VM-SVM2	poly, 5	5	60.3%	0.58	0.74	0.15
VM-SVM2	poly, 5	10	62.8%	0.55	0.77	0.22
SVM	*RBF, 5*		*78.5%*			
VM-SVM2	RBF, 5	2	73.4%	0.46	0.87	0.41
VM-SVM2	RBF, 5	5	77.6%	0.37	0.98	0.61
VM-SVM2	RBF, 5	10	56.7%	0.29	1.00	0.71

Table A.9: Performance of Venn machines on the UKCTOCS BC data in the leave-one-out mode: forced accuracy, average probability interval start, end and length. Accuracy of corresponding simple predictors (random forests and SVMs) is provided for comparison.

				Average interval		
Algorithm	Kernel, parameter	K'	Accuracy	start	end	length
VM-1NN			66.7%	0.66	0.67	0.01
Random forest			*66.0%*			
VM-RF1			66.0%	0.66	0.68	0.02
VM-RF2A		2	66.7%	0.65	0.68	0.03
VM-RF2A		5	62.3%	0.62	0.70	0.07
VM-RF2A		10	66.0%	0.61	0.74	0.13
SVM	*linear*		*53.1%*			
VM-SVM2	linear	2	53.7%	0.49	0.84	0.35
VM-SVM2	linear	5	42.0%	0.39	0.94	0.55
VM-SVM2	linear	10	48.8%	0.38	0.95	0.58
SVM	*poly, 5*		*46.3%*			
VM-SVM2	poly, 5	2	66.7%	0.64	0.68	0.03
VM-SVM2	poly, 5	5	66.7%	0.62	0.68	0.06
VM-SVM2	poly, 5	10	61.1%	0.60	0.72	0.12
SVM	*RBF, 5*		*64.2%*			
VM-SVM2	RBF, 5	2	58.6%	0.41	0.92	0.51
VM-SVM2	RBF, 5	5	42.0%	0.19	1.00	0.81
VM-SVM2	RBF, 5	10	32.1%	0.06	1.00	0.94

Table A.10: Performance of Venn machines on the UKCTOCS HD data in the leave-one-out mode: forced accuracy, average probability interval start, end and length. Accuracy of corresponding simple predictors (random forests and SVMs) is provided for comparison.

				Average interval		
Algorithm	Kernel, parameter	K'	Accuracy	start	end	length
VM-1NN			54.0%	0.67	0.67	0.00
Random forest			*71.7%*			
VM-RF1			72.0%	0.70	0.72	0.02
VM-RF2A		2	66.7%	0.66	0.68	0.02
VM-RF2A		5	72.7%	0.68	0.72	0.04
VM-RF2A		10	72.0%	0.68	0.74	0.06
SVM	*linear*		*71.7%*			
VM-SVM2	linear	2	62.4%	0.63	0.69	0.06
VM-SVM2	linear	5	70.1%	0.70	0.78	0.08
VM-SVM2	linear	10	70.2%	0.68	0.80	0.12
SVM	*poly, 5*		*65.4%*			
VM-SVM2	poly, 5	2	65.1%	0.49	0.84	0.36
VM-SVM2	poly, 5	5	66.0%	0.37	0.89	0.51
VM-SVM2	poly, 5	10	47.1%	0.29	0.89	0.60
SVM	*RBF, 5*		*70.9%*			
VM-SVM2	RBF, 5	2	64.4%	0.54	0.83	0.29
VM-SVM2	RBF, 5	5	64.4%	0.52	0.95	0.43
VM-SVM2	RBF, 5	10	65.8%	0.49	0.95	0.47

Table A.11: Performance of Venn machines on the Competition data in the leave-one-out mode: forced accuracy, average probability interval start, end and length. Accuracy of corresponding simple predictors (random forests and SVMs) is provided for comparison.

				Average interval		
Algorithm	Kernel, parameter	K'	Accuracy	start	end	length
VM-1NN			75.8%	0.75	0.76	0.01
Random forest			*83.7%*			
VM-RF1			84.3%	0.81	0.84	0.04
VM-RF2A		2	84.3%	0.82	0.84	0.02
VM-RF2A		5	75.8%	0.79	0.84	0.05
VM-RF2A		10	81.7%	0.77	0.86	0.08
SVM	*linear*		*88.2%*			
VM-SVM2	linear	2	23.5%	0.24	1.00	0.76
VM-SVM2	linear	5	77.1%	0.45	0.94	0.49
VM-SVM2	linear	10	40.5%	0.22	1.00	0.78
SVM	*poly, 5*		*49.0%*			
VM-SVM2	poly, 5	2	0.0%	0.50	0.51	0.01
VM-SVM2	poly, 5	5	0.0%	0.50	0.51	0.01
VM-SVM2	poly, 5	10	0.0%	0.50	0.51	0.01
SVM	*RBF, 5*		*74.5%*			
VM-SVM2	RBF, 5	2	11.1%	0.12	1.00	0.88
VM-SVM2	RBF, 5	5	79.7%	0.47	0.97	0.49
VM-SVM2	RBF, 5	10	39.9%	0.11	1.00	0.89

Table A.12: Performance of Venn machines on three-class data sets in the leave-one-out mode: forced accuracy, average probability interval start, end and length. Accuracy of a bare random forest is also provided

				Average interval		
Data	Machine	K'	Accuracy	start	end	length
UKOPS	VM-1NN		58.6%	0.58	0.59	0.01
	VM-RF1		84.3%	0.81	0.84	0.03
	VM-RF2B2	1	84.3%	0.81	0.84	0.04
	VM-RF2B2	2	77.1%	0.78	0.84	0.06
	VM-RF2B2	3	83.7%	0.77	0.86	0.09
	Random forest		*72.6%*			
7 biomarkers	VM-1NN		57.2%	0.56	0.58	0.01
	VM-RF1		73.1%	0.72	0.76	0.04
	VM-RF2B2	1	75.2%	0.72	0.76	0.04
	VM-RF2B2	2	74.9%	0.72	0.76	0.04
	VM-RF2B2	3	74.6%	0.71	0.76	0.04
	Random forest		*74.6%*			
Abdominal pain	VM-1NN		89.3%	0.88	0.89	0.01
	VM-RF1		92.0%	0.89	0.92	0.04
	VM-RF2B2	1	92.0%	0.89	0.92	0.03
	VM-RF2B2	2	91.3%	0.88	0.93	0.05
	VM-RF2B2	3	92.0%	0.88	0.94	0.06
	Random forest		*91.7%*			
Microarray	VM-1NN		90.0%	0.89	0.90	0.01
	VM-RF1		92.0%	0.88	0.92	0.03
	VM-RF2B2	1	92.0%	0.89	0.92	0.03
	VM-RF2B2	2	92.0%	0.88	0.92	0.04
	VM-RF2B2	3	91.8%	0.88	0.93	0.05
	Random forest		*92.0%*			
Iris	VM-1NN		55.8%	0.55	0.56	0.01
	VM-RF1		96.0%	0.93	0.96	0.03
	VM-RF2B2	1	96.0%	0.94	0.96	0.02
	VM-RF2B2	2	95.3%	0.93	0.96	0.04
	VM-RF2B2	3	95.3%	0.92	0.96	0.04
	Random forest		*95.3%*			

Table A.13: Noise robustness testing of Venn machines on two-class data sets except for the Competition data: difference in accuracy, average probability interval start, end and length due to injected 10% noise (only those experiments are shown which resulted in forced accuracy higher than the majority rate)

			Decrease in			
				average interval		
Data/Machine	Kernel, parameter	K'	accuracy	start	end	length
Sonar						
VM-1NN			0.03	0.05	0.05	0.00
VM-RF1			0.03	0.07	0.07	0.00
VM-RF2A		2	0.04	0.08	0.09	0.02
VM-RF2A		5	0.000	0.065	0.069	0.005
VM-RF2A		10	0.034	0.077	0.057	-0.020
VM-SVM2	linear, 5	2	0.04	0.00	0.08	0.08
VM-SVM2	linear, 5	5	0.04	0.04	0.06	0.02
VM-SVM2	linear, 5	10	0.06	0.00	0.05	0.05
VM-SVM2	rbf, 5	2	0.06	0.07	0.02	-0.05
VM-SVM2	rbf, 5	5	0.06	0.08	-0.02	-0.11
VM-SVM2	rbf, 5	10	0.01	0.08	0.01	-0.07
UKCTOCS OC						
VM-RF1			0.01	0.07	0.07	0.00
VM-RF2A		2	0.02	0.03	0.04	0.00
VM-RF2A		5	0.01	0.07	0.06	-0.01
VM-RF2A		10	0.00	0.08	0.07	-0.02
VM-SVM2	linear, 5	2	0.02	-0.03	0.02	0.06
VM-SVM2	linear, 5	5	0.03	0.01	0.05	0.04
VM-SVM2	linear, 5	10	0.05	0.02	0.06	0.04
VM-SVM2	rbf, 5	2	0.02	0.05	-0.04	-0.09
VM-SVM2	rbf, 5	5	0.04	0.15	-0.01	-0.16
VM-SVM2	linear, 5	5	0.01	0.05	0.05	0.00
UKCTOCS HD						
RF1			0.01	0.06	0.06	0.00
RF2A		5	0.01	0.06	0.06	0.00
RF2A		10	0.02	0.06	0.06	0.00
VM-SVM2	linear, 5	5	0.01	0.05	0.05	0.00
VM-SVM2	linear, 5	10	0.01	0.03	0.06	0.03

Table A.14: Noise robustness testing of Venn machines on the Competition data: difference in accuracy, average probability interval start, end and length due to injected 10% noise (only those experiments are shown which resulted in forced accuracy higher than the majority rate)

			Decrease in			
				average interval		
Data/Machine	Kernel, parameter	K'	accuracy	start	end	length
Competition						
VM-1NN			0.03	0.06	0.06	0.00
VM-RF1			0.01	0.03	0.04	0.00
VM-RF2A		2	0.00	0.03	0.03	0.00
VM-RF2A		5	-0.04	0.05	0.04	-0.01
VM-RF2A		10	-0.01	0.05	0.04	-0.01
VM-SVM2	linear, 5	5	0.00	0.10	-0.05	-0.15
VM-SVM2	rbf, 5	5	0.04	0.10	-0.03	-0.13

Table A.15: Noise robustness testing of Venn machines on three-class data sets: difference in accuracy, average probability interval start, end and length due to injected 10% noise

