

Bachelor's Thesis



**Czech
Technical
University
in Prague**

Faculty of Electrical Engineering

Department of Cybernetics

Schizophrenia Relapse Detection in the ITAREPS Clinical Programme

Predrag Božović

Open Informatics—Computer and Information Science

May 2018

Supervisor: Ing. Eduard Bakštein, PhD



BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Božović Predrag** Personal ID number: **435033**
Faculty / Institute: **Faculty of Electrical Engineering**
Department / Institute: **Department of Cybernetics**
Study program: **Open Informatics**
Branch of study: **Computer and Information Science**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Schizophrenia Relapse Detection in the ITAREPS Clinical Programme

Bachelor's thesis title in Czech:

Detekce relapsů schizofrenie v programu ITAREPS

Guidelines:

ITAREPS is a clinical m-health programme for relapse prevention in schizophrenia, based on weekly self-reports, sent via SMS or mobile app. Typically, schizophrenia patients stay in a state of well-controlled disease symptoms, called „remission“ most of the time. Occasionally during the course of the disease, a sudden and clinically severe state of worsened symptoms called a „relapse“ occurs.

The aim of this thesis is to develop a new classifier for relapse detection, considering inter-individual differences in patient responses using principal component analysis (PCA):

- 1) Perform exploratory analysis of data from the ITAREPS programme, with special attention to patient behavior prior to relapse.
- 2) Use the PCA (or similar appropriate method) to analyze internal structure of questionnaire responses, especially comparing remission and relapse on individual and group level.
- 3) Develop a classifier for relapse detection using the remission-based transformation from previous point. Compare its results to current existing solution and to classifier based on the raw data directly.

Bibliography / sources:

- [1] Španiel, F., Vohlídková, P., Hrdlička, J., Kožený, J., Novák, T., Motlová, L., ... Höschl, C. (2008). ITAREPS: Information Technology Aided Relapse Prevention Programme in Schizophrenia. *Schizophrenia Research*, 98(1-3), 312-317. <http://doi.org/10.1016/j.schres.2007.09.005>.
- [2] Španiel, F., Bakstein, E., Anyz, J., Hlinka, J., Sieger, T., Hrdlička, J., ... Höschl, C. (2016). Relapse in schizophrenia: definitively not a bolt from the blue. *Neuroscience Letters*, S0304-3940(i6).
- [3] Housková, A. Individualizovaná detekce relapsu schizofrenních pacientů v programu ITAREPS, diplomová práce, ČVUT Praha, leden 2017.

Name and workplace of bachelor's thesis supervisor:

Ing. Eduard Bakštein, Ph.D., Analysis and Interpretation of Biomedical Data, FEE

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **12.01.2018** Deadline for bachelor thesis submission: **25.05.2018**

Assignment valid until: **30.09.2019**

Ing. Eduard Bakštein, Ph.D.
Supervisor's signature

doc. Ing. Tomáš Svoboda, Ph.D.
Head of department's signature

prof. Ing. Pavel Ripka, CSc.
Dean's signature

Acknowledgement / Declaration

I would like to immensely thank Eduard Bakštein, my thesis supervisor, for introducing me to the intricacies of the topic and guiding me through the realisation of this thesis.

Further, I would like to thank my partner, friends and family for proof-reading, supporting and encouraging me throughout the whole process.

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

Prague, date 25. 5. 2018

.....

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 25. 5. 2018

.....

Abstrakt / Abstract

Schizofrenie je jednou z nejzávažnějších psychiatrických poruch, která postihuje přibližně 0,5-1 % populace ve vyspělých zemích, s příznaky jako zkreslení reality, bludy a halucinace, má ničivý dopad na životy pacientů, jejich rodiny a okolí.

ITAREPS (*Information Technology Aided Relapse Prevention Programme in Schizophrenia*) představuje řešení telemedicine založené na týdenním vzdáleném sledování pacientů mobilním telefonem a zvládnutí onemocnění při schizofrenii a psychotických poruchách obecně. Zdravotníci dostávají upozornění, když se stav pacienta zhorší, což indikuje budoucí relaps. To umožňuje včas intervenovat a vyhnout se zbytečným hospitalizacím.

Tato práce popisuje úsilí o zlepšení přesnosti detekce relapsu zavedením komplexnějšího návrhu klasifikátoru. Data byla získána z klinického programu ITAREPS. Data v souboru jsou nejprve označena, abychom vydělili dvě nevyvážené třídy a získali sadu dodatečných atributů, na základě nichž lze sestavit binární klasifikátor. Vymodelujeme klasifikátor založený na metodě *gradient boosting*, který je pak natrénován a vyhodnocen za účelem získání dostatečné senzitivity, aniž bychom přehlíželi méně reprezentovanou kritickou třídu.

Klíčová slova: ITAREPS, schizofrenie, klinický program, binární klasifikace, gradient boosting, analýza časových řad

Schizophrenia is one of the most severe psychiatric disorders affecting around 0.5–1% of the population in developed countries, with symptoms such as reality distortion, delusions and hallucinations, it has a devastating impact on lives of patients and of their families, and surroundings.

The Information Technology Aided Relapse Prevention Programme in Schizophrenia presents a mobile phone-based telemedicine solution for weekly remote patient monitoring and disease management in schizophrenia and psychotic disorders in general. Healthcare professionals receive alerts when the patient's condition worsens which indicates a future relapse, to enable early intervention and avoid unnecessary hospitalisations.

This thesis presents an effort to improve relapse detection accuracy by introducing a more complex classifier design. The data is obtained from the ITAREPS clinical programme. The dataset is first labelled in order to extract two unbalanced classes as well as to extract a set of additional features upon which a binary classifier can be built. The classifier based on a gradient boosting machine is modelled, and then trained and evaluated with the aim of yielding sufficient sensitivity as to not overlook the under-represented critical class.

Keywords: ITAREPS, schizophrenia, clinical programme, binary classification, gradient boosting, time-series analysis

/ Contents

1 Introduction	1
1.1 Schizophrenia	1
1.2 ITAREPS	3
2 Methodological Review	5
2.1 Principal Component Analysis ..	5
2.2 Decision Tree Learning	6
2.3 Gradient Boosting	7
2.4 Performance Evaluation	8
2.4.1 k-fold Cross-validation.....	8
2.4.2 Confusion Matrix	9
2.4.3 Class Imbalance.....	9
2.4.4 Relative Feature Im- portance.....	10
2.4.5 Labelling Time Series....	10
3 Data Overview and Analysis	11
3.1 Exploring the Data Set	11
3.2 The Factor of Time	13
3.3 Splitting Data in Periods	15
3.4 Applying PCA	17
4 Preparation for Learning	19
4.1 Feature Extraction	19
4.1.1 Accounting for History ..	19
4.1.2 Resulting Feature Vec- tor	21
4.2 Resampling	21
4.3 Cross-validation	22
4.4 Hyperparameter Optimisa- tion	23
4.5 Classifier Model Variants	25
4.6 State of the Art Classifier	25
5 Results and Discussion	27
5.1 Classifier Evaluation	27
5.2 Feature Evaluation.....	29
5.3 Discussion and Limitations....	30
6 Conclusion	32
6.1 Future Work	33
References	34
A Abbreviations	37

Tables / Figures

<ul style="list-style-type: none"> 1.1. EWSQ 10 items for patients and family members3 1.2. Meaning of EWSQ 10 response values4 3.1. Hospitalisation count per patient..... 11 3.2. Chosen Period Lengths for Training a Classifier 17 3.3. Share of Sum-Zeros in Classification Periods 17 3.4. Top contributor principal component weights 18 4.1. Final layout of the feature vector 21 4.2. Effect of resampling procedure on the dataset 22 4.3. CV fold distribution 23 5.1. Comparison of classifiers with <code>alertInds</code> 28 	<ul style="list-style-type: none"> 1.1. Four typical patters in the course of schizophrenia1 2.1. An example of a scree plot5 2.2. An example of a decision tree ...6 3.1. Division of participants by type 11 3.2. Share of all sent messages by type of participants 11 3.3. Empirical cumulative density of rehospitalisation 12 3.4. EWSQ sent SMS density per week 12 3.5. ECDF of Sum of SMS 13 3.6. Pre- and Post-relapse time-series EWSQ scores 14 3.7. Pre-relapse varying growth 14 3.8. Pre-relapse varying constant... 14 3.9. Pre-relapse sum-zero then increase 14 3.10. Pre-relapse sum zero 14 3.11. Patient sum-zero, family member varying 14 3.12. Family member sum-zero, patient varying 14 3.13. Pre- and Post-relapse percentiles EWSQ 15 3.14. Weekly population mean and 95 percentile line 16 3.15. Bar graph of the contributions to variance by each principal component 18 4.1. Illustration of the resampling method 22 4.2. Explanation of how k-fold cross-validation is used..... 23 4.3. Illustration of the grid searching method 24 5.1. Final classifier relative predictor importance 29 5.2. No history considered classifier relative predictor importance 30 6.1. Relative predictor importance of partial sums..... 33
---	--

Chapter 1

Introduction

1.1 Schizophrenia

Schizophrenia is one of the most severe psychiatric disorders affecting around 0.5–1% of the population in developed countries [22]. Due to its tendency towards chronicity and the nature of its symptoms ranging from thought disorders to various forms of reality distortion, such as delusions and hallucinations, the illness has a devastating impact on lives of patients and of their families, and surroundings.

The onset of schizophrenia is typically within the ages of 20–39 years, but it may occur before puberty or in later years, as well. There is a greater relative risk of developing the illness in people born or brought up in inner cities [11], in people with a lower socioeconomic background [23] and in people who have close relatives suffering from schizophrenia [12].

Its symptoms can be divided into two categories: positive and negative. Positive (surplus, excess) symptoms include hallucinations and delusions, i.e. false perceptions and false personal beliefs held with absolute conviction. They frequently remain in chronic schizophrenia. Negative (deficit) symptoms include social withdrawal, apathy, self-neglect, poverty in form and content of speech etc. They tend to precede the onset of positive symptoms and may also be seen in acute episodes.

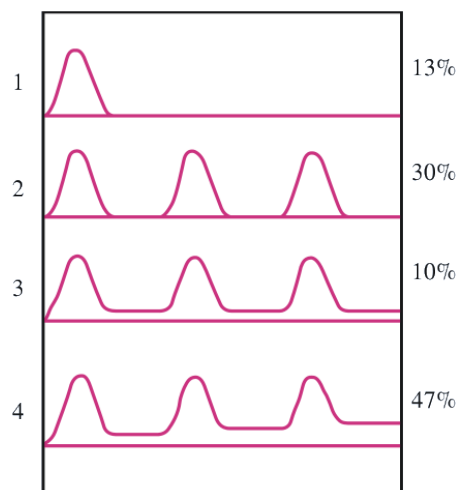


Figure 1.1. Four typical patterns in the course of schizophrenia and their prevalence acquired in a five-year follow-up [22][18] of 102 patients with schizophrenia.