		Decrease in			
			average interval		
Data/Machine	K'	accuracy	start	end	length
UKOPS					
VM-1NN		0.02	0.06	0.06	0.00
VM-RF1		0.00	0.05	0.04	-0.01
VM-RF2B2	1	0.00	0.05	0.06	0.01
VM-RF2B2	2	-0.03	0.04	0.04	0.00
VM-RF2B2	3	0.01	0.05	0.04	-0.01
7 biomarkers					
VM-1NN		0.01	0.02	0.02	0.00
VM-RF1		0.00	0.07	0.07	0.00
VM-RF2B2	1	0.01	0.07	0.07	0.00
VM-RF2B2	2	0.01	0.08	0.08	0.00
VM-RF2B2	3	0.00	0.05	0.06	0.01
Abdominal pain					
VM-1NN		0.07	0.13	0.13	0.00
VM-RF1		0.00	0.08	0.09	0.01
VM-RF2B2	1	0.00	0.09	0.08	-0.01
VM-RF2B2	2	0.00	0.08	0.08	0.00
VM-RF2B2	3	0.00	0.08	0.08	0.00
Microarray					
VM-1NN		0.06	0.09	0.09	0.00
VM-RF1		0.03	0.07	0.07	0.00
VM-RF2B2	1	0.00	0.10	0.09	0.00
VM-RF2B2	2	0.00	0.08	0.08	0.00
VM-RF2B2	3	0.01	0.09	0.09	0.00
Iris					
VM-1NN		0.07	0.11	0.11	0.00
VM-RF1		0.04	0.13	0.13	0.00
VM-RF2B2	1	0.03	0.12	0.11	-0.01
VM-RF2B2	2	0.02	0.10	0.10	-0.01
VM-RF2B2	3	0.03	0.13	0.12	-0.01

Table A.16: M/z-values of statistically significant peaks for the UKCTOCS data sets

Data set	Peak number	M/z-value
Ovarian cancer	2	7772.1
	3	9297.8
Breast cancer	19	6637.8
Heart disease	4	4211.1
	6	4055.0
	7	5338.3

Appendix B

Triplet Analysis of the UKCTOCS OC Data Set

This appendix presents the triplet analysis of the UKCTOCS OC data. The description of the original mass spectrometry data and applied pre-processing is given in Section 4.2. The triplet analysis was carried out in a triplet setting: we took into account that each case sample is accompanied by two controls taken from healthy individuals, and these controls were matched on patient age and on when and where sample were collected.

Let us recall that each sample is represented by intensities of 67 most common profile peaks $I(1), \ldots, I(67)$ and a CA125 measurement C. Each triplet τ is assigned time to diagnosis $T(\tau) > 0$, the time to diagnosis confirmed by histology/cytology for the case patient in the group.

B.1 Problem Statement

We consider different time slots as described in Section 4.3.3. The slots are six months wide and finish $t = 0, 1, 2, \ldots$ months in advance of the moment of diagnosis. In each of these time slots, we consider S_t, the set of triplets of measurements taken t months before the diagnosis, or as little earlier as possible and not exceeding $t+6$ months. Two triplets with case measurements taken from the same patient are not allowed to be used in the same set because they can be dependent on each other. Hence, if there are two measurement taken from the same patient which fall in the time slot, only the latest one is considered, together with corresponding controls.

We will use a rather limited class of rules for *triplet classification*, i.e., identification of the labelled sample within a triplet. These will be the simplest linear combinations 4.1 of CA125 and one peak with $h = 2$, n_1 corresponding to CA125 levels, n_2 corresponding to a peak number.

Therefore, each classification rule can be specified by three numbers (n_2, w_1, w_2), which are a peak number $n_2 \in P$ (P is a set of peak numbers and can be equal to $\{1, \ldots, 67\}$ or one peak number) and weights $w_1 \in W_1 = \{0, 0.5, 1, 2\}$, $w_2 \in W_2 = \{-1, 0, 1\}$. For each triplet, the classification rule (n_2, w_1, w_2) assigns a 'case' label to the sample with the largest value of

$$w_1 \log C + w_2 \log I(n_2)\,, \tag{B.1}$$

where C and $I(n_2)$ are the CA125 level and the intensity of peak n_2, respectively. The logarithms are taken to remove the arbitrary units of measurements.

If $w_1 \neq 0$ (that is, CA125 is taken into account), the rule has an equivalent form: $\log C + \frac{w_2}{w_1} \log I(n_2)$. Thus, we are looking for a term additional to CA125, w_2 is responsible for the possible sign of its influence (whether it is expected to be larger at cases or at controls), and $1/w_1$ determines to what degree it is taken into account.

Let $\mathrm{err}(\tau; n_2, w_1, w_2)$ be 0 if (n_2, w_1, w_2) correctly identifies the labelled sample in a triplet τ and 1 otherwise (in the case of a wrong classification), and

$$\mathrm{Err}(S; n_2, w_1, w_2) := \sum_{\tau \in S} \mathrm{err}(\tau; n_2, w_1, w_2)$$

stand for the number of errors made by (n_2, w_1, w_2) on a set S of triples. As our test statistic we take the pair (E_0, n_0) of the least number of errors and the number of the most frequent peak where it is achieved. Formally,

$$E_0 := \min_{n_2 \in P, w_1 \in W_1, w_2 \in W_2} \mathrm{Err}(S_t; n_2, w_1, w_2)\,,$$

$$n_0 := \min\{n_2 : \exists w_1 \exists w_2\, \mathrm{Err}(S_t; n_2, w_1, w_2) = E_0\}\,.$$

Pairs (E_0, n_0) are ordered lexicographically, and n_0 is added to break ties: if two rules lead to the same error rate, the rule involving a more frequent peak has priority when we calculate p-values.

B.2 Summary of the Main Findings

For each set of samples S_t, the best pair (E_0, n_0) was selected for certain sets of parameters P, W_1 and W_2 as described above. Tests were designed in order to check the significance of our findings. The p-values given by these tests for the set of all 67 peaks ($P = \{1, \ldots, 67\}$) and separately for peak 2 ($P = \{2\}$) and peak 3 ($P = \{3\}$) are represented in Table B.1. The tests check the null hypothesis that the assignment of labels within triplets is independent of the information contained in CA125 levels and intensities of certain peaks: all peaks, peak 2 only or peak 3 only. The methodology of p-value calculation is described in detail in Section B.3.

The first column of Table B.1 shows the time to diagnosis t, which is the most recent end-point of the time window. The second column shows the size of S_t — the number of samples in the considered 6-month time slot. The other columns represent p-values for the set of all peaks (these p-values do not require adjustment) and adjusted p-values for peak 2 and peak 3 separately.

The table demonstrates that CA125 and 67 peak intensities allow us to reject the null hypothesis at significance level of 5% for up to 15 months with a single exception of month 12, which still has a p-value less than 6%, while the information contained in CA125 and peak 2 or CA125 and peak 3 provide significant p-values for detection up to 15 and 13 months before diagnosis, respectively.

B.3 Statistical Analysis of All Peaks

In order to check the robustness of our triplet classification using mass spectrometry profile peaks, we designed three types of statistical tests which reject the following null hypotheses about classification of S_t:

Table B.1: Summary of p-values for triplet classification of the UKCTOCS OC

t	$\lvert S_t \rvert$	p-values for all peaks	Adjusted p-values for peak 2	Adjusted p-values for peak 3
0	68	0.0001	0.001	0.001
1	56	0.0001	0.001	0.001
2	47	0.0001	0.001	0.001
3	36	0.0001	0.001	0.001
4	27	0.0001	0.001	0.001
5	23	0.0006	0.004	0.007
6	20	0.0028	0.010	0.046
7	17	0.0141	0.017	0.098
8	17	0.0019	0.020	0.020
9	20	0.0076	0.010	0.009
10	28	0.0003	0.001	0.001
11	28	0.0042	0.008	0.004
12	28	0.0585	0.033	0.049
13	30	0.0168	0.007	0.015
14	25	0.0304	0.015	0.301
15	20	0.0464	0.022	0.577
16	10	0.4101	5.165	5.979

1. The null hypothesis that assignment of labels within triplets is independent of CA125 and peak intensities.

2. The null hypothesis that assignment of labels within triplets is independent of CA125 levels.

3. The null hypothesis that when assigning labels within triplets, pairs (label, CA125) are independent of peak intensities, that is, peak intensities do not contain any information useful to improve the predictive ability of CA125.

The detailed explanation of corresponding p-value calculation and their meaning is given below.

B.3.1 Main p-values

The *main p-value* (or just 'p-value') is based on the null hypothesis that the assignment of the case label within each triplet in S_t is independent of the information contained in CA125 and all mass spectrometry peaks.

We calculated these and all other p-values experimentally, using the Monte-Carlo method. Suppose $P = \{1, \dots, 67\}$ (which implies that any peak can complement CA125 in a linear rule); $E_0 := \min_{n_2 \in P, w_1 \in W_1, w_2 \in W_2} \mathrm{Err}(S_t; n_2, w_1, w_2)$, that is the minimum number of errors occurring in prediction by CA125 and one of the peak intensities, and $n_0 := \min\{n_2 : \exists w_1 \exists\, w_2 \mathrm{Err}(S_t; n_2, w_1, w_2) = E_0\}$, the peak with the highest commonality that can provide this minimum error rate.

The statistic equal to the pair (E', n') calculated the same way as (E_0, n_0) is collected for a large number N (we used $N = 10^4$) of times on the same data set with randomly reassigned case labels.

We then calculate the number Q of times the statistic is as good as or better than the statistic (E_0, n_0) computed from the true labels, where 'as good as or better than' is understood in the sense of the lexicographical order: $(E', n') \leq (E_0, n_0)$ means that either $E' < E_0$ or $E' = E_0$ and $n' \leq n_0$. The p-value is then estimated as $(Q + 1)/(N + 1)$.

Algorithm 2 Main p-value calculation

```
Input: $t$, time to diagnosis.
Input: $N = 10^4$, number of trials.
$E_0 := \min_{n_2, w_1, w_2} \mathrm{Err}(S_t; n_2, w_1, w_2)$
$n_0 := \min\{n_2 : \exists w_1 \exists w_2 \mathrm{Err}(S_t; n_2, w_1, w_2) = E_0\}$
$Q := 0$
for $j := 1, \dots, N$ do
  Assign the case label to a randomly chosen sample in each triplet in $S_t$
  Recalculate $E' := \min_{n_2 \in P, w_1 \in W_1, w_2 \in W_2} \mathrm{Err}(S_t; n_2, w_1, w_2)$
  Recalculate $n' := \min\{n_2 : \exists w_1 \exists w_2 \mathrm{Err}(S_t; n_2, w_1, w_2) = E'\}$
  if $(E', n') \leq (E_0, n_0)$ then
    $Q := Q + 1$
  end if
end for
Output: $(Q + 1)/(N + 1)$ as the $p$-value.
```

B.3.2 CA125 p-values

The previous p-value demonstrates significance of predictions provided by CA125 levels in combination with peak intensities. To compare discriminating ability of the combination of CA125 and peaks with the well-known benchmark, CA125 on its own, we need to calculate *CA125 p-values.* This test checks the null hypothesis that labels at S_t are independent of CA125 levels.

These p-values can be calculated theoretically as follows. Suppose $E_0 = \mathrm{Err}(S_t; n_2, w_1, w_2)$. In this case, random prediction leads to 1/3 probability to guess the correct result in each of $|S_t|$ independent cases. Probability to make at most E_0 errors by chance is

$$\text{CA125 } p\text{-value} = \sum_{i=0}^{E_0} (2/3)^i (1/3)^{|S_t|-i} C^i_{|S_t|}\,, \tag{B.2}$$

where $C^k_n = \frac{n!}{k!(n-k)!}$ is the number of combinations of k elements without repetitions out of n elements.

Formula B.2 provides theoretical calculation of CA125 p-values. However, we will again use the Monte-Carlo method. Suppose $E_0 = \mathrm{Err}(S_t; 1, 1, 0)$, that is, the number of errors occurring in prediction by CA125 only. The statistic equal to number of errors is collected for $N = 10^4$ times on the same data set with randomly reassigned case labels, and then the algorithm counts the number Q of times the statistic is as good as or better than the statistic E_0 computed from the true labels. The p-value is then estimated as $(Q + 1)/(N + 1)$.

B.3.3 Conditional p-values

In this study we also introduced *conditional p-values*, in addition to main p-values and CA125 p-values that had been considered in [26].

Suppose that the main p-value is significant. That means that CA125 with mass spectrometry profile peaks are able to discriminate between controls and cases. In this case we wish to separate contributions of CA125 and mass spectrometry profile peaks. This can be achieved by the use of conditional p-values.

Algorithm 3 CA125 p-value calculation

```
**Input:** $t$, time to diagnosis.
**Input:** $N = 10^4$, number of trials.
$E_0 := \mathrm{Err}(S_t; 1, 1, 0)$
$Q := 0$
**for** $j := 1, \ldots, N$ **do**
  Assign the case label to a randomly chosen sample in each triplet in $S_t$
  Recalculate $E' := \mathrm{Err}(S_t; 1, 1, 0)$
  **if** $E' \leq E_0$ **then**
    $Q := Q + 1$
  **end if**
**end for**
**Output:** $(Q + 1)/(N + 1)$ as the $p$-value.
```

The null hypothesis to consider is the independence between a pair (label, C) and $(I(1), \ldots, I(67))$ as opposed to the main p-value hypothesis that can be interpreted as the independence between a label and $(C, I(1), \ldots, I(67))$.

The only difference from the computation of the main p-values is the following. At each loop step instead of assigning the case label randomly, we generate a random permutation of three elements for each triplet and apply the permutation to both labels ('case', 'control', 'control') and CA125 levels of triplet measurements.

Neither of p-values described above require adjustment. Main p-values as well as conditional p-values do not require any adjustment as the numbers of errors are calibrated by the Monte-Carlo procedure taking into account all the rules at the same stage. CA125 p-value does not require any adjustment, since in this case there is no set of rules to select from.

B.3.4 Experimental Results

Given that we have only a limited number of samples, we considered 6-month time slots starting in different months. For each of these slots we checked hypotheses of random label distribution, calculating p-values described above, and looked for certain peaks that carry the most useful information for discrimination between cancer and healthy samples. The initial results of the

Algorithm 4 Conditional p-value calculation

```
Input: $t$, time to diagnosis.
Input: $N = 10^4$, number of trials.
$E_0 := \min_{n_2,w_1,w_2} \operatorname{Err}(S_t; n_2, w_1, w_2)$
$n_0 := \min\{n_2 : \exists w_1 \exists w_2 \operatorname{Err}(S_t; n_2, w_1, w_2) = E_0\}$
$Q := 0$
for $j := 1, \dots, N$ do
  for each triplet in $S_t$ do
    Consider a random permutation $s : \{1, 2, 3\} \to \{1, 2, 3\}$
    Apply $s$ to the labels (case, control, control) of this triple
    Apply the same $s$ to the CA125 values $(C_1, C_2, C_3)$ of this triple
  end for
  Recalculate $E' := \min_{n_2 \in P, w_1 \in W_1, w_2 \in W_2} \operatorname{Err}(S_t; n_2, w_1, w_2)$
  Recalculate $n' := \min\{n_2 : \exists w_1 \exists w_2 \operatorname{Err}(S_t; n_2, w_1, w_2) = E'\}$
  if $(E', n') \leq (E_0, n_0)$ then
    $Q := Q + 1$
  end if
end for
Output: $(Q + 1)/(N + 1)$ as the $p$-value.
```

analysis are represented in Table B.2.

The first and second columns of Table B.2 are the same as in Table B.1. Columns 3 and 4 represent analysis results for prediction with CA125 only. Columns 5–9 show results for prediction with CA125 in combination with the set of all peaks.

The third column provides the number E_C of errors made on the triplets S_t by the classification rule $\log C$, i.e., classification by CA125 only.

The fourth column (CA125 p-values) is the measure of significance for this result, the probability to obtain an equal or even more extreme result, by chance doing classification at random. Recall that the expected probability of error in triplet classification is 2/3, so misclassifying 16 of 30 triplets (as for $t = 13$) is not significant (p-value = 9%) but still better than random. The results for classification using CA125 only are significant for up to 9 months.

The fifth column E_{C,n_2} gives the best quality (the smallest number of errors) which may be achieved by a classification rule from our list. The following rules are considered: classification by $w_1 \log C + w_2 \log I(n_2)$ where $n_2 \in \{1, \dots, 67\}$,

Table B.2: Initial statistical analysis of the UKCTOCS OC. The columns are: time t to diagnosis in months; the number $|S_t|$ of cases with measurement taken between t and $t+6$ months before diagnosis; the number E_C of errors when classifying the triplets in S_t with CA125 alone; CA125 p-value; the minimal number E_{C,n_2} of errors when classifying with CA125 (taken with weight $w_1 \in \{0, 1/2, 1, 2\}$) and intensity $I(n_2)$ of a peak (taken with weight $w_2 \in \{-1, 0, 1\}$); number n_2 of this peak; w_2/w_1; the main p-value for overall significance of this result; the conditional p-value for significance of non-CA125 component.

		CA125 only		CA125 and all peaks				
		E_C	CA125	E_{C,n_2}	n_2	w_2/w_1	Main	Conditional
t	$\|S_t\|$		p-value				p-value	p-value
0	68	2	0.0001	1	01	$+1$	0.0001	0.3164
1	56	4	0.0001	2	07	$+1$	0.0001	0.2446
2	47	6	0.0001	3	15	-2	0.0001	0.1795
3	36	8	0.0001	4	15	-2	0.0001	0.0746
4	27	7	0.0001	4	15	-2	0.0001	0.6734
5	23	7	0.0008	4	15	-1	0.0006	0.4885
6	20	6	0.0010	4	15	-1	0.0028	0.6973
7	17	6	0.0071	4	01	-1	0.0141	0.6034
8	17	5	0.0021	3	01	-1	0.0019	0.1523
9	20	7	0.0042	5	02	-1	0.0076	0.4497
10	28	14	0.0503	6	03	-1	0.0003	0.0013
11	28	15	0.1028	8	03	-1	0.0042	0.0078
12	28	17	0.3164	10	02	-2	0.0585	0.0658
13	30	16	0.0895	10	02	-2	0.0168	0.0428
14	25	16	0.4661	8	02	-2	0.0304	0.0206
15	20	13	0.5211	6	02	-2	0.0464	0.0609
16	10	6	0.4406	2	67	$+1/0$	0.4101	0.5066

w_1 is any of $0, 1/2, 1, 2$, w_2 is 1, 0 or -1. Equivalent form for the rule is $\log C + (w_2/w_1) \log I(n_2)$ where w_2/w_1 is any of $-1/0$, -2, -1, $-1/2$, 0, 1/2, 1, 2, 1/0. Notations 1/0 and $-1/0$ mean that the peak $I(n_2)$ is used alone without CA125. The sixth and seventh columns of the table are n_2 and w_2/w_1 for the best classification rule.

Experiments show that any result can be more or less improved in such a way (E_{C,n_2} is less than E_C), but we need to check that this is not obtained by chance, due to large choice of classification rules. Moreover, we need to check both the probability to obtain low error rate E_{C,n_2} by chance and the probability to decrease it from E_C to E_{C,n_2} by chance. Two types of p-values answer these two questions. Main p-values measure the chance to obtain the error rate not worse than E_{C,n_2} at random, and conditional p-values the chance to get the error equal to or less than E_{C,n_2} if peak intensities are reshuffled, but the CA125 value is real. Actually, conditional p-values are much more important for us as we know in advance that CA125 is a useful biomarker; therefore, we are mainly interested in whether addition of anything else to it may lead to statistically significant improvement.

Main and conditional p-values are represented in the last two columns of Table B.2. Main p-values are also shown in the summary table, Table B.1. From 'Main p-value' column, we can see that the null hypothesis can be rejected at level 5% for up to 15 months (with the only exception being 12 months where it is about 6%). This contrasts with using CA125 alone ('CA125 p-value' column), which produces significant results for up to 9 months. Conditional p-values show that contribution of the added mass spectrometry peak becomes essential only from 10 to 15 months, so we should pay more attention to the best quality rules for these months, that is, peaks 2 and 3 with corresponding coefficients w_1 and w_2.

B.4 Statistical Analysis of Peaks 2 and 3

In this section, we check significance of peaks 2 and 3 identified as the most informative. Thus, we check whether a specific peak (2 or 3) contains some information useful to improve triplet classification in comparison with the use of CA125. These peaks are plotted in Figure B.3.

Main p-values and conditional p-values presented before are adjusted to this problem. Main p-values are given by the test checking the null hypothesis that assignment of labels within triplets is independent of CA125 and peak 2 (3). Conditional p-values are calculated by the test checking the null hypothesis that, when assigning labels within triplets, pairs (label, CA125) are independent of peak 2 (3).

The only difference in calculation of these p-values from Algorithms 2 and 4 is the set of rules we are selecting from. In this case, we consider the following sets of parameters: P is peak 2 (or peak 3), W_1 and W_2 are the same ($\{0, 0.5, 1, 2\}$ and $\{-1, 0, 1\}$, respectively).