A five-year study [18] conducted in the UK produced four typical patterns in the course of schizophrenia, as seen in Figure 1.1. All patterns have one thing in common: relapse—a return or worsening of symptoms—preceded and followed by periods of relative stability—remission. The first pattern is that a patient, after suffering a relapse,

returns to normality with minimal impairment. The second is similar to the first, with the exception that relapses reoccur. The third pattern also features recurring relapses, however there is persistent impairment after the first relapse—the patient doesn't return to normality. In the fourth pattern, the patient's condition worsens after each relapse; 47% of the participants in the study follow this pattern.

Given that roughly a half of patients suffer a worsening of their condition after each relapse, it is desirable to predict prodromal symptoms and thus prevent relapses by adjusting the doses of antipsychotic medication or other appropriate mechanisms of early intervention.

1.2 ITAREPS

The Information Technology Aided Relapse Prevention Programme in Schizophrenia presents a mobile phone-based telemedicine solution [21][19][20] for weekly remote patient monitoring and disease management in schizophrenia and psychotic disorders in general. Healthcare professionals receive alerts when the patient's condition worsens which indicates a future relapse, to enable early intervention and avoid unnecessary hospitalisations. The patient and, optionally, a family member are instructed to fill out and send via SMS (in the future via mobile application, which is currently under testing) a 10-item Early Warning Sign Questionnaire (EWSQ), which differs for patients and family members. The questionnaire consists of the items described in table 1.1.

No.	EWSQ Patient Version	EWSQ Family Member Version
1	Has your sleep worsened since the last evaluation?	Change of the sleep pattern
2	Has your appetite decreased since the last evaluation?	Marked behavioral changes
3	Has your concentration, e.g., ability to read or watch TV, worsened since the last evaluation?	Social withdrawal
4	Have you experienced fear, suspiciousness, or other uneasy feelings while being around other people since the last evaluation?	Deterioration in daily activities and functioning
5	Have you experienced increased restlessness, agitation, or irritability since the last evaluation?	Deterioration in personal hygiene
6	Have you noticed that something unusual or strange is happening around you since the last evaluation?	Loss of initiative, motivation
7	Have you experienced loss of energy or interest since the last evaluation?	Eccentric thought content, marked preoccupation with strange ideas
8	Has your capability to cope with everyday problems worsened since the last evaluation?	Marked poverty of speech and content of thoughts
9	Have you experienced hearing other people's voices even when nobody was around since the last evaluation?	Irritability, restlessness, agitation, aggressivity
10	Have you noticed any other of your individual early warning signs since the last evaluation?	Have you noticed any other individual early warning signs since the last evaluation?

Table 1.1. EWSQ 10 items for patients and family members

The participants receive an SMS alert to fill the questionnaire, which they do by replying to the SMS message with ten integer score values ordered correspondingly to the items in the questionnaire. The integer values are explained in Table 1.2.

Score	Meaning
0	No changes or improved condition
1	Slight deterioration
2	Medium deterioration
3	Significant deterioration
4	Extreme deterioration

Table 1.2. Meaning of EWSQ 10 response values

The questionnaire is designed to monitor changes in the patient’s condition on a weekly basis. Should the participant-submitted response values cross a posited threshold, the patient’s psychiatrist is notified by e-mail. The psychiatrist, on the basis of the Early Intervention Algorithm, proceeds to increase the patient’s dose of antipsychotic medication by 20% for a twenty-four hour period—an approach shown to be effective [20].

After the issued alert, a three-week alert period follows in which the participant is prompted to fill the questionnaire twice per week. If the threshold is crossed in this period, the alert period is extended by three more weeks, otherwise the psychiatrist is notified that the patient’s state is not deteriorating and that his/her dose of medication can be lowered to usual levels.

Chapter 2

Methodological Review

This chapter provides an overview of the methods used in analysis, data preprocessing, feature and classifier modelling, and performance evaluation of the task at hand. The main environment used to carry out the work is MATLAB; other used libraries will be listed throughout this chapter.

2.1 Principal Component Analysis

Principal component analysis (PCA) is a procedure that orthogonally transforms a set of vectors to uncorrelated principal components [15] [7]. It can be used to reduce dimensionality with minimal information loss and for noise reduction. In other words, PCA is designed to compute the dimensions of largest variance in a multi-dimensional space, which is often used in data preprocessing to reduce dimensionality and redundancy of the dataset, while still maintaining most of the original data variance.

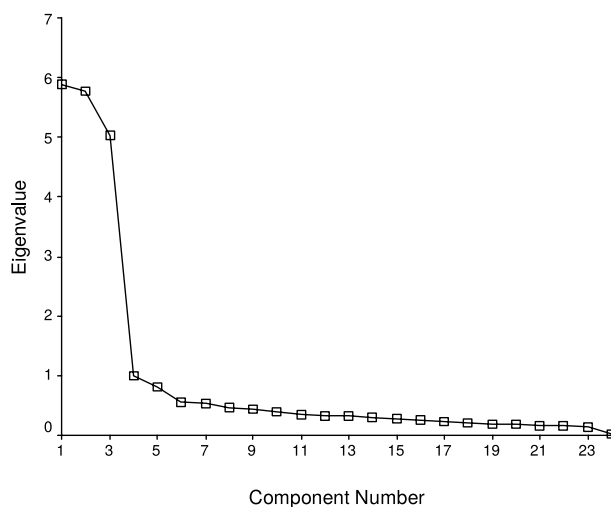


Figure 2.1. An example of a scree plot, where the principal components and their respective eigenvalues are plotted [9]. The principal components with higher eigenvalues contribute to larger variance.

As an optimisation problem, PCA can be algebraically defined [24] as the task of finding a linear subspace $X \subseteq \mathbb{R}^n$, $\dim X = k$, such that the sum of squares of distances of the points $\mathbf{a}_1, \dots, \mathbf{a}_m \in \mathbb{R}^n$ to the linear subspace X is minimal. Practically, PCA is done by singular value decomposition of $\mathbf{A}^T \mathbf{A} = (\mathbf{U} \mathbf{S} \mathbf{V}^T)^T \mathbf{U} \mathbf{S} \mathbf{V}^T = \mathbf{V} \mathbf{S}^T \mathbf{S} \mathbf{V}^T$, where $\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_m]^T \in \mathbb{R}^{m \times n}$. $\mathbf{S}^T \mathbf{S} \in \mathbb{R}^{n \times n}$ is a matrix where, after ordering the diagonal elements—the eigenvalues of the matrix $\mathbf{A}^T \mathbf{A}$ —and correspondingly their respective eigenvectors in \mathbf{V} , the orthonormal base of the subspace X and its orthogonal complement X^\perp are acquired from the column vectors of the latter matrix. Using those bases, $\mathbf{a}_1, \dots, \mathbf{a}_m$ can be projected into the k -dimensional principal component subspace X .

For the task at hand, PCA will be used to project EWSQ score values to a lower-dimensional space.

2.2 Decision Tree Learning

Decision trees [13], used for the purpose of classification, are a means of representing a conditional algorithm whose goal is to split an input data set into mutually disjoint subsets which belong to different classes. Nodes represent subsets and branches represent the conditions by which subsets are split until reaching the leaf nodes of the decision tree, at which point a class label is assigned to them. Decision trees can also be used for the purposes of regression, the difference from classification being that at the leaf nodes of the decision tree a real number value, instead of a classification label, is assigned to the resulting subset. As the outcome of a learning algorithm, decision trees are an intuitive way to represent a decision model and they are easily interpretable, as is illustrated below in Figure 2.2.

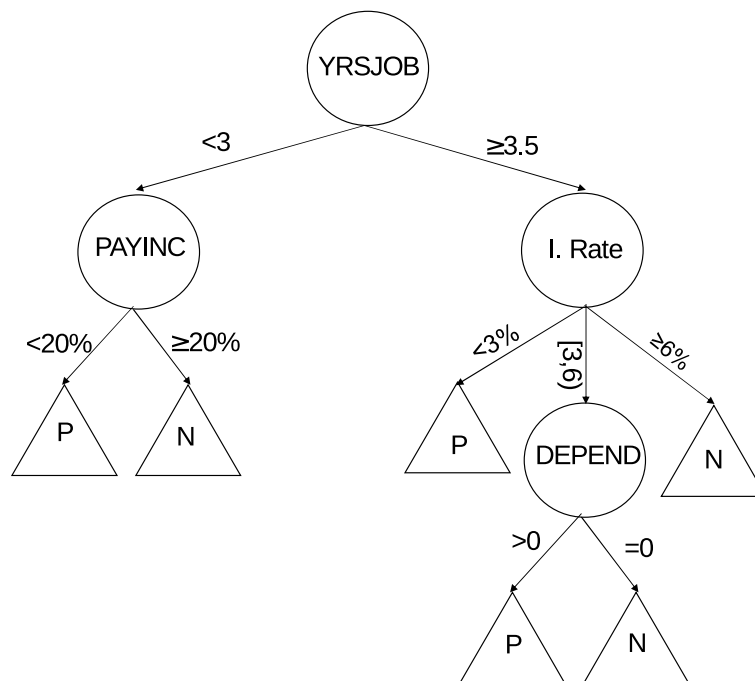


Figure 2.2. An example of a decision tree that splits a dataset into two classes—positive (P) and negative (N). Branches are labelled with conditions, internal branch nodes are labelled with the attribute names that are subjected to a condition check and the leaf nodes shaped as triangles are labelled with the class assigned to the resulting subset [13].

When learning decision trees, an important factor is choosing the criterion or metric by which conditions are selected to *best split* the set of training feature vectors, i.e. the splitting function. Another aspect of learning decision trees is that it occurs in two phases: a growth (top-down) phase, in which the splitting conditions are determined on the basis of the previously mentioned metrics, and a pruning (bottom-up) phase, in which the tree is pruned with the goal of improving some evaluation criterion, e.g. reducing the estimated error.

One such metric is information gain—a measure of the expected reduction in entropy when a set is partitioned into disjoint subsets. Given that a set P is partitioned into

disjunct subsets P_1, \dots, P_k on the basis on attribute A , the information gain $IG(A)$ can be expressed as follows

$$IG(A) = H(P) - \sum_{i=1}^k \frac{|P_i|}{|P|} H(P_i), \text{ where}$$

$$H(P) = - \sum_j \frac{|P^{(j)}|}{|P|} \log \frac{|P^{(j)}|}{|P|} \quad \text{and} \quad H(P_i) = - \sum_j \frac{|P_i^{(j)}|}{|P_i|} \log \frac{|P_i^{(j)}|}{|P_i|}$$

represent the entropy of P and P_i respectively, and $P_i^{(j)}$ represents the set of points of class j in set P_i . A better split is a split that best divides a set into more homogenous subsets, and such a split will yield higher information gain. To maximise information gain, the sum of the entropies of the resulting subsets after the split are to be minimised, i.e. the resulting splitting function $S : P, \theta \mapsto \{P_1, \dots, P_k\}$, where θ is a vector of splitting parameters, can be defined as the optimisation task

$$S = \arg \min_{\theta} \left(\frac{|P_1|}{|P|} H(P_1) + \dots + \frac{|P_k|}{|P|} H(P_k) \right).$$

Another commonly used metric in determining the splitting function is Gini purity—an impurity based criterion that measures the divergences between the probability distributions of the target attribute values [13]. It is defined as

$$GI(A) = \text{Gini}(P) - \sum_{i=1}^k \frac{|P_i|}{|P|} \text{Gini}(P_i), \text{ where}$$

$$\text{Gini}(P) = 1 - \sum_j \left(\frac{|P^{(j)}|}{|P|} \right)^2 \quad \text{and} \quad \text{Gini}(P_i) = 1 - \sum_j \left(\frac{|P_i^{(j)}|}{|P_i|} \right)^2.$$

2.3 Gradient Boosting

Gradient boosting [4] is a classification method that relies on sequentially constructing an ensemble of weak classifiers, the weighted sum of which determines the class of a given feature vector:

$$F(\mathbf{x}) = \sum_m \alpha_m h_m(\mathbf{x}),$$

where $H(\mathbf{x}) = \text{sign}(F(\mathbf{x}))$ is a classifier function which assigns a class to a feature vector \mathbf{x} , $h_m(\mathbf{x})$ is a function that represents the m -th weak classifier in the ensemble and α_m is a learned weight applied to the outcome of the weak classifier $h_m(\mathbf{x})$.