B.4.1 Experimental Results

Tables B.3 and B.4 represent the results for prediction by CA125 and peak 2 or CA125 and peak 3, respectively. Error rates and p-values for prediction using CA125 only are included in the beginning of the tables for comparison (columns 3 and 4). 'Rule' columns show the best rules selected: CA in this columns means $\log C$, p2 means $\log I(2)$, p3 means $\log I(3)$. Three last columns represent error rates for the best rule, main p-values and conditional p-values.

Conditional p-values show that improvement achieved by adding information from peak 2 or 3 is not significant up to 9 months as CA125 itself is good for separation, and it cannot be improved considerably by adding a peak-dependent term.

The table demonstrates that in time slots finishing in 10–16 months, main p-values and conditional p-values are similar, hence significant or insignificant at the same time. This confirms that possible improvement in the predictive ability of CA125 with a peak is achieved due to information contained in this peak.

Main and conditional p-values shown in the table require adjustment. When we adjust by 10 peaks, the threshold for significance is $0.05/10 = 0.005$ and thus the results are statistically significant for up to 15 months (with peak 2) and 13 months (with peak 3), respectively. Main p-values for peak 2 and peak

Table B.3: Experimental results for triplet classification with peak 2

		CA125 only		CA125 and peak 2			
t	$\|S_t\|$	Errors	p-value	Rule	Errors	p-value	Cond. p-value
0	68	2	0.0001	2CA−p02	2	0.0001	1.0000
1	56	4	0.0001	2CA−p02	4	0.0001	1.0000
2	47	6	0.0001	2CA−p02	5	0.0001	0.5295
3	36	8	0.0001	2CA−p02	7	0.0001	0.8340
4	27	7	0.0001	2CA−p02	6	0.0001	0.9297
5	23	7	0.0008	CA−p02	6	0.0004	0.4103
6	20	6	0.0010	CA−p02	5	0.0010	0.1618
7	17	6	0.0071	CA−p02	4	0.0017	0.1612
8	17	5	0.0021	CA−p02	4	0.0020	0.3303
9	20	7	0.0042	CA−p02	5	0.0010	0.0830
10	28	14	0.0503	CA/2−p02	7	0.0001	0.0007
11	28	15	0.1028	CA/2−p02	9	0.0008	0.0020
12	28	17	0.0895	CA/2−p02	10	0.0033	0.0045
13	30	16	0.3164	CA/2−p02	10	0.0007	0.0014
14	25	16	0.4661	CA/2−p02	8	0.0015	0.0011
15	20	13	0.5211	CA/2−p02	6	0.0022	0.0011
16	10	6	0.4406	CA/2+p02	5	0.5165	0.4836

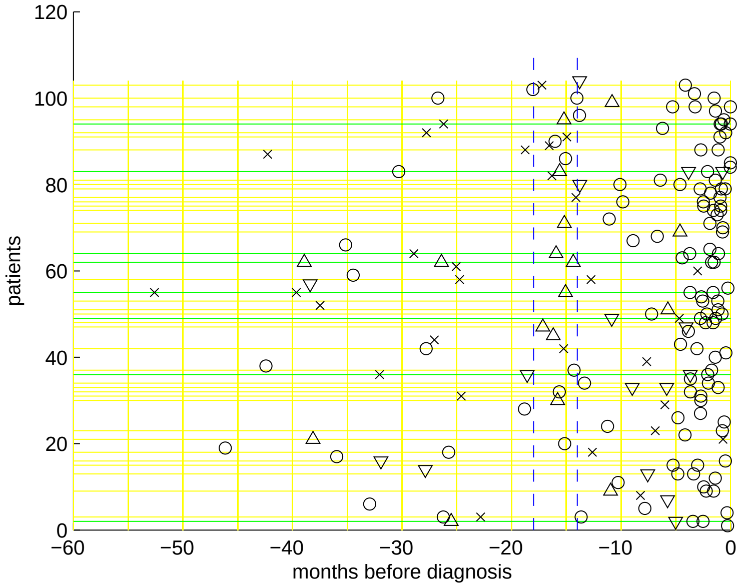
3 represented in Table B.1 are adjusted.

Figures B.1a and B.1b illustrate the performance of classification rules $\log C - 2\log I(2)$ and $\log C - \log I(3)$, respectively, in comparison with the performance of $\log C$. The horizontal axis shows time to diagnosis, the vertical one — triplet groups (corresponding to case patients) in this time interval. A circle means that a triple was correctly classified by both rules. A cross means misclassification in both cases. A triangle shows either improvement (up-directed) or deterioration (down-directed) after addition of a $-2\log I(2)$ or $-\log I(3)$ component. The figures demonstrate that most triplets with the measurement date close to diagnosis date are predicted correctly even by the $\log C$ rule. Most samples where addition of a peak to CA125 allows correct classification are in the interval of 13–16 months before the diagnosis.

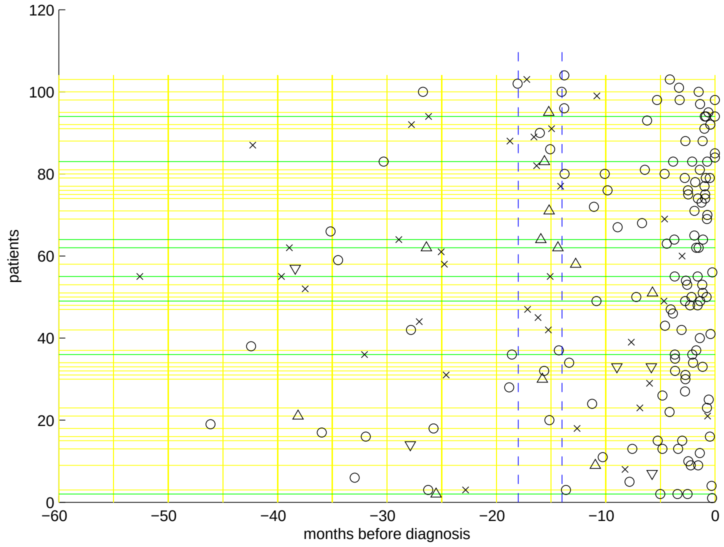
Figures 4.6 and B.2 show the median dynamics of $\log C$ versus $\log C - \log I(3)$ and $\log C - 2\log I(2)$ for case measurements. For each time moment,

Table B.4: Experimental results for triplet classification with peak 3

		CA125 only		CA125 and peak 3			
t	$\|S_t\|$	Errors	p-value	Rule	Errors	p-value	Cond. p-value
0	68	2	0.0001	2CA−p03	2	0.0001	0.9114
1	56	4	0.0001	CA+p03	4	0.0001	0.8815
2	47	6	0.0001	2CA−p03	5	0.0001	0.5279
3	36	8	0.0001	2CA−p03	7	0.0001	0.6777
4	27	7	0.0001	CA+p03	6	0.0001	0.8735
5	23	7	0.0008	2CA−p03	6	0.0007	0.6316
6	20	6	0.001	CA−p03	6	0.0046	0.6872
7	17	6	0.0071	CA−p03	5	0.0098	0.5735
8	17	5	0.0021	CA−p03	4	0.002	0.2781
9	20	7	0.0042	CA−p03	5	0.0009	0.0693
10	28	14	0.0503	CA−p03	6	0.0001	0.0002
11	28	15	0.1028	CA−p03	8	0.0004	0.0005
12	28	17	0.0895	CA−p03	10	0.0049	0.0049
13	30	16	0.3164	CA−p03	10	0.0015	0.0016
14	25	16	0.4661	CA−p03	10	0.0301	0.0181
15	20	13	0.5211	CA−p03	8	0.0577	0.0683
16	10	6	0.4406	CA/2+p03	5	0.5979	0.7643



(a) Comparison of $\log C$ with $\log C - 2\log I(2)$



(b) Comparison of $\log C$ with $\log C - \log I(3)$

Figure B.1: Comparison of $\log C$ with $\log C - 2\log I(2)$ and $\log C - \log I(3)$ rules on time/patient scale

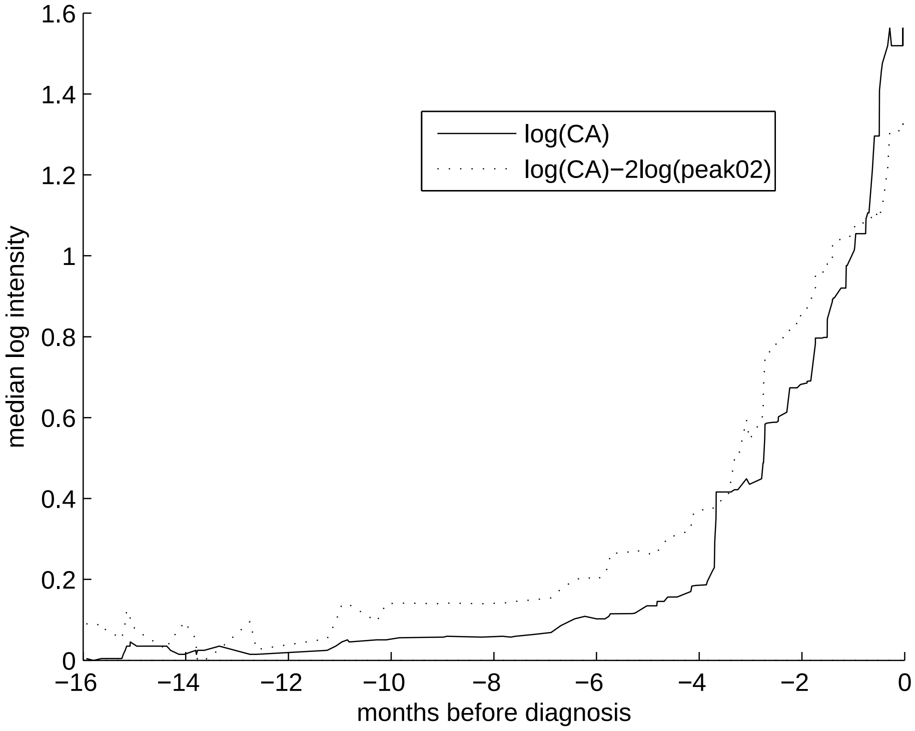


Figure B.2: UKCTOCS OC: median dynamics of rules $\log C$ and $\log C - 2\log I(2)$ (for cases only)

the latest available case measurement for each triplet group is taken into account. These measurements are averaged by median through all triplet groups. The figures illustrate that rules combining CA125 with peak intensity start to grow earlier than $\log C$. However, the CA125 growth at the moments close to diagnosis is quicker due to the exponential growth of CA125.