Gradient boosting is a sequential learning algorithm, because in each learning step, i.e. the latest weak classifier being learned depends on the performance of the earlier weak classifiers; and the earlier weak classifiers and their weights are not modified. Given a set of training features and their respective classes $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_i^N$, the $(m+1)$ -th weak classifier is being learned, i.e. the equations

$$\begin{aligned} F_m(\mathbf{x}_1) + h_{m+1}(\mathbf{x}_1) &\approx y_1 \\ F_m(\mathbf{x}_2) + h_{m+1}(\mathbf{x}_2) &\approx y_2 \\ &\dots \\ F_m(\mathbf{x}_N) + h_{m+1}(\mathbf{x}_N) &\approx y_N \end{aligned}$$

are being solved so as to best approximate training classes. In principle, this is done by minimising a loss function $L(y, F_m(\mathbf{x}) + h_{m+1}(\mathbf{x}))$ in order to better fit the training set

$$F_{m+1}(\mathbf{x}) = F_m(\mathbf{x}) + \arg \min_{h_{m+1}} \sum_{i=1}^N L(y_i, F_m(\mathbf{x}_i) + h_{m+1}(\mathbf{x}_i)),$$

however, this an impractical optimisation problem to solve for each iteration of the learning algorithm. Instead, a step of gradient descent is performed on the loss function, which generates a set of *pseudo-residuals*

$$g_m(\mathbf{x}_i) = - \left[\frac{\partial L(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)} \right]_{F(\mathbf{x})=F_m(\mathbf{x})} \quad \text{for } i = 1, \dots, N,$$

which constitute a new training set $\{(\mathbf{x}_i, g_m(\mathbf{x}_i))\}_i^N$ that is used to learn the $(m+1)$ -th weak classifier (e.g. a decision tree). The weight α_{m+1} applied to the weak learner is also acquired by single-variable minimisation of the loss function, i.e.

$$\alpha_{m+1} = \arg \min_{\alpha} \sum_{i=1}^N L(y_i, F_m(\mathbf{x}_i) + \alpha h_{m+1}(\mathbf{x}_i)).$$

The loss function L must be differentiable but otherwise the choice is arbitrary. In the solving of this task an exponential loss function was used with the following form

$$L(y, F(\mathbf{x})) = e^{-yF(\mathbf{x})}, \quad \frac{\partial L(y, F(\mathbf{x}))}{\partial F(\mathbf{x})} = -ye^{-yF(\mathbf{x})}.$$

The implementation of the gradient boosting method used in the solving of this task was written by Carlos Becker [2] and can be found at <https://sites.google.com/site/carlosbecker/resources/gradient-boosting-boosted-trees>. An alternative that was considered was using Matlab's Classification ensembles from its Statistical and Machine Learning Toolbox, which, although having a wider choice of options and tools, performed more slowly than the solution used.

2.4 Performance Evaluation

2.4.1 k-fold Cross-validation

Supervised learning is the task of learning a classifier using a given set of features for which the true classes are known. The classifier can be both trained and tested on the whole given set, however, since the classifier is tested on the same data it has been trained on, the performance measurement that arises from such testing does not account for the fact that the training set is almost never necessarily representative of the total set of possible features, i.e. the classifier might be overfitted to the training dataset and insensitive to data outside of it.

An approach to account for this is separating the given set of features and their classes into a training and testing set. k-fold cross-validation [10] is a technique that builds upon this approach and therefore tests the ability of a classification model to generalise to an unknown dataset.

The given finite dataset $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_i$ is partitioned, to some extent randomly, into k similarly-sized mutually disjoint folds $\mathcal{F}_1, \dots, \mathcal{F}_k \subset \mathcal{T}$, such that $\mathcal{T} = \mathcal{F}_1 \cup \dots \cup \mathcal{F}_k$. Then, for each value $i = 1, \dots, k$, \mathcal{F}_i is used as the testing set for a classifier trained on an union of all folds except \mathcal{F}_i , and a misclassification error rate ε_i is calculated for each testing fold. The resulting average misclassification error $\varepsilon = \frac{1}{k} \sum_{i=1}^k \varepsilon_i$ is a good measure of how well the classification model generalises over unknown data, especially in comparison to other classification models relative to the dataset \mathcal{T} .

Stratification within the context of k-fold cross-validation is a scheme of stratifying the folds so that they contain approximately the same proportions of labels as the original dataset [10]. Stratification is used in the implementation of cross-validation for the task at hand, however it is limited due to the fact that data provided by individual patients is self-similar and time-dependent. In order to ensure fold independence, data for individual patients is contained in a single fold.

2.4.2 Confusion Matrix

A classifier maps features to classes. When testing a trained binary classifier, we assume that the true mapping of a set of testing features to classes is known and that the classifier's mapping can be generated. The classifier is more accurate the closer its mapping is to the true mapping. One way to measure the difference between the two is by counting *true positive (TP)*, *true negative (TN)*, *false positive (FP)* and *false negative (FN)* classifications. Positive and negative represent the two classes in binary classification, while true and false represent whether the true and trained mappings coincide or differ, respectively.

These values can be used, for example [16], when expressing the classifier's

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

or its

$$\text{misclassification rate} = \frac{FP + FN}{TP + TN + FP + FN}.$$

Another use of these values is to express the classifier's

$$\text{sensitivity} = \frac{TP}{TP + FN} \quad \text{and} \quad \text{specificity} = \frac{TN}{TN + FP}.$$

Sensitivity (or true-positive rate) can be interpreted as the proportion of correctly classified positive examples; specificity as the proportion of correctly classified negative examples.

These measures will be useful, among others, in comparing the performance of different classifiers.

2.4.3 Class Imbalance

An imbalanced classification task is such where a single class significantly outnumbers the rest. A practical problem in such tasks is that, given a class that comprises 99% of a dataset, a classifier can yield 99% accuracy by merely classifying all observations as belonging to the outnumbering class.

In tasks where the goal is detection of an outnumbered critical class, measures have to be taken to amplify the presence of the critical class. One such measure is resampling—a term covering a wide variety of methods [17] that undersample an overrepresented class or oversample an underrepresented class. A synthesis of both is applied at a certain point in the task at hand, see Section 4.2.

■ 2.4.4 Relative Feature Importance

Relative feature (or predictor) importance in the context of a gradient boosting machine [14] is a consequence of the gradient boosting learning process, which offers insight into the relative contribution of individual features towards splitting a labelled dataset. A feature with greater relative importance was selected more often to split a dataset, to split a dataset of a greater size, or both.

For the gradient boosting library at hand, relative feature importance is implemented as the sum of the count of data vectors that a feature is selected to split, across all decision trees in the ensemble, normalised to sum to one.

■ 2.4.5 Labelling Time Series

In the presented case, the events to be detected—relapses—are events in time, rather than fixed labels. In order to be able to use standard machine learning models for supervised classification, it was decided to divide time series into discrete time windows, each assigned with a class label. The issue is discussed in Section 3.3.

Chapter 3

Data Overview and Analysis

3.1 Exploring the Data Set

The available data set contains all EWSQ participant submissions from both patients and family members, including periods of patients' hospitalisations. All in all, 62002 SMS were sent by a total of 349 patients.

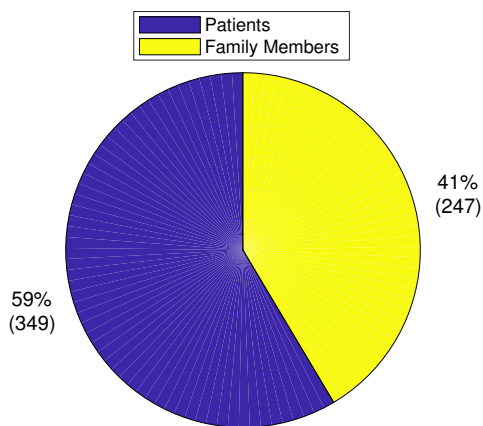


Figure 3.1. Division of participants by type.

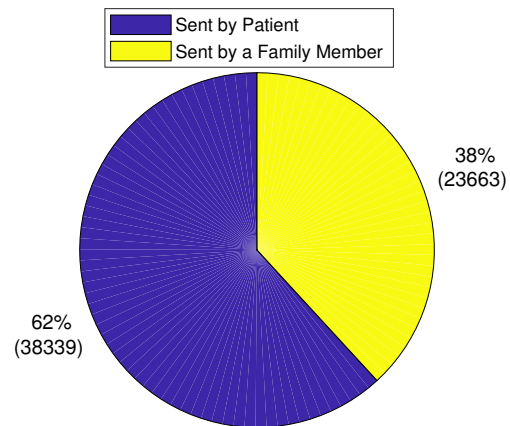


Figure 3.2. Share of all sent messages by type of participants.

Since participation of a family member is recommended but not obligatory, there are more patient participants than family members as seen in Figure 3.1 and both types send survey answers by a roughly equal rate as seen in comparison with Figure 3.2.

Times hospitalised	No. of patients
1	54
2	15
3	4
Σ	73

Table 3.1. Hospitalisation count per patient

Of all patients, 73 have been hospitalised with a total of 96 hospitalisations recorded in the programme, i.e. out of those who have been hospitalised, most patients have

been hospitalised only once, as is evident in Table 3.1. These patients will provide useful data of the EWSQ answer trend in the relapse period preceding hospitalisation.

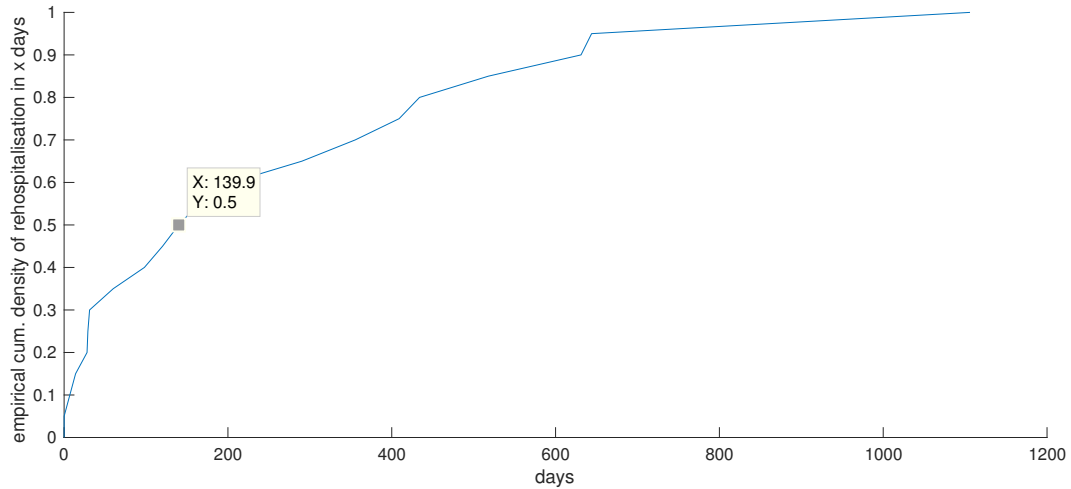


Figure 3.3. Empirical cumulative density function of number of days between rehospitalisation, gathered from the 19 patients with at least 2 hospitalisations.