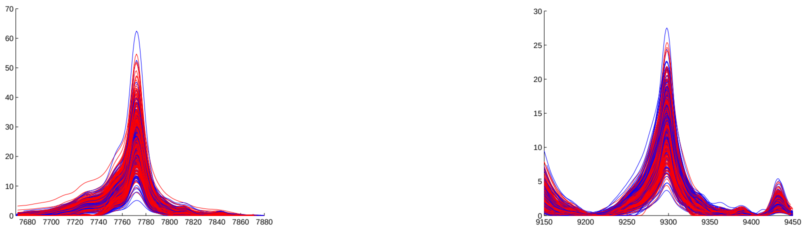
B.5 Conclusions

The purpose of this study has been to demonstrate that mass spectra carry significant information to make an early diagnosis of ovarian cancer and to identify peaks that allow making this early diagnosis. Intensities of certain peaks combined with the level of CA125 provide statistically reliable information for cancer prediction. The predictive power of CA125 alone is more limited.

We can pinpoint two peaks that can make early detection of ovarian cancer possible and deserve much attention:

- peak 2 (m/z-value = 7772 Da);
- peak 3 (m/z-value = 9297 Da).

These groups 2 and 3 plotted together for all cases and controls are shown in Figure B.3. As the peak plots show, there is no clear visual separation between cases and controls that was achieved in [63]. And this cannot be caused by the difference in pre-processing since the preprocessing described in this thesis was applied to the data described in [63] and confirmed visual separation.



(a) Peak 2

(b) Peak 3

Figure B.3: Peak groups 7772 Da (peak 2) and 9297 Da (peak 3): blue plots correspond to controls, red plots correspond to cases

The detailed results of our current research are shown in Tables B.2, B.3 and B.4. The main findings are accumulated in Table B.1. The difference between Tables B.2 and B.3–B.4 is in the hypothesis being checked:

- Hypothesis 1 checked in Table B.2 is that all the peaks contain some information useful to improve triplet classification in comparison with the use of CA125.
- Hypothesis 2 checked in Tables B.3–B.4 is that a specific peak (peak 2 or peak 3) contains such useful information.

In [26], it was found out that Hypothesis 1 could be confirmed statistically; here we also checked Hypothesis 2.

In general, we see that CA125 itself is enough for satisfactory prediction for up to 9 months before the diagnosis. It follows from conditional p-values that, in this range, other peaks cannot add significant improvement to this, and the quality of non-conditional p-values is caused by CA125 itself.

For more than 9 months before the diagnosis, CA125 produces less information, and this can be significantly completed by other peaks. The time where combination with mass spectrometry profile peaks works varies for different hypotheses. This time is 15 months for Hypothesis 1 and 15 or 13 months for Hypothesis 2 for peaks 2 or 3, respectively. These results are summarised in Table B.5.

Table B.5: The predictive ability of CA125 on its own, with all peaks and certain peaks

Features	Period of significant discrimination
CA125	9 months
CA125 + all peaks	15 months
CA125 + peak 2	15 months
CA125 + peak 3	13 months

Thus, we can come to the following conclusions:

1. Mass spectra contain information extending the period of statistically significant discrimination between controls and cases provided by CA125 to up to 15 months in advance of the moment of diagnosis.

2. Peak 2 (m/z-value = 7772 Da) and peak 3 (m/z-value = 9297 Da) separately contain such information for up to 15 and 13 months in advance of the moment of diagnosis.

An interesting point in this discussion is that the moment $T = 0$ is the moment of the diagnosis confirmed by histology/cytology but not the clinical diagnosis. The women had no clinical symptoms and were picked up by the screening either by the CA125/Risk of Ovarian Cancer Algorithm (ROC) strategy [40] or the ultrasound strategy. The time of clinical diagnosis is not known for any of these women since the doctors involved in the study did not wait for the women to become symptomatic and have a clinical diagnosis. They operated on them and diagnosed the cancer earlier. Previous studies suggested that screening may be able to pick up ovarian cancer 18 months before they would have presented clinically as ovarian disease. This means that CA125 and mass spectrometry profile peaks are able to discriminate between healthy samples and samples with ovarian cancer up to 33 months in advance of the clinical diagnosis.

To sum up, different techniques allow us to provide statistically significant predictions of ovarian cancer up to 33 months in advance of the date of clinical diagnosis. This period can be broken down by the cause of discrimination:

- 18 months due to screening methods;
- 10 months due to information contained CA125 levels;
- 5 months due to information contained in mass spectrometry profile peaks.

Appendix C

Application of Confidence and Venn Machines to the VLA Data

This appendix represents the results of application of confidence and Venn machines to data provided by VLA. There are two data sets:

1. Salmonella microarray data
2. Salmonella mass spectrometry data

Both sets represent Salmonella strains. However, they are not related to each other, and the objectives of their analysis were different. The aim of microarray data study was to differentiate Salmonella serotypes from each other whereas mass spectrometry data were processed in order to discriminate vaccine strains from wild type strains of the same serotype.

We applied different types of confidence machines (including the ones developed in Chapter 3) to the microarray data. The mass spectrometry data set was analysed by logistic Venn machines designed in Chapter 4 for proteomic data.

C.1 Application of Confidence Machines to the Microarray Data

C.1.1 Microarray Data of Salmonella

The microarray data of Salmonella pertain to strains collected from epidemiological surveys from food animals that are likely to be a source of Salmonella contamination for humans. The task is to discriminate among three Salmonella serotypes: S. Enteritidis, S. Newport and S. Typhimurium. Each strain is represented by at least two replicates. There are 56 strains (120 replicates) in total. The numbers of analysed strains and replicates of each serotype are provided in Table C.1. Each replicate has 3858 features — real numbers between 0 and 1. Only genes without missing information are used. These features were averaged across replicates and rounded to 0 and 1.

Table C.1: The number of analysed strains and replicates in each serotype of the Salmonella microarray data

Serotype	Number of strains	Number of replicates
S. Enteritidis	24	51
S. Newport	14	33
S. Typhimurium	18	36

C.1.2 Results

The Salmonella microarray data were applied to different confidence machines, including the ones designed in Chapter 3: CM-RF, CM-1NN and CM-RF-5NN (see Section 3.1.1 for details).

Table C.2 represents p-values, confidence and credibility for CM-RF applied to the first 25 strains of the microarray data in the online mode, for illustrative purposes: the first column shows the number of strains we consider, the others contain p-values for three serotypes, confidence, credibility, predicted serotype (or a tie) and a true serotype.

Table C.2: Classification with confidence for the CM-RF applied to the Salmonella microarray data in the online mode: each strain is complemented with p-values, one for each serotype, as well as values of confidence and credibility

No.	p-value for Enteritidis	p-value for Newport	p-value for Typhimurium	Confidence	Credibility	Predicted label	True label
3	100.0%	66.7%	100.0%	0.0%	100.0%	Tie	1
4	75.0%	75.0%	75.0%	25.0%	75.0%	Tie	2
5	40.0%	100.0%	40.0%	60.0%	100.0%	2	2
6	33.3%	50.0%	33.3%	66.7%	50.0%	2	1
7	28.6%	28.6%	14.3%	71.4%	28.6%	Tie	3
8	75.0%	12.5%	12.5%	87.5%	75.0%	1	1
9	11.1%	11.1%	11.1%	88.9%	11.1%	Tie	1
10	70.0%	10.0%	10.0%	90.0%	70.0%	1	1
11	9.1%	9.1%	45.5%	90.9%	45.5%	3	3
12	8.3%	8.3%	8.3%	91.7%	8.3%	Tie	2
13	7.7%	7.7%	84.6%	92.3%	84.6%	3	3
14	7.1%	7.1%	100.0%	92.9%	100.0%	3	3
15	6.7%	26.7%	6.7%	93.3%	26.7%	2	2
16	81.3%	6.3%	6.3%	93.8%	81.3%	1	1
17	5.9%	41.2%	5.9%	94.1%	41.2%	2	2
18	5.6%	5.6%	100.0%	94.4%	100.0%	3	3
19	5.3%	15.8%	5.3%	94.7%	15.8%	2	2
20	10.0%	5.0%	5.0%	95.0%	10.0%	1	1
21	4.8%	4.8%	95.2%	95.2%	95.2%	3	3
22	4.5%	4.5%	9.1%	95.5%	9.1%	3	3
23	30.4%	4.3%	4.3%	95.7%	30.4%	1	1
24	4.2%	4.2%	50.0%	95.8%	50.0%	3	3
25	4.0%	4.0%	68.0%	96.0%	68.0%	3	3

Let us consider several strains to illustrate the outputs. Strain 21 has three p-values one of which is high (95.2%), the others are low (4.8%). This results in high confidence and high credibility of 95.2% and identifies the prediction as reliable: only one serotype conforms well with the rest of the set. If this strain were classified as S. Enteritidis or S. Newport, this would mean that an event of probability $\leq 4.8\%$ occurred. For this reason, we expect this strain to be S. Typhimurium, which is correct.

In contrast, both strains 12 and 22 have low p-values for each serotype, which means that we cannot be confident when assigning these strains to any of the serotypes. These low p-values produce high confidence (91.7% and 95.5%) and low credibility (8.3% and 9.1%). For this reason, we are likely to make an erroneous prediction. As a result, the output for strain 12 is a tie, but prediction for strain 22 is correct.

In general, one can see from Table C.2 that when the number of strains is relatively low (up to 12 examples), the situations occur when forced predictions make errors or output ties. However, starting from strain 13, we never output erroneous predictions, confidence is always high (it has the maximum possible value), but credibility varies a lot. High value of confidence implies that we are confident in rejecting the serotypes that are not true. This tendency can be observed starting from example 8.

These experiments demonstrated that the Salmonella microarray data set is clean, and it is enough to process 12 strains to start making correct predictions with high confidence. This will be further confirmed when identifying the number of strains starting from which confidence machines stop producing multiple predictions.