By examining the empirical cumulative density of inter-hospitalisation periods, gathered from rehospitalised patients shown in Figure 3.3, it can be concluded that 50% of such patients have less than cca. 140 days or 20 weeks between hospitalisations. This trend may be useful in better understanding individual patients’ transitions from remission into relapse, especially when this occurs within a relatively short timespan of 20 weeks, while filtering out such trends in patients whose course in the development of their illness follows a different pattern.

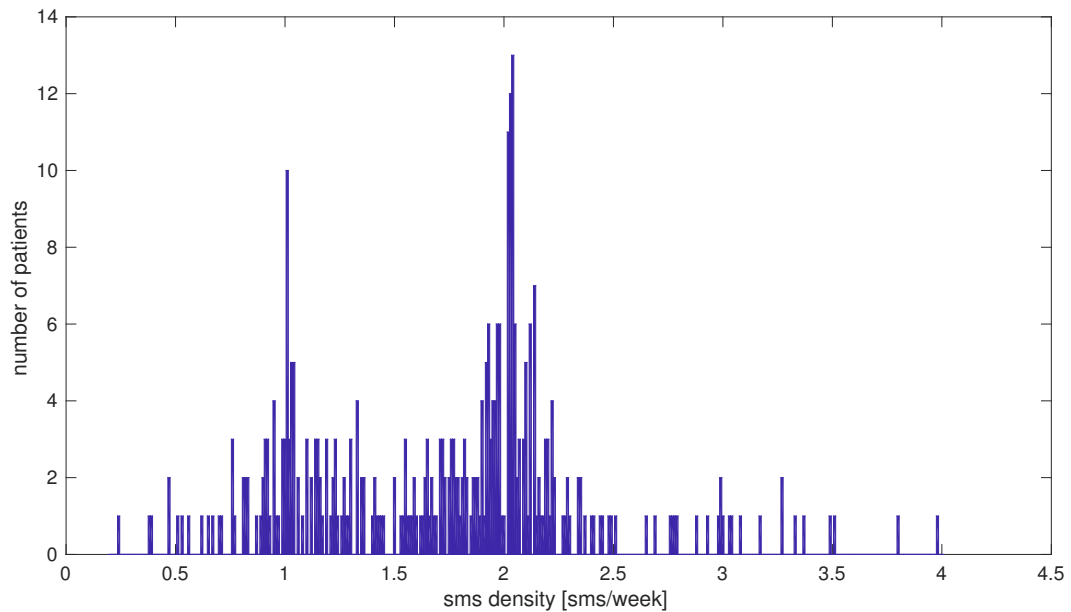


Figure 3.4. EWSQ sent SMS density per week.

Looking at the histogram of all submitted EWSQ questionnaires, including those sent by patients and family members, two clusters are clearly visible—those who sent one SMS weekly, probably patients who send messages irregularly or who do not have

family members participating in the programme, and patients and family members who send reports regularly twice per week.

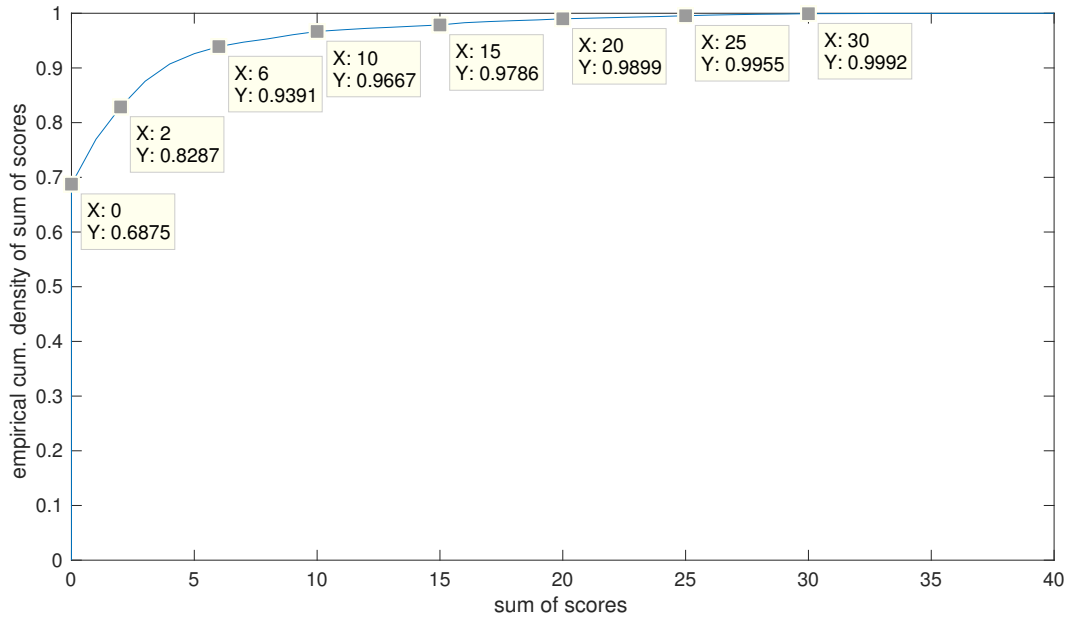


Figure 3.5. Empirical cumulative density function of the sum of all questions of each submitted SMS, including all SMS by both patients and family members. The theoretical maximum sum of scores is 40.

Looking at the scores of both patient and family member submitted questionnaire entries, one thing is immediately noticeable—most of the entries are composed of all zeros, in fact 68.75% of all entries’ scores sum to zero, as is evident in Figure 3.5. One side of this matter is that since a sum-zero SMS entry is the minimum possible value that an entry can have, we can immediately classify it as remission, were we to classify questionnaire entries as such. The other side is that by doing so, we discard more than two-thirds of our data. This provides incentive to explore strategies that consider more features than just the contents of a single questionnaire entry.

3.2 The Factor of Time

The ITAREPS programme instructs patients and family members to submit an SMS questionnaire entry once weekly if they are in a stable state or twice weekly if in an alert state of possible relapse. This influences the dataset in a way that differentiates it from that of a hypothetical study of the state of schizophrenic patients before a relapse, as it contains the self-corrective factor that when the current classifier indicates an alert state, the patient receives an increase in the dose of antipsychotic medication, which might defer a possible relapse: the easily detectable relapses, marked correctly by the current classifier and successfully treated are thus not a part of our dataset.

An attempt could be made to account for this self-correcting feedback loop by using the current classifier to mark the points in which a patient received an increase of dose in their medication and treat these points in the time series of submitted questionnaire entries as *pseudo-relapses*, which might enrich the training set. Aside from this factor, the dataset is linear in time.

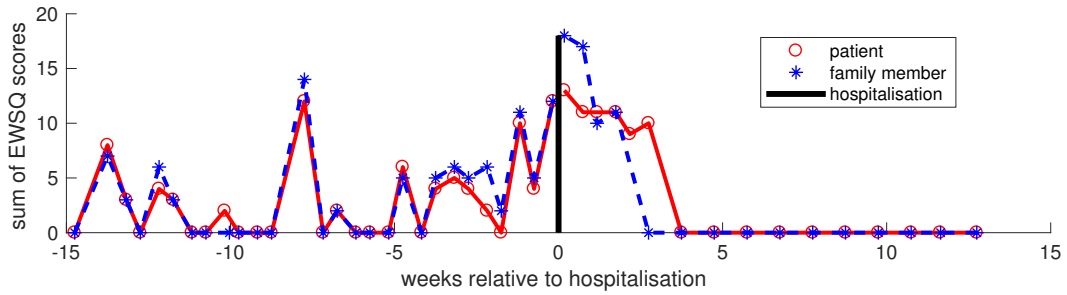


Figure 3.6. Timeseries of the sum of EWSQ scores of a patient containing both the pre- and post-hospitalisation period. There is a rise of the sum before the relapse and a fall some time after the relapse as the patient stabilises.

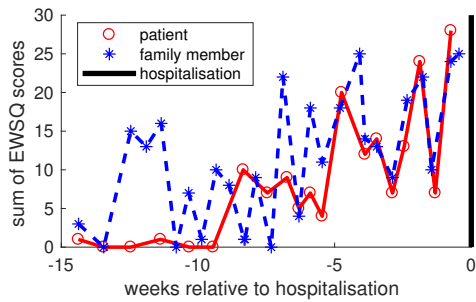


Figure 3.7. Varying growth of both patient and family member score sums before relapse.

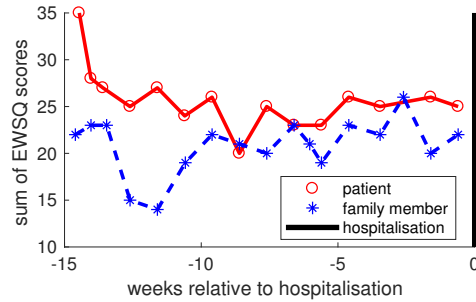


Figure 3.8. A varyingly constant sum of scores in both patient and family member before relapse.

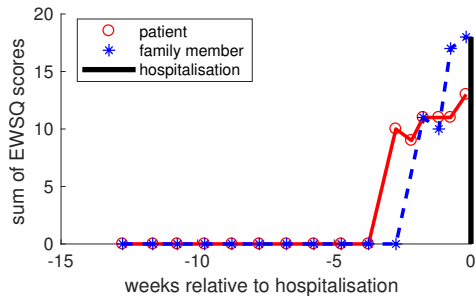


Figure 3.9. A sudden increase in sum of scores before relapse.

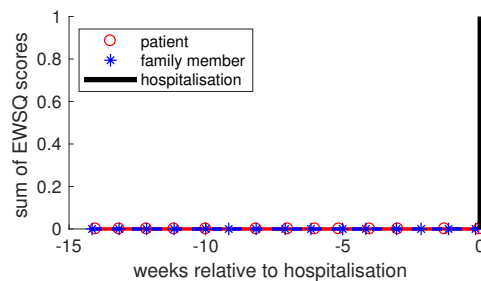


Figure 3.10. Constant sum-zero in both patient and family member before relapse.

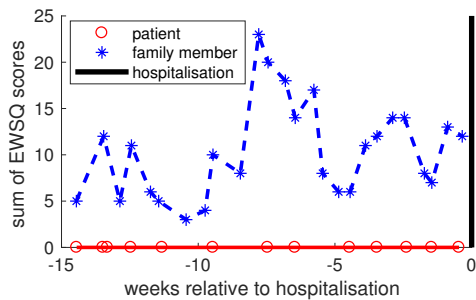


Figure 3.11. Continuous sum-zero in patient and higher score in family member entries.