Table C.3: Forced accuracy of confidence machines applied to the Salmonella microarray data in the leave-one-out mode

Algorithm	Forced accuracy
CM-RF	98.2–100.0%
CM-RF-1NN	98.2–100.0%
CM-RF-5NN	98.2–100.0%
CM-1NN	98.2–100.0%
CM-5NN	98.2–100.0%
CM-SVM (rbf, 5)	92.9–94.6%
CM-SVM (poly, 5)	98.2–100.0%
Max bare prediction	100.0%

Since the Salmonella microarray data set is so pure, when we launched experiments in the leave-one-out mode, most confidence machines provided high forced accuracy on the data: 55 correct predictions and 1 tie for 56 objects. The accuracy of different confidence machines can be seen in Table C.3. The lower boundary of the range shown is the accuracy when all ties make erroneous predictions, the upper — when all ties are correct. The last row of the table represents the maximum accuracy of simple predictors achieved on the data. Among these predictors we considered random forests, SVM and kNN. Comparison shows that confidence machines are not beaten by simple predictors in accuracy when forced to output singletons as predictions. At the same time, they assign confidence to each prediction and produce the second type of predictions — region predictions, which are always valid.

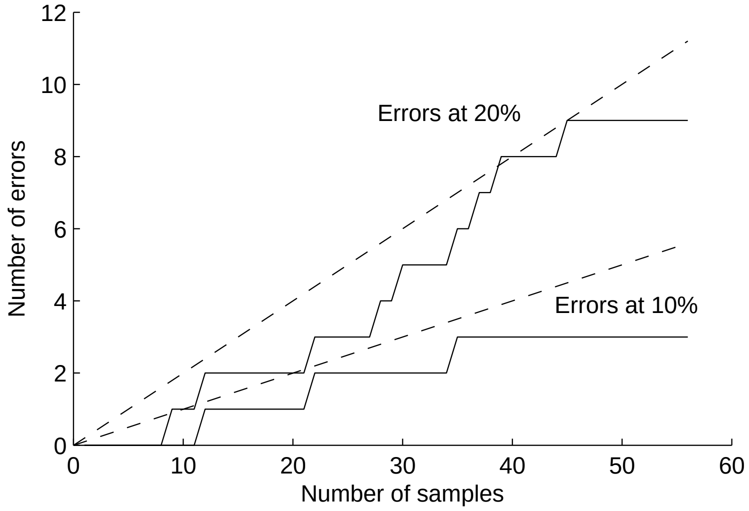


Figure C.1: Validity of CM-RF-1NN applied to the Salmonella microarray data in the online mode: the horizontal axis represents the number of examples in the online mode, dotted and solid lines demonstrate the expected and actual numbers of errors for different significance levels, respectively

To demonstrate the validity of confidence machines, we launched them as region predictors. These predictors proved to be valid, that is, for a given significance level $\epsilon > 0$ the rate of erroneous predictions (predictions not containing an actual label) does not exceed ϵ. This was the case in all experiments.

The validity example is demonstrated in Figure C.1. The figure displays the erroneous predictions dynamics and confirms the validity of the CM-RF-1NN applied to the Salmonella microarray data: solid lines, which represent the actual number of errors, are close to dotted lines, which demonstrate the expected number of errors.

Not only did our predictions have a guaranteed error rate, but also most predictions obtained in the experiments were certain, that is, the algorithm usually output one label as a prediction. In addition, all certain predictions were correct, which implies that most errors were made by empty predictions. The exact rates of certain predictions and correct certain predictions are shown in Table C.4.

Figure C.2 demonstrates the dynamics of efficiency characteristics at significance level of 10% of the CM-RF-1NN applied in the online mode to the data. The characteristics shown are the number of multiple predictions, certain predictions and empty predictions. The figure demonstrates that while the number of analysed strains is low (up to 9 examples), they do not carry enough information to make certain predictions without losing validity, that's why most predictions are multiple. But starting from Salmonella strain 10, we have accumulated enough information so that multiple predictions stopped

Table C.4: Efficiency of confidence machines in the leave-one-out mode: certain prediction and correct certain prediction rates for different confidence machines at significance levels $\epsilon = 5\%$ and 10%

	$\epsilon = 10\%$		$\epsilon = 5\%$	
Confidence machine	Certain	Correct certain	Certain	Correct certain
CM-RF	91.1%	91.1%	96.4%	96.4%
CM-RF-1NN	91.1%	91.1%	98.2%	98.2%
CM-RF-5NN	92.9%	92.9%	98.2%	98.2%
CM-1NN	91.1%	91.1%	96.4%	96.4%
CM-5NN	91.1%	91.1%	96.4%	96.4%
CM-SVM (rbf, 5)	94.6%	91.1%	94.6%	91.1%
CM-SVM (poly, 5)	91.1%	91.1%	96.4%	96.4%

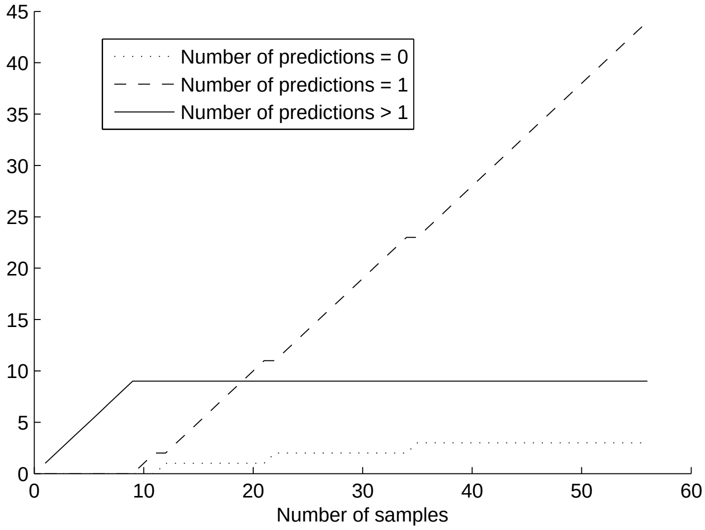


Figure C.2: Efficiency at significance level of 10% for the CM-RF-1NN applied to the Salmonella microarray data in the online mode

occurring, all region predictions contain exactly one label, and this label is correct. The plot confirms purity of the Salmonella microarray data and allows us to determine the number of strains required for producing correct predictions in the online mode: ten strains contain enough information to make valid correct predictions at the significance level of 10%.

C.2 Application of Confidence and Venn Machines to Proteomic Data of Salmonella

C.2.1 Proteomic Data of Salmonella

The aim of this study is to discriminate Salmonella vaccine strains from wild type field strains. We analysed the set of 50 vaccine strains (Gallivac vaccine strain) and 43 wild type strains. Both vaccine and wild type strains belong to the same serotype S. Enteritidis.

Each strain was represented by three spots; each spot produced 3 spot replicates. Therefore, there were 9 replicates per strain. Pre-processing described in Section 4.2.1 was applied to each replicate and resulted in representation of each mass spectra as a vector of 25 features corresponding to the most common peaks. The median was later taken for each feature across replicates of the same strain.

C.2.2 Results

The proteomic data of Salmonella strains were processed by methods designed for mass spectrometry data analysis in this thesis. Here we show the results of application of logistic Venn machines (Section 4.3.2). Categorized confidence machines based on linear rules (Section 4.3.1) were also applied, but the accuracy we could achieve was low (65.6%).

The logistic Venn machines were applied with the taxonomy comprising 5 categories to avoid a small number of categories and a small number of strains in each category. Before logistic Venn machine was launched, logarithm transformation was applied to the data. Hence each object was represented by a vector comprising the following features: intensities of the most frequent peaks on the logarithmic scale and value '1' for possible absolute term in logistic regression model.

When applied in the leave-one-out mode, logistic Venn machines output forced predictions with the accuracy of 79.6% whereas accuracy of its underlying algorithm, logistic regression, on the same data is 75.3%. In addition, the logistic Venn machine complements each prediction with the interval of probability that this prediction is correct. Several examples can be found in Table C.5.

For each example, the table contains the true label $y = y_{\mathrm{new}}$, and a Venn prediction — the interval $[P^-_{\mathrm{new}}, P^+_{\mathrm{new}}]$ of probability that $y = 1$. Label 1 corresponds to vaccine strains, label 0 to wild type strains. From Table C.5 one can see that logistic Venn machine outputs prediction intervals [0, 0.167] and [0.944, 1] for probabilities that examples 1 and 45 are cases ($y = 1$). As prediction intervals indicate, the correct labels for example 1 and 45 are 0 and 1, respectively. The table also includes direct predictions P_{new} output by logistic regression for each example. The table demonstrates that both

Table C.5: Leave-one-out Venn predictions for the Salmonella mass spectrometry data

No.	True label	Venn prediction	Direct prediction
1	0	0–0.167	0.039
2	0	0.111–0.684	0.427
3	0	0–0.053	0.003
4	0	0.632–0.944	0.792
5	0	0.111–0.684	0.409
6	0	0.111–0.737	0.433
7	0	0.111–0.684	0.174
8	0	0.111–0.632	0.118
9	0	0–0.278	0.005
10	0	0.111–0.737	0.633
11	0	0–0.053	0.000
12	0	0–0.222	0.031
13	0	0–0.684	0.038
14	0	0–0.222	0.035
15	0	0–0.278	0.002
		...	
44	1	0.111–0.632	0.307
45	1	0.944–1	0.986
46	1	0.889–1	0.914
47	1	0.579–0.944	0.814
48	1	0.579–0.632	0.674
49	1	0.833–1	0.963
50	1	0.684–1	0.908
51	1	0.526–0.944	0.736
52	1	0.526–0.944	0.880
53	1	0.111–0.632	0.304
54	1	0.895–1	0.983
55	1	0.579–1	0.950
56	1	0.889–1	0.919
57	1	0.111–0.632	0.268
58	1	0.056–0.111	0.146
		...	

direct and Venn predictions can be correct or erroneous. For example, for wild type strain 4 and vaccine strain 44, both direct and Venn predictions are not correct.

Now we will demonstrate implications of validity of logistic Venn machines. We aim to show that true probabilities of label distribution are covered or almost covered by the interval between lower Venn prediction and upper Venn prediction. Since we do not know true probabilities of label distribution, we compare empirical probabilities, that is, mean true labels, with mean direct and Venn predictions.

Figure C.3 is a graphical representation of corresponding cumulative results and conforms with validity of Venn machine outputs. The horizontal axis shows the number of observed examples. The vertical axis shows the cumulative values of: (1) true labels y_{new} (a solid line); (2) lower and upper Venn predictions $P^{-}_{\text{new}}, P^{+}_{\text{new}}$ (two dot-dashed lines) and (3) cumulative direct predictions P_{new} (a dashed line). The examples are sorted according to direct predictions.