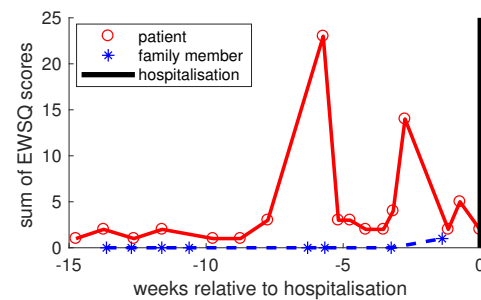


Figure 3.12. Almost continuous sum-zero in family member's submissions and a higher score in patient entries.

Out of all patients, only 73 have been hospitalised at least once. For the purposes of learning a classifier, we will consider them to have experienced a relapse and the rest to be in uninterrupted remission. Since the point of interest is the relapse, the questionnaire entry data should be considered relative to the point of relapse in time, i.e. all patients that have been hospitalised at least once provide us with two time-series of data per hospitalisation: pre-relapse and post-relapse data. In Figures 3.6–3.12 seven examples of such time-series are shown where both patients and family members participated. The presented examples document the high variability in patient and family member behaviour that needs to be taken into account when performing feature extraction for training.

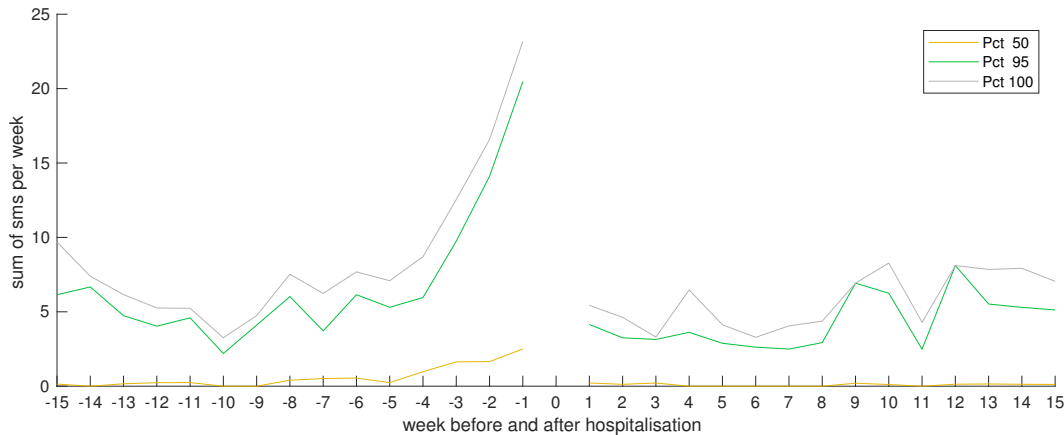


Figure 3.13. The development of the weekly sum of EWSQ scores before and after relapse in percentiles. Zero on the x-axis represents the event of relapse, i.e. hospitalisation.

According to a follow up paper on the ITAREPS study [25], the worsening of symptoms is evident in the EWSQ score as early as two months before a relapse occurs. The percentile graph as seen on Figure 3.13 portrays this trend of growth of the sum of scores before the relapse, and the stabilisation of the patient after the relapse, i.e. after discharge from hospitalisation of varying length.

The 50th percentile, i.e. the median, is nearly zero, which reflects the factor in the dataset discussed before, that two-thirds of all sums of scores are zero, even before relapse. This could be accounted for by a combination of any of the following possible factors: the lack of adherence of some patients to the study guidelines, to a possible paranoid delusion related to the disclosure of such personal data to a third party, or in the worst case, the diminished capacities of self-reflection in pre-relapse schizophrenic patients. In the third case, the family member’s scores might prove to be useful over the patient’s self-evaluation, as is apparent by the mean score value shift by cca. one in Figure 3.14.

3.3 Splitting Data in Periods

An approach in predicting relapses in patients is binary classification, i.e. either *the patient will have a relapse in a certain period of time* (e.g. n weeks) or *the patient is in remission*. In the former case, the patient is considered to be in a critical period, otherwise he/she is in remission.

In order to render differences in the individual developments of patients’ EWSQ scores negligible, padding or *safe periods* can be introduced between the pre-relapse and

remission periods, namely: a pre-critical safe period, in which the patient transitions from a remission to a pre-relapse state, and a post-hospitalisation safe period, in which the patient, having been until recently hospitalised and medicated, might continue to be under this extraordinary influence that otherwise differs from his/her usual state.

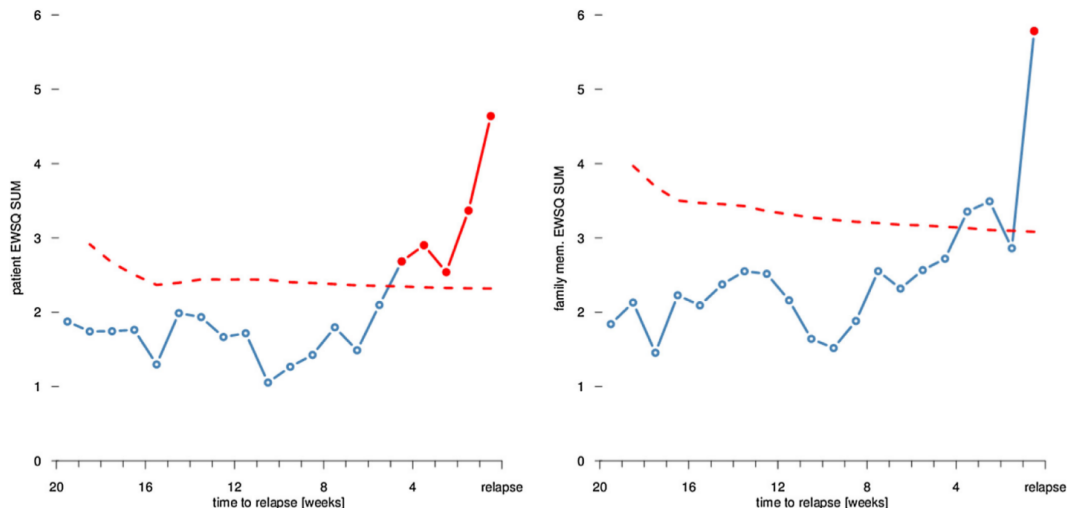


Figure 3.14. Simple visualization of prodromal onset time point detection results for mean EWSQ sum score for patients (left) and family members (right). The solid line represents weekly population mean (significant region in red), the dashed line indicates the 95th percentile of the resampled bootstrap population means. (Source: [25]) The method chosen for colouring the solid line representing the weekly population mean was, that all adjacent weekly data points and the lines connecting them are coloured red if they are continuously above the 95th percentile line.

According to a recent study on ITAREPS data [25], the mean sums of weekly EWSQ scores achieve statistically significant increase four weeks before relapse, as is seen in Figure 3.14. The beginning of the critical period can thus be safely set as late as four weeks before relapse. It is also noted in the study that for the individual EWSQ questionnaire entries as well as for their sum, there are observable long-term rising trends that range up to nine weeks (for individual EWSQ questions) or eight weeks before relapse (for the sum). Therefore, the critical period used for learning the classifier can be safely set at some point between eight and four weeks before relapse.

For the purpose of training a classifier, the period lengths as described in Table 3.2 have been chosen.

Name of Period	Interval in Weeks	EWSQ Entry Count
Remission	$(-\infty, -10]$	9950 (16.05 %)
Pre-critical Safe	$(-10, -5]$	610 (0.98 %)
Critical	$(-5, 0]$	713 (1.15 %)
Post-relapse Safe	$(0, 5]$	535 (0.86 %)
Remission	$(5, \infty)$	9950 (16.05 %)
No-Hosp. Remission	n/a	50091 (80.79 %)
Discarded	n/a	103 (0.17 %)

Table 3.2. Separation of hospitalised patients’ relapse-relative EWSQ score timeline in periods to be used for learning a classifier. Relapse occurs at the zeroth week in the timeline, i.e. when the patient is hospitalised. The pre-relapse and post-relapse remission periods are considered to be as a single category with the same entry count. Percentage of the whole dataset are given in parentheses after the count value.

Considering patients’ timelines relative to the date of relapse and attempting to partition it in fixed-length time periods implies that there exist EWSQ entries in the dataset that can be considered to belong to more than one time period. E.g. supposing a patient is released from hospital care, in the first week he/she submits an EWSQ entry and is rehospitalised the next week—does the submitted EWSQ entry fall into the critical period or the post-relapse safe period? In order to avoid these ambiguities, all such data is discarded.

As was already stated and evident in Figure 3.5, sum-zero entries—EWSQ entries whose scores sum to zero—amount to 68.75 % of the whole dataset. Having the dataset partitioned into discrete periods, the share of sum-zero entries in all periods can be inspected.

Period Name	Critical	Remission Hosp.	Remission No-Hosp.
Total	713	9950	50091
Sum-Zero Count	346	6487	34992
Percent of Whole	48.53 %	65.20 %	69.86 %

Table 3.3. Share of all EWSQ entries in the dataset that contain only zeros per period.

As is evident in Table 3.3, the critical period contains a lesser share of sum-zero entries than the other categories used for classification, which is in line with expectations: when patients feel worse they are less likely to submit an EWSQ entry with all zeros.

Furthermore, two conclusions can be made from this observation. The first is that if only the ten EWSQ score values are used to classify it into a period, sum-zero entries will always be classified as *in remission* and the a priori misclassification rate for the critical period will be roughly at least a half. The second conclusion is that the two periods are unbalanced in size, i.e. there is significantly more data for remission than for the critical period. This is why over-fitting to the majority class should be avoided and monitored with methods such as cross-validation and either resampling of critical period data or random undersampling of remission data should be undertaken.

3.4 Applying PCA

Since the dataset has been separated into classes and the critical period and remission period data has been labelled, principal component analysis can be applied to the dataset.

From previous studies on the ITAREPS programme [25][8] and as is shown in Figure 3.15, a single component contributes to a majority of variability in the dataset. In order to test this hypothesis, i.e. that a single component represents responses from the remission and critical period, the component with the greatest contribution to variance begins to lose its major role to the benefit of other components. Learning a classifier based on the first several principal components may thus bring increase in classification

accuracy if this assumption were true. Thus, the principal component subspace will be generated using remission period data and relapse period EWSQ score data will be projected to the remission principal component subspace using the parameters acquired from generating the principal component subspace.

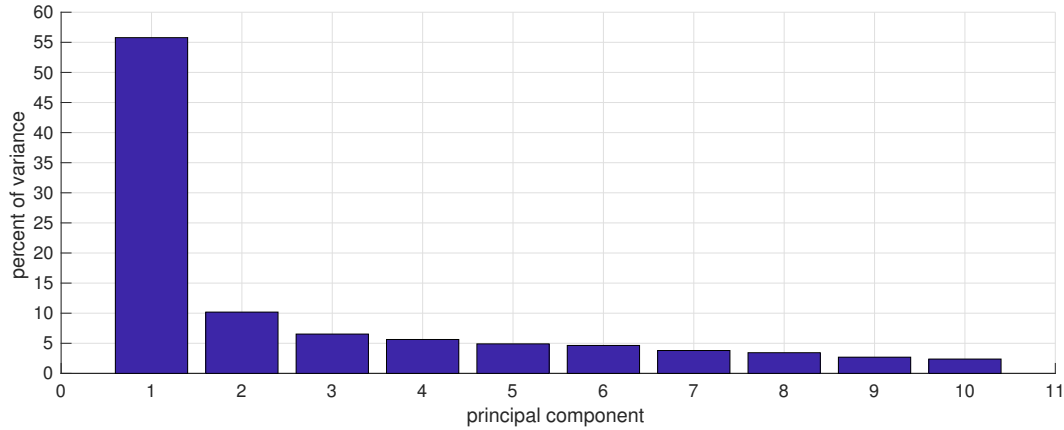


Figure 3.15. Contribution by percentage to variance by each principal component acquired by applying PCA to remission period data.

As is evident in Figure 3.15, the first principal component accounts for more than 55% of the variance in remission data and the rest individually contribute 10% and less. Looking at the weights applied to score values when projecting them to a dimensionally reduced PC subspace in Table 3.4, the first component (the component most contributing to variance) applies a nearly uniform vector of weights to the score values. The second principal component is already more peculiar in that it applies negative weights to most score values except the first and the last two question scores, which are arguably in themselves good predictors of the worsening state of a schizophrenic patient (worsening of sleep, auditory hallucinations and general worsening of symptoms). The second principal component also diminishes the impact of the second question (loss of appetite) by applying a near-zero weight of -0.0012 to its score value.

Question no.	Principal component				
	First	Second	Third	Fourth	Fifth
1	0.3102	0.5084	0.6889	-0.1239	-0.3761
2	0.2419	-0.0012	-0.0939	0.0037	-0.0633
3	0.3125	-0.2091	-0.0560	0.0622	0.0617
4	0.3603	-0.2574	-0.1855	0.1050	-0.4834
5	0.2780	-0.2997	0.0908	0.2186	-0.1460
6	0.3266	-0.0712	-0.2627	0.0082	-0.1708
7	0.3484	-0.2436	0.3303	-0.2658	0.6552
8	0.3406	-0.2659	0.1201	-0.1125	0.0541
9	0.3262	0.4649	-0.5151	-0.5365	0.0919
10	0.3002	0.4365	-0.1133	0.7421	0.3549

Table 3.4. Coefficients or weights that make the basis of the principal component subspace. When projecting EWSQ score vectors to the subspace, weights are individually multiplied with the appropriate weight and summed, and as such their position vector in the principal component space is determined. For question numbers, see Table 1.1.

Chapter 4

Preparation for Learning

4.1 Feature Extraction

Considering what other features to use, this section will follow the principle of *more is better*, since the learning technique used is boosting of decision trees, which rely on the dataset being well separated by introducing splits dependent on the features. Thus more features enable more potential data splits and therefore a potentially better separated dataset. Another reason for the necessity of augmenting the feature set is that, as is apparent in Table 3.3, a large portion of data from the ten questions of the EWSQ consists solely of zero scores and as such can only be immediately classified into the remission class, thus by adding more features, sum-zero entries will be more meaningfully separable.

The data acquired from the ITAREPS study includes two data tables: one where each row represents a submitted SMS with EWSQ entries and the other where each row represents a patient's hospitalisation with some additional metadata. The classifier will be based on individual SMS entries as feature vectors, so for every feature vector, the following usable data is made available:

- **EWSQ score values:** either the ten score values as they are or by projecting them to an uncorrelated and possibly lesser-dimensional space using PCA.
- **sum of EWSQ score values:** in the state of the art classifier used to indicate an alert state, not only are the individual scores considered but also their sum, to account for possible general fluctuations in a single SMS entry, that may not be accounted for, given that decision trees split values only depending on the value of a single feature.
- **is the SMS sent by a patient or a family member:** a binary categorical variable that can separate the dataset in two within a decision tree, which will be useful since, as evident in Figures 3.6–3.12, patient and family member submitted data may and do differ and this can provide useful insight into the patient's condition timeline.
- **time since the previous hospitalisation:** patients who have been previously hospitalised, i.e. have experienced a relapse, might have a greater probability of repeated relapse and this might again aid in better separating the dataset to perhaps more easily classify such patients as in an alert pre-relapse state. This variable is either a continuous real value, i.e. elapsed days since the previous hospitalisation or a null type value, such as NaN in MATLAB.

4.1.1 Accounting for History

Since a patient's data is given in time and the definitions of the questions in the EWSQ are related to the history of the patients condition—the general formulation in the questionnaire always references the patient's state in the past with constructions in the mode of *have you noticed something new since the last evaluation*, as can be seen

in Table 1.1—the patient’s history has to be accounted for and it should lead to an improvement in accuracy as opposed to a past-ignorant feature set. One variable that accounts for time since the previous hospitalisation has already been described above, however this feature does not provide sufficient separability on the basis of a patient’s history, since it together all unhospitalised patients categorises—276 out of a total of 349—that make up a large share of the whole dataset of questionnaire values.

One problem of accounting for history of a patient’s EWSQ entries is in a limitation of the classification method, i.e. that it does not support timeseries of data by design and thus the feature vector length must be limited and equal for all entries in the classification set. Assuming a simple approach of a single feature corresponding to a single submission, this would mean that, if patient A has a history of 5 entries and patient B has a history of 100 entries, the feature vector could either account for only five entries, and therefore ignore patient B’s previous 95 entries, or account for all of patient B’s entries and fill patient A’s history with 95 counts of null, missing data values (represented by NaN in MATLAB), which might flood the decision tree boosting process and feature space with features irrelevant for most patients and require an impractical increase in the number of boosted trees and thus increase training time and hinder optimisation efforts.

The solution to this is to use the first strategy, which effectively means that each feature vector based on a submitted EWSQ SMS will contain a trail of n previous submissions, i.e. their EWSQ score sums. This means that the first n will have less than n entries in their history, and they could therefore be removed from the dataset, which practically means that relapse detection will not occur in the first n entries (practically first n weeks of the patient’s participation in the programme) and they will be automatically classified to be in remission. Because of the differences between submissions sent by patients and family members, the historical trail should reflect this and be kept separate for patients and family members. The value that will be used in training is $n = 3$. Given that value and the method described above, 2171 out of a total of 62002 (3.5%) submissions are discarded from the dataset on the basis of not having at least $n = 3$ previous trailing SMS.

For each i -th SMS, there is a value S_i —the positive integer sum of the ten EWSQ scores submitted in the SMS, and the values $S_{i-1}, S_{i-2}, S_{i-3}$ —the positive integer sums of the ten scores from the trail of three previous SMS. The three trail sum values can be introduced as features as such, however, when interpreting the meanings of the questions in Table 1.1, since the questions refer to the patient’s previous state, it would be better to introduce cumulative sums for each trailing SMS in the manner below:

$$\begin{aligned}
 \text{trailing cum. sum 1} &:= S_{i-3} \\
 \text{trailing cum. sum 2} &:= S_{i-3} + S_{i-2} \\
 \text{trailing cum. sum 3} &:= S_{i-3} + S_{i-2} + S_{i-1} \\
 \text{trailing cum. sum 4} &:= S_{i-3} + S_{i-2} + S_{i-1} + S_i
 \end{aligned} \tag{1}$$

where each row represents the value of a single feature.

Another use of the trailing sum values could be calculating the difference between consecutive SMS in the manner below:

$$\begin{aligned}
 \text{trailing cons. diff. 1} &:= S_{i-3} - S_{i-2} \\
 \text{trailing cons. diff. 2} &:= S_{i-2} - S_{i-1} \\
 \text{trailing cons. diff. 3} &:= S_{i-1} - S_i
 \end{aligned} \tag{2}$$

where each row represents the value of a single feature. Since the trailing features represent change of the patient’s state between submissions, an interpretation of including the difference of consecutive sums would be to signify the rate of change of the patients state between consecutive responses.

■ 4.1.2 Resulting Feature Vector

The considerations above result in the structure of the feature vector that will be used for learning and prediction described in Table 4.1.

Order	Data type	Description
1–10	integer or real (PCA)	The ten EWSQ score values. If PCA is applied and the score values are dimensionally reduced, then there will be less than 10 values in the feature vector.
11	integer	Sum of the ten EWSQ score values.
12–15	integer	Trailing SMS cumulative sums, see (1).
16–18	integer	Trailing SMS consecutive differences, see (2).
19	categorical	Was the SMS sent by the patient or a family member?
20	real or NaN	Days since the previous hospitalisation if any, else NaN.

Table 4.1. Final layout of the feature vector.

■ 4.2 Resampling

As can be seen in Table 3.2 in the column *EWSQ Entry Count*, the dataset is significantly unbalanced having cca. 85 times more data for the remission class as opposed to the critical class. The boosting decision tree method used for training a classifier in this task is sensitive to unbalanced data, so an effort is to be made to equally represent both classes in the training dataset.

Given that the remission class is overrepresented in the dataset, random undersampling can be applied by randomly discarding elements of this class, which can balance the ratio of the classes in the dataset. The penalty of this approach is that a large amount of data is randomly discarded—the training dataset will be reduced to cca. 1500 entries.

On the other hand, relapse data can be oversampled either by simply duplicating or copying it without changes until the dataset is equally representative of both classes, or by randomly generating new relapse samples similar to the ones acquired in the study.

A synthesis of both approaches will be applied in resampling the dataset, namely, critical class data will be identically copied as many times as it is necessary until it almost reaches the same count as remission class data, and then the remission dataset will be minimally randomly undersampled so that both classes have an equal count of

entries in the dataset and therefore equal representation, as is illustrated in Figure 4.1 below.

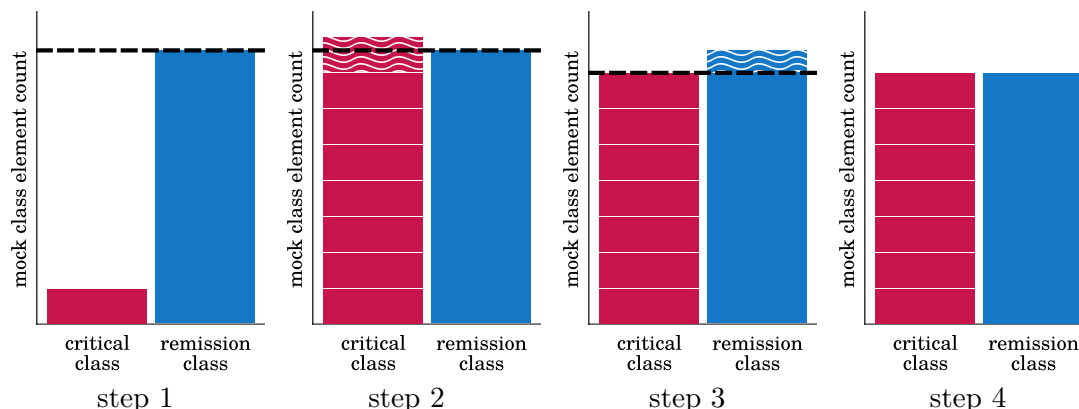


Figure 4.1. Illustration in four steps of how the classes are resampled. The bar graphs do not represent the actual size ratios of the critical and remission classes.