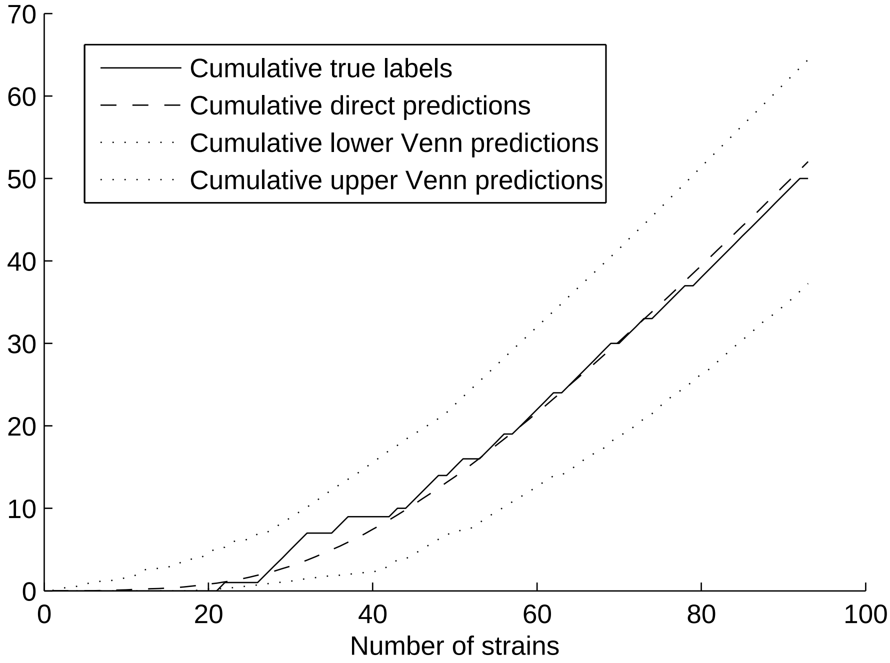


Figure C.3: Cumulative Venn and direct predictions output by the logistic Venn machine applied to the Salmonella mass spectrometry data

Bibliography

[1] N. Leigh Anderson and Norman G. Anderson. The human plasma proteome: history, character, and diagnostic prospects. *Molecular and Cellular Proteomics.* 1:845–867, 2002.

[2] Alexander I. Archakov, Yurii D. Ivanov, Andrey V. Lisitsa, Victor G. Zgoda. AFM fishing nanotechnology is the way to reverse the Avogadro number in proteomics. *Proteomics.* 7:4–9, 2007.

[3] Jennifer J. Barrett, David A. Cairns. Application of the random forest classification method to peaks detected from mass spectrometric proteomic profiles of cancer patients and controls. *Statistical Applications in Genetics and Molecular Biology.* 7, Art. 4, 2008.

[4] R. C. Bast Jr., D. Badgwell, Z. Lu, R. Marquez, D. Rosen, J. Liu, K. A. Baggerly, E. N. Atkinson, S. Skates, Z. Zhang, A. Lokshin, U. Menon, I. Jacobs, K. Lu. New tumor markers: CA125 and beyond. *International Journal of Gynecological Cancer.* 15, Suppl. 3:274–281, 2005.

[5] R. C. Bast, F. J. Xu, Y. H. Yu, S. Barnhill, Z. Zhang, G. B. Mills. CA 125: the past and the future. *The International Journal of Biological Markers.* 13:179-87, 1998.

[6] Tony Bellotti, Zhiyuan Luo, Alex Gammerman, Frederick W. van Delft, Vaskar Saha. Qualified predictions for microarray and proteomics pattern diagnostics with confidence machines. *International Journal of Neural Systems.* 15:247–258, 2005.

[7] Leo Breiman. Random forests. *Machine Learning.* 45:5–32, 2001.

[8] Leo Breiman and Adele Cutler. Random forests. http://www.stat.berkeley.edu/users/breiman/RandomForests/cc_home.htm#intro.

[9] P. A. Brioschi, O. Irion, P. Bischof, M. Bader, M. Forni, F. Krauer. Serum CA 125 in epithelial ovarian cancer. A longitudinal study. *An International Journal of Obstetrics and Gynaecology.* 94:196–201, 1987.

[10] Brian Burford, Dmitry Devetyarov, Ilia Nouretdinov, Zhiyuan Luo, Alexey Chervonenkis, Volodya Vovk, Mike Waterfield, Ali Tiss, Celia Smith, Rainer Cramer, Alex Gentry-Maharaj, Rachel Hallett, Stephane Camuzeaux, Jeremy Ford, John Timms, Usha Menon, Ian Jacobs and Alex Gammerman. Early detection of breast cancer in UKCTOCS using proteomic biomarkers. *Technical report CLRC-TR-08-03*, http://www.clrc.rhul.ac.uk/projects/proteomic3.htm, 2009.

[11] Brian Burford, Ali Tiss, Stephane Camuzeaux, Jeremy Ford, Alex Gentry-Maharaj, Usha Menon, Ian Jacobs, Dmitry Devetyarov, Zhiyuan Luo, Iila Nouretdinov, Volodya Vovk, John Timms, Rainer Cramer, Alex Gammerman. Identification of proteomic biomarkers in the UKCTOCS heart disease data set. *Technical report CLRC-TR-08-04*, http://www.clrc.rhul.ac.uk/projects/proteomic3.htm, 2008.

[12] Sabarni K. Chatterjee, Bruce R. Zetter. Cancer biomarkers: knowing the present and predicting the future. *Future Oncology.* 1:37–50, 2005.

[13] Nello Cristianini, John Shawe-Taylor. *An Introduction to Support Vector Machines (and Other Kernel Based Learning Methods).* Cambridge University Press, Cambridge, 2000.

[14] Martin J. Crowder and David J. Hand. *Analysis of Repeated Measures.* Chapman and Hall, London, 1990.

[15] Dmitry Devetyarov, Ilia Nouretdinov, Brian Burford, Zhiyuan Luo, Alexey Chervonenkis, Volodya Vovk, Mike Waterfield, Ali Tiss, Celia Smith, Rainer Cramer, Alex Gentry-Maharaj, Rachel Hallett, Stephane Camuzeaux, Jeremy Ford, John Timms, Usha Menon, Ian Jacobs and

Alex Gammerman. Analysis of serial UKCTOCS-OC data: discriminating abilities of proteomics peaks. *Technical report CLRC-TR-08-02*, `http://www.clrc.rhul.ac.uk/projects/proteomic3.htm`, 2008

[16] Dmitry Devetyarov, Ilia Nouretdinov, Brian Burford, Stephane Camuzeaux, Alex Gentry-Maharaj, Ali Tiss, Celia Smith, Zhiyuan Luo, Alexey Chervonenkis1, Rachel Hallett, Volodya Vovk, Mike Waterfield, Rainer Cramer, John F. Timms, Ian Jacobs, Usha Menon and Alex Gammerman. Prediction with Confidence prior to Cancer Diagnosis. Submitted to *International Journal of Proteomics.*

[17] Persi Diaconis. The Markov chain Monte Carlo revolution. *Bulletin (New Series) of the American Mathematical Society.* 46:179–205, 2009.

[18] Eleftherios P. Diamandis. Mass spectrometry as a diagnostic and a cancer biomarker discovery tool: opportunities and potential limitations. *Molecular and Cellular Proteomics.* 3:367–78, 2004.

[19] Rebecca Ferrini. Screening asymptomatic women for ovarian cancer: American College of Preventive Medicine practice policy. `http://www.acpm.org/ovary.htm`.

[20] Sally Floyd and Manfred K. Warmuth. Comple compression, learnability, and the Vapnik-Chervonenkis dimension. *Machine Learning.* 21:269–304, 1995.

[21] Eric T. Fung. A recipe for proteomics diagnostic test development: the OVA1 test, from biomarker discovery to FDA clearance. *Clinical Chemistry.* 56:327–9, 2010.

[22] Alex Gammerman, Ilia Nouretdinov, Brian Burford, Alexey Chervonenkis, Vladimir Vovk, Zhiyuan Luo. Clinical mass spectrometry proteomic diagnosis by conformal predictors. *Statistical Applications in Genetics and Molecular Biology.* 7, Art. 13, 2008.

[23] A. Gammerman, A.R. Thatcher. Bayesian diagnostic probabilities without assuming independence of symptoms. *Methods of Information in Medicine*. 30:15–22, 1991.

[24] Alex Gammerman, Vladimir Vovk. Hedging predictions in machine learning. *Computer Journal*. 50:151–163, 2007.

[25] Alex Gammerman, Vladimir Vovk. Prediction algorithms and confidence measures based on algorithmic randomness theory. *Theoretical Computer Science*. 287:209–217, 2002.

[26] Alex Gammerman, Volodya Vovk, Brian Burford, Ilia Nouretdinov, Zhiyuan Luo, Alexey Chervonenkis, Mike Waterfield, Rainer Cramer, Paul Tempst, Josep Villanueva, Musarat Kabir, Stephane Camuzeaux, John Timms, Usha Menon and Ian Jacobs. Serum proteomic abnormality predating screen detection of ovarian cancer. *The Computer Journal*. 52:326–333, 2008.

[27] Alex Gammerman, Volodya Vovk, Vladimir Vapnik. Learning by transduction. *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pages 148–155, 1998.

[28] Andrew Gelman, John B. Carlin, Hal S. Stern, Donald B. Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, Boca Raton, FL, 2003.

[29] David J. Hand. Breast cancer diagnosis from proteomic mass spectrometry data: a comparative evaluation. *Statistical Applications in Genetics and Molecular Biology*. 7, Art. 15, 2008.

[30] David J. Hand and Martin J. Crowder. *Practical Longitudinal Data Analysis*. Chapman and Hall, London, 1996.

[31] Glen L. Hortin. The MALDI-TOF mass spectrometric view of the plasma proteome and peptidome. *Clinical Chemistry*. 52:1223–1237, 2006.

[32] John M. Koomen, Donghui Li, Lian-chun Xiao, Thomas C. Liu, Kevin R. Coombes, James Abbruzzese, and Ryuji Kobayashi. Direct tandem mass

spectrometry reveals limitations in protein profiling experiments for plasma biomarker discovery. *Journal of Proteome Research.* 4:972–81, 2005.

[33] Katherine R. Kozak, Feng Su, Julian P. Whitelegge, Kym Faull, Srinivasa Reddy, Robin Farias-Eisner. Characterization of serum biomarkers for detection of early stage ovarian cancer. *Proteomics.* 5:4589–96, 2005.

[34] Nick Littlestone and Manfred K. Warmuth. Relating data compression and learnability. *Technical report*, University of California, Santa Cruz, 1986.

[35] Joseph A. Ludwig, John N. Weinstein. Biomarkers in cancer staging, prognosis and treatment selection. *Nature Reviews Cancer.* 5:845–56, 2005.