Due to also performing k -fold cross-validation and its requirement that critical class data for a single patient is contained in a single fold, as described in the next section, there is one further limitation placed on the resampling method, namely that resampling has to be performed only after the data has been separated into folds for cross-validation and it can be copied only a certain amount of times so not to actually exceed the count of remission data in any individual fold. After the resampling approach is applied, the effect of the transformation is described in Table 4.2.

Class	Pre-resample	Post-resample
Critical	713	44919
Remission	60041	44919
Σ	60754	89838

Table 4.2. Effect of the resampling procedure on the dataset. The critical class has been copied 62 times while the remission class has been randomly undersampled to approximately 74.81% of its data.

4.3 Cross-validation

Usually k -fold cross-validation for binary classification, the k folds are stratified so that all folds contain similar proportions of data representative of both classes. When applying cross-validation to this classification task, there is one more factor that needs to be considered to better divide data into folds, namely that a patient's pre-relapse data is not separated into different folds so that it does not occur that a classifier is trained and tested on the same patient's data, which would lend an additional opportunity for the classifier to overfit to the testing set.

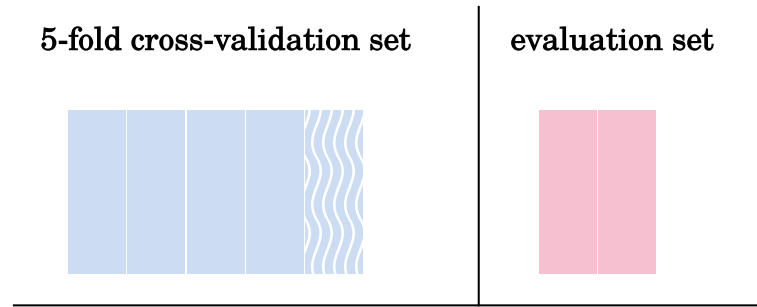


Figure 4.2. Illustration of the method of cross-validation used in the task. The dataset is divided in seven folds. The first five are used in grid search for optimal hyperparameters in which each fold is tested on exactly once. The last two folds are used in the evaluation phase to assess the optimised classifier and compare it to different ones.

Otherwise, standard k -fold cross-validation is used in this task and, assuming that resampling as described in the previous chapter has been applied. Two of the folds are not used in the training and testing phase of hyperparameter optimisation and are left for the last stage of evaluating the resulting optimised classifiers. The dataset is divided into $k = 7$ folds, since there are 73 hospitalised patients and so each fold will have data for at least 10 patients’ critical periods. Table 4.3 displays the division of data in folds.

Fold	Critical c.	Remission c.	Σ	Remission undersampled c.
1	6867	6867	13734	1844
2	6804	6804	13608	2290
3	8883	8883	17766	80
4	6489	6489	12978	1404
5	5103	5103	10206	3012
6	6111	6111	12222	2213
7	4662	4662	9324	4279
Σ	44919	44919	89838	15122

Table 4.3. Count of elements in each cross-validation fold. The sixth and seventh folds are left to be used for later evaluation of the resulting classifiers. The third fold has a count of 80 remission elements removed in undersampling, which is the reason why critical class data was oversampled only 62 times, i.e. were it copied more times, it would exceed the number of remission class elements in its fold.

4.4 Hyperparameter Optimisation

Boosting decision trees require a few parameters to be set before training, so called hyperparameters. Those that will be optimised will be:

- the **learn rate**— ν or `learnRate`—determines the impact of individual trees on the final classifier produced by the ensemble boosting model. Values range in the interval $(0, 1]$ and when the value is set too high, the trained classifier may tend to overfit to the training data and may thus fail to generalise well to previously unknown data. In the gradient boosting method, it is expressed as an additional weight that decreases the contribution of the weak classifiers, i.e. for $\nu \in (0, 1]$

$$F_{m+1}(\mathbf{x}) = F_m(\mathbf{x}) + \nu \alpha_{m+1} h_{m+1}(\mathbf{x}).$$

- the **maximum tree depth**— d or `maxTreeDepth`—constrains the depth of the decision trees used as the weak learners in the boosting ensemble, e.g. if it is set to 0, then the decision tree is actually a decision stump with only one node that, on the basis of a single condition, splits the feature set into the two classes of the binary classification problem.
- the **number of trees** for learning rate ν — M_ν or `numOfTrees`—is the number of weak classifiers being boosted in the learning ensemble. The number of trees should usually increase when decreasing the learn rate to counter overfitting and bad generalisation on previously unknown data.
- the **subsampling factor** determines a percentage of observations that are randomly sampled in training each tree. It helps reduce variance, i.e. overfitting.
- the **number of features considered for splitting a node**—`mtry`—reduces the number of features and may influence feature importance.

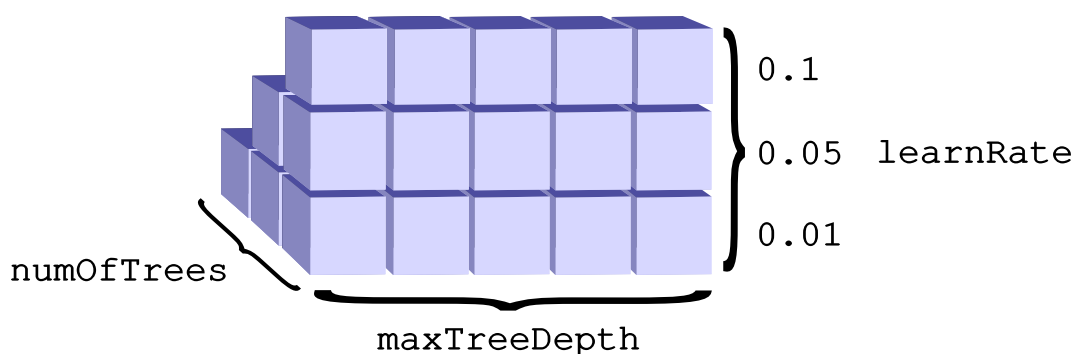


Figure 4.3. An illustration of the grid search method used in hyperparameter optimisation. The number of cubes for `numOfTrees` does not represent the actual value of the parameter but is simplified to reflect that for lower learning rates a higher maximum `numOfTrees` value is considered.

The following approach to optimising these parameters for the task at hand will be used. The dataset is divided into seven folds and the first five are used for cross-validation, i.e. for performing a grid search to find the optimal values for `maxTreeDepth` = 0, 1, 2, 3, 4, `learnRate` = 0.1, 0.05, 0.01 and a range of `numOfTrees` respectively for each learning rate `numOfTrees` = 1..3000, 1..5000, 1..7000 as illustrated in Figure 4.3. The metric for determining the optimal value of the first two parameters is the minimum misclassification rate across all values of `numOfTrees`, averaged for all five cross-validations. The misclassification rate is implicitly balanced for both classes, given that the folds have been resampled to contain an equal count of relapse and remission data. The number of features considered for splitting a node is fixed: `mtry` = `ceil(sqrt(number of features))` and `subsample` = 0.8.

In order to further clarify the employed method for grid search optimisation of the two parameters, the previous paragraph will be expressed in notation:

$$\nu^*, d^* = \arg \min_{\nu, d} \min_m \bar{\varepsilon}_m(\nu, d), \quad 1 \leq m \leq M_\nu,$$

where $\bar{\varepsilon}_m(\nu, d)$ is the mean misclassification rate evaluated on 5 cross-validation folds, i.e.

$$\bar{\varepsilon}_m(\nu, d) = \frac{1}{5} \sum_{i=1}^5 \varepsilon_{m,i}(\nu, d),$$

where $\varepsilon_{m,i}(\nu, d)$ is the misclassification rate evaluated on the i -th cross-validation fold (trained on the four remaining folds) for a count of m weak learners (decision trees), given `learnRate` = ν and `maxTreeDepth` = d .

The resulting values ν^*, d^* acquired by grid search are finally used to train a classifier with `numOfTrees` = M_{ν^*} .

4.5 Classifier Model Variants

The approach described in the previous section will be applied to three different feature sets, i.e.

- All the features as described in Table 4.1, referenced below as *No PCA*.
- The features as described in Table 4.1, except with the EWSQ scores projected to a $k = 5$ dimensional principal component space.
- The features as described in Table 4.1, except with the EWSQ scores projected to a $k = 10$ dimensional principal component space.

For each feature set there will be two resulting classifiers:

- *single*—one trained on the whole training dataset and evaluated on the whole evaluation set, and
- *dual*—a compound classifier composed of two classifiers, each trained exclusively on patient and family member data and evaluated on the respective evaluation data sets, in order to guarantee that the disparity between patient and family member datasets is captured in the classification model.

Additionally, for each feature set, there will also be provided a fitted decision tree for reference with the same evaluation measurements applied to it as are to the gradient boosting ensembles as well as a single ensemble classifier, which does not contain historical features—trailing sums.

4.6 State of the Art Classifier

Additionally to the trained classifiers, the state of the art classifier (henceforth referred to as `alertInds`) used in the ITAREPS programme is introduced to compare with the trained classifiers. It is again worth noting, as was already done in Section 3.2, that the dataset used for training and evaluation has *influenced itself* via the `alertInds` classifier, as patients who participated in the programme may have received an increase in their dose of antipsychotic medication upon being classified as in an alert state by `alertInds`. Thus the remission class might contain entries that, without the intervention of `alertInds`, would have resulted in a hospitalisation and thus would have actually belonged to the critical class. Until a dataset without such an influence is provided, any classifier trained on it will be most useful as a secondary classifier to `alertInds`.

Another consideration worth noting when comparing `alertInds` to the trained classifiers is that `alertInds` was modelled several years ago when far fewer data were available from the ITAREPS programme.

The `alertInds` classifier classifies based on a disjunction of three conditions which rely on three parameters: whether an entry was submitted by a patient or family member, time difference between entries and EWSQ score sums. Firstly, if the sum of scores 4, 6, 9 for a patient submitted entry or 2, 7, 9 (see Table 1.1) for a family member submitted entry is greater than 3, then the patient is classified as in an alert state. Secondly, if the sum of all scores exceeds 8, the patient is classified as in an alert state. Finally, if the sum of scores of each entry and its preceding (though not more than 14 days apart) entry exceeds 12, the patient is classified as in an alert state.

Chapter 5

Results and Discussion

This chapter follows the training of several classifiers optimised with the method described in the previous chapter and evaluated on the evaluation dataset.

5.1 Classifier Evaluation

As is evident in Table 5.1, the classifier models which detected all detectable relapses when predicting on the evaluation set are the single models with no principal component dimension reduction applied to the EWSQ scores. In comparison with `alertInds`, other than the notable difference in relapses predicted from patient and family member data, there is also a considerable difference in the false positive rates.

The gradient boosting classifier has a higher false positive rate and lower accuracy that correspond to a trade-off between sensitivity and specificity. The `alertInds` classifier has half the sensitivity of the gradient boosting classifier while having a higher specificity rate by cca. 10%. The necessity behind the existence of the sensitivity–specificity trade-off is due to both classes not being fully separable and also because one of the goals of constructing the gradient boosting classifier was to increase the sensitivity of detecting the critical class, for which there is a low amount of data—1.15% of the whole dataset, see Table 3.2. This was accomplished by optimising the model’s hyperparameters on the *balanced* misclassification rate, produced by measuring it on a resampled testing set, and then learning the model on a resampled training set, where the critical class had been oversampled and the remission class undersampled.

The classifiers in which EWSQ score features were projected to principal component space yielded interesting results. When the EWSQ scores are projected to a $k = 5$ dimensional principal component space, the sensitivity of the classifier further increases, while given $k = 10$, the sensitivity decreases and the predicted relapse counts approach those of `alertInds`.

Classifier and feature model			Confusion matrix and derived metrics						Predicted relapses			
S/D	PCA dim	Type	History	TP	TN	FP	FN	Sens.	Spec.	Accuracy	Patient †	Family member ‡
Single	No PCA	Ens.	Yes	54	14123	2661	104	34.2 %	84.1 %	83.7 %	12 (52.2 %/100 %)	5 (21.7 %/100 %)
Single	5	Ens.	Yes	58	13825	2959	100	36.7 %	82.4 %	81.9 %	11 (47.9 %/91.7 %)	5 (21.7 %/100 %)
Single	10	Ens.	Yes	38	14930	1854	120	24.1 %	89 %	88.4 %	7 (30.4 %/58.3 %)	3 (13 %/60 %)
Dual	No PCA	Ens.	Yes	48	14259	2525	110	30.4 %	85 %	84.5 %	12 (52.2 %/100 %)	4 (17.4 %/80 %)
Dual	5	Ens.	Yes	45	14761	2023	113	28.5 %	87 %	87.4 %	11 (47.9 %/91.7 %)	5 (21.7 %/100 %)
Dual	10	Ens.	Yes	40	14786	1998	118	25.3 %	88.1 %	87.5 %	12 (52.2 %/100 %)	2 (8.7 %/40 %)
Single	No PCA	Tree	Yes	34	15339	1445	124	21.5 %	91.4 %	90.7 %	5 (21.7 %/41.7 %)	2 (8.7 %/40 %)
Single	5	Tree	Yes	37	15360	1424	121	23.4 %	91.5 %	90.9 %	6 (26.1 %/50%)	4 (17.4 %/80 %)
Single	10	Tree	Yes	35	15399	1385	123	22.2 %	91.7 %	91.1 %	6 (26.1 %/50%)	2 (8.7 %/40 %)
Single	No PCA	Ens.	No	55	14028	2756	103	34.8 %	83.6 %	83.1 %	12 (52.2 %/100 %)	5 (21.7 %/100 %)
		alertInds		27	16002	762	131	17.1 %	95.5 %	94.7 %	5 (21.7 %/41.7 %)	2 (8.7 %/40 %)

Table 5.1. † Patient predicted relapses and ‡ family member predicted relapses out of a maximum of 23 each. Two percentage values are given in parentheses. The first indicates the share of total relapses predicted, i.e. at least one respective patient or family member submission was classified as critical. The second percentage value indicates the share of all *predictable* relapses that have been predicted by the classifier. A relapse is predictable if at least one EWSQ SMS submitted in the pre-relapse critical period contains at least one value greater than zero. There are 12 out of 23 predictable relapses for patient submitted EWSQ scores and 5 out of 23 relapses for family member submitted scores in the evaluation set. The boldened rows represent the best performing classifiers (evaluated by the predicted relapses) as well as the state of the art `alertInds` classifier.

5.2 Feature Evaluation

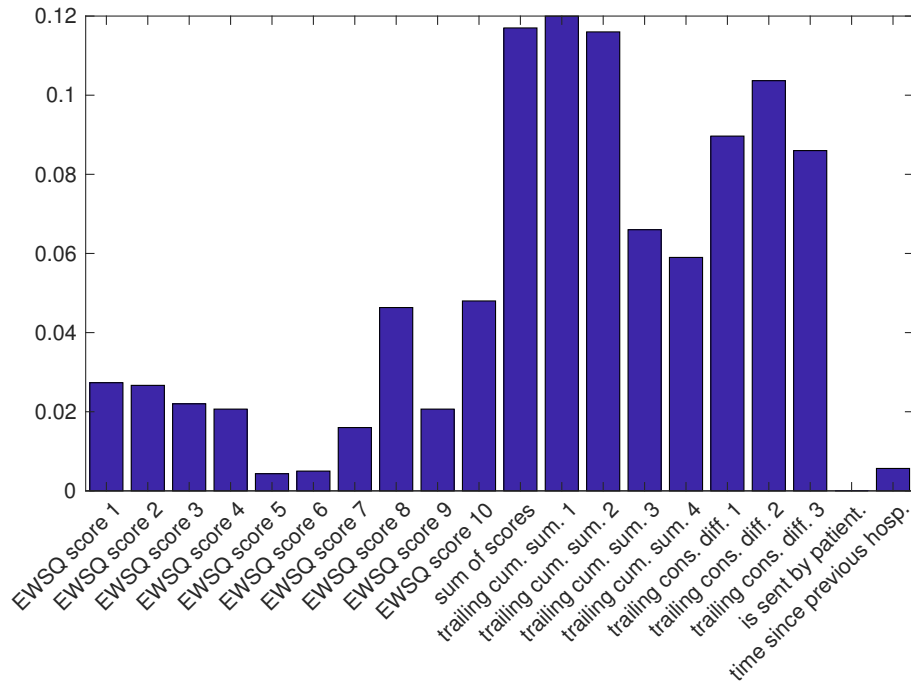


Figure 5.1. Relative predictor (feature) importance for the final classifier. In terms of classification with a gradient boosting machine, relative importance for a feature can be interpreted [1] as the mean count of observations, summed across all decision trees in the ensemble, that reach a non-leaf node which splits the observation set on the basis of the feature.

The dual classification models, which rely on training and then using for prediction two gradient boosting classifiers separately for EWSQ score submitted by patients and family members, exhibit weaker performance than the single classifiers and seemingly do not contribute anything more than the single classifiers. Moreover, when inspecting the relative predictor importance rates for each feature in the gradient boosting classification model, as can be seen in Figure 5.1, it is evident that the feature extraction approach to account for the patients EWSQ score history, described in Subsection 4.1.1, was effective and significantly contributed to data separability in the gradient boosting classification model, so much so that the categorical predictor *is sent by patient* was not significantly employed in separating the dataset in decision tree learning, possibly due to low information gain. This stands in contrast to the difference between the questions offered to patients and family members in the questionnaire that share the same order.

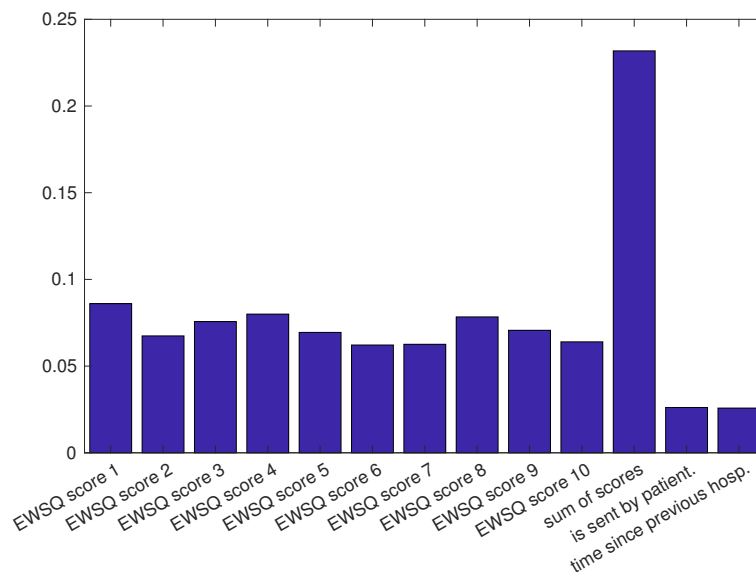


Figure 5.2. Relative predictor (feature) importance for the classifier that does not account for the trail of historical data.

However, in opposition to the enthusiastic claim in the previous paragraph, stands the evaluation of the classifier that does not take account of the patient’s previous EWSQ submissions—see the penultimate row and compare with the first row in Table 5.1. When the historical trails generated by previous submissions are removed, the boosting ensemble compensates by relying more on individual EWSQ scores and the previously ignored *is sent by patient* predictor, as can be seen in Figure 5.2. While both approaches yield similar results on the given evaluation set, which might lead to a claim towards the redundancy of historical data, the no-history classifier still relies mostly on the *sum of scores* predictor for data separability as opposed to individual EWSQ score predictors. Comparably, the classifier that takes history into account (Figure 5.1) further relies mostly on the sums of scores not only of the current but even more so of the previous submissions, in comparison to all other non-sum features. This implies that in both cases, sums of scores yield greater relative importance than other types of predictors, and assuming that having more predictors that lead to better separability, as opposed to less, yields a better classifier in the long run, then the inclusion of and reliance on historical predictors is justified and might be shown to perform better on new evaluation sets in the future.


5.3 Discussion and Limitations

The task at hand could be summarised as an effort of labelling a dataset in order to extract two unbalanced classes as well as extract a set of features upon which a binary classifier is modelled, and then trained and evaluated with the aim of yielding sufficient sensitivity as to not overlook the under-represented critical class.

The labelling of the dataset is accomplished using findings presented in a paper [25] studying the window of predictability of patient relapse before it results in hospitalisation. In addition to hospitalisation data being the only provided events present in the dataset that imply that a patient is experiencing relapse, it is also known and has been discussed that the dataset is influenced by the state of the art `alertInds` classifier,

which, serving its purpose, has surely prevented hospitalisations by causing adjustment of patients' medication doses. However, in labelling the dataset, this has not been taken into account as the original provided dataset does not contain information on whether and when a potential relapse might have been prevented. A future study on the ITAREPS programme in Bavaria, Germany should provide more patient data that have not been influenced in such a way, which should lead to a better labelling scheme and therefore an improvement of the solution of the task at hand.

Regarding feature extraction, there are two areas of interest: working with data dependent in time (time-series) and principal component analysis. In a certain way, both areas present two sides of a coin. The employed method of integrating time dependence into the feature set was somewhat successful, yet not optimal. It was successful because it produced a set of features which yield high relative predictor importance in the resulting classifier. However, it was not optimal because it does not regard sequences of more than three consequent submissions for each observation or, in other words, it is *short-sighted*. On the other hand, the employed method of applying PCA did not reflect the time-dependent nature of the dataset. In other words, perhaps instead of considering the principal components of individual EWSQ score submissions, it would be better to consider the decomposition of the development of the scores in time, modelled as an autoregressive model.



Chapter 6

Conclusion

The goal of this thesis is to improve the detection of relapses in patients suffering from schizophrenia, in order to foresee their worsening state and treat it timely and accordingly, so that long and possibly unnecessary stays in psychiatric hospitals are minimised, following a trend of decentralisation of psychiatric care from big institutions to local community centres [6]. The classifier model that is the outcome of the thesis achieves the goal to increase sensitivity of detection, predicting all practically predictable relapses. As such, it could be put into practice and integrated into the broader workings of the ITAREPS programme to help advance its long-term objectives.

6.1 Future Work

Regarding time-series analysis, future attempts should utilise more advanced methods that integrate time-series decomposition, such as singular spectrum analysis