[36] Peter McCullagh, Vladimir Vovk, Ilia Nouretdinov, Dmitry Devetyarov, and Alex Gammerman. Conditional prediction intervals for linear regression. *Proceedings of the 8th International Conference on Machine Learning and Applications*, pages 131–138, 2009.

[37] Thomas Melluish, Craig Saunders, Ilia Nouretdinov, Vladimir Vovk. The typicalness framework: a comparison with the bayesian approach. *Technical report 01-05*, Royal Holloway, University of London, 2001.

[38] Usha Menon, Aleksandra Gentry-Maharaj, Rachel Hallett, Andy Ryan, Matthew Burnell, Aarti Sharma, Sara Lewis, Susan Davies, Susan Philpott, Alberto Lopes, Keith Godfrey, David Oram, Jonathan Herod, Karin Williamson, Mourad W. Seif, Ian Scott, Tim Mould, Robert Woolas, John Murdoch, Stephen Dobbs, Nazar N. Amso, Simon Leeson, Derek Cruickshank, Alistair Mcguire, Stuart Campbell, Lesley Fallowfield, Naveena Singh, Anne Dawnay, Steven J. Skates, Mahesh Parmar, Ian Jacobs. Sensitivity and specificity of multimodal and ultrasound screening for ovarian cancer, and stage distribution of detected cancers: results of the prevalence screen of the UK Collaborative Trial of Ovarian Cancer Screening (UKCTOCS). *Lancet Oncology.* 10:327–340, 2009.

[39] Usha Menon, Aleksandra Gentry-Maharaj, Andy Ryan, Aarti Sharma, Matthew Burnell, Rachel Hallett, Sara Lewis, Alberto Lopez, Keith God-

frey, David Oram, Jonathan Herod, Karin Williamson, Mourad Seif, Ian Scott, Tim Mould, Robert Woolas, John Murdoch, Stephen Dobbs, Nazar Amso, Simon Leeson, Derek Cruickshank, Ali McGuire, Stuart Campbell, Lesley Fallowfield, Steve Skates, Mahesh Parmar, and Ian Jacobs. Recruitment to multicentre trialslessons from UKCTOCS: descriptive study. *British Medical Journal*. 337:a2079, 2008.

[40] Usha Menon, Steven J. Skates, Sara Lewis, Adam N. Rosenthal, Barnaby Rufford, Karen Sibley, Nicola MacDonald, Anne Dawnay, Arjun Jeyarajah, Robert C. Bast, Jr, David Oram, Ian J. Jacobs. Prospective study using the risk of ovarian cancer algorithm to screen for ovarian cancer. *Journal of Clinical Oncology*. 23:7919–7926, 2005.

[41] Richard von Mises. Grundlagen der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*. 5:52–99, 1919.

[42] Richard von Mises. *Wahrscheinlichkeitsrechnung, Statistik und Wahrheit*. Julius Springer, Wien, 1928.

[43] Tom M. Mitchell. *Machine Learning*. McGrow-Hill, New York, 1997.

[44] Ilia Nouretdinov, Brian Burford, Alex Gammerman. Application of inductive confidence machine to ICMLA competition data. *Proceedings of the 8th International Conference on Machine Learning and Applications*, pages 435–438, 2009.

[45] Ilia Nouretdinov, Brian Burford, Zhiyuan Luo, Alex Gammerman. Data analysis of 7 biomarkers. *Technical report*, Royal Holloway, University of London, 2008.

[46] Ilia Nouretdinov, Volodya Vovk, Michael Vyugin, and Alex Gammerman. Pattern recognition and density estimation under the general i.i.d. assumption. *Proceedings of the 14th Annual Conference on Computational Learning Theory*, volume 2011 of *Lecture notes in Artificial Intelligence*, pages 337–353, 2001.

[47] N. Osman, N. O'Leary, E. Mulcahy, N. Barrett, F. Wallis, K. Hickey, R. Gupta. Correlation of serum CA125 with stage, grade and survival of patients with epithelial ovarian cancer at a single centre. *Irish Medical Journal.* 101:245-7, 2008.

[48] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: A. J. Smola, P. Bartlett, B. Schoelkopf and D. Schuurmans (eds.), *Advances in Large Margin Classiers*, 61-74. MIT Press, Cambridge, 1999.

[49] Konstas Proedrou, Ilia Nouretdinov, Volodya Vovk, Alex Gammerman. Transductive confidence machines for pattern recognition. *Technical report 01-02*, Royal Holloway, University of London, 2001.

[50] Y. Qi, J. Klein-Seetharaman, Z. Bar-Joseph. Random forest similarity for protein-protein interaction prediction from multiple sources. *Preceedings of the Pacific Symposium on Biocomputing*, pages 531-542, 2005.

[51] Daniel G. Rosena, Lin Wangb, J. Neeley Atkinsonc, Yinhua Yub, Karen H. Lud, Eleftherios P. Diamandise, Ingegerd Hellstromf, Samuel C. Mokg, Jinsong Liua, Robert C. Bast Jr. Potential markers that complement expression of CA125 in epithelial ovarian cancer. *Gynecologyc Oncology.* 99: 267-277, 2005.

[52] Craig Saunders, Alex Gammerman, and Vladimir Vovk. Transduction with confidence and credibility. *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-1999)*, pages 722–726, 1999.

[53] O. David Sparkman. Mass spectrometry desk reference. Global View Pub, Pittsburgh, 2000.

[54] Foteini Strimenopoulou, Philip J. Brown. Empirical bayes logistic regression. *Statistical Applications in Genetics and Molecular Biology.* 7:Art. 9, 2008.

[55] Feng Su, Jennifer Lang, Ashutosh Kumar, Carey Ng, Brian Hsieh, Marc A. Suchard, Srinivasa T. Reddy, Robin Farias-Eisner. Validation of

candidate serum ovarian cancer biomarkers for early detection. *Biomark Insights*. 2:369–375, 2007.

[56] John F. Timms, Rainer Cramer, Stephane Camuzeaux, Ali Tiss, Celia Smith, Brian Burford, Ilia Nouretdinov, Dmitry Devetyarov, Aleksandra Gentry-Maharaj, Jeremy Ford, Zhiyuan Luo, Alex Gammerman, Usha Menon and Ian Jacobs. Peptides generated ex vivo from abundant serum proteins by tumour-specific txopeptidases are not useful biomarkers in ovarian cancer. *Clinical Chemistry*. 56:262–271, 2010.

[57] John F. Timms, Usha Menon, Dmitry Devetyarov, Ali Tiss, Stephane Camuzeaux, Katherine McCurrie, Ilia Nouretdinov, Brian Burford, Celia Smith, Alex Gentry-Maharaj, Rachel Hallett, Jeremy Ford, Zhiyuan Luo, Volodya Vovk, Alex Gammerman, Rainer Cramer and Ian Jacobs. Detection of ovarian cancer in pre-diagnosis samples using CA125 and MALDI-MS peaks. *Gynecologic Oncology*. Forthcoming.

[58] Ali Tiss, Celia Smith, Usha Menon, Ian Jacobs, John F. Timms, Rainer Cramer. A well-characterised peak identification list of MALDI MS profile peaks for human blood serum. *Proteomics*. 10:3388–3392, 2010.

[59] Ali Tiss, John F. Timms, Celia Smith, Dmitry Devetyarov, Aleksandra Gentry-Maharaj, Stephane Camuzeaux, Brian Burford, Ilia Nouretdinov, Jeremy Ford, Zhiyuan Luo, Ian Jacobs, Usha Menon, Alex Gammerman and Rainer Cramer. Highly accurate detection of ovarian cancer using CA125 but limited improvement with serum MALDI-TOF MS profiling. *International Journal of Gynecological Cancer*. 2010, in press.

[60] Stijn Vanderlooy, Laurens van der Maaten, and Ida Sprinkhuizen-Kuyper. Off-line learning with transductive confidence machines: an empirical evaluation. *Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition*, pages 310–323, 2007.

[61] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[62] John Venn. *The Logic of Chance*. Macmillan, London, 1866.

[63] Josep Villanueva, David R. Shaffer, John Philip, Carlos A. Chaparro, Hediye Erdjument-Bromage, Adam B. Olshen, Martin Fleisher, Hans Lilja, Edi Brogi, Jeff Boyd, Marta Sanchez-Carbayo, Eric C. Holland, Carlos Cordon-Cardo, Howard I. Scher and Paul Tempst. Differential exoprotease activities confer tumor-specific serum peptidome patterns. *The Journal of Clinical Investigation.* 116:271–284, 2006.

[64] Sonja Vorderwuelbecke, Steve Cleverley, Scot R. Weinberger and Andreas Wiesner. Protein quantification by the SELDI-TOF-MS-based ProteinChip system. *Nature Methods.* 2:393–395, 2005.

[65] Vladimir Vovk, Alexander Gammerman, Glenn Shafer. Algorithmic learning in a random world. Springer, New York, 2005.

[66] Vladimir Vovk, David Lindsay, Ilia Nouretdinov, Alex Gammerman. Mondrian confidence machine. *On-line Compression Modelling Project*, working paper 4, `http://www.vovk.net/cp/04.pdf`, 2003.

[67] Vladimir Vovk, Glenn Shafer and Ilia Nouretdinov. Self-calibrating probability forecasting. *On-line Compression Modelling Project*, working paper 9, `http://vovk.net/cp/09.pdf`, 2003.

[68] Martijn P. J. van der Werff, Bart Mertens, Mirre E. de Noo, Marco R. Bladergroen, Hans C. Dalebout, Rob A. E. M. Tollenaar, Andre M. Deelder. Case-control breast cancer study of MALDI-TOF proteomic mass spectrometry data on serum samples. *Statistical Applications in Genetics and Molecular Biology.* 7:Art. 2, 2008.

[69] J. D. Wulfkuhle, C. P. Paweletz, P. S. Steeg, E. F. Petricoin, 3rd, L. Liotta. Proteomic approaches to the diagnosis, treatment, and monitoring of cancer. *Advances in Experimental Medicine and Biology.* 532:59–68, 2003.

[70] Yong Zhou, Yan Wang, Monika Kovacs, Jinghua Jin and Jing Zhang. Microglial activation induced by neurodegeneration. *Molecular and Cellular Proteomics.* 4:1471–1479, 2005.

[71] Protocol for the United Kingdom Collaborative Trial of Ovarian Cancer Screening (UKCTOCS). Version 4. http://www.instituteforwomenshealth.ucl.ac.uk/academic_research/gynaecologicalcancer/gcrc/ukctocs/design, September 2008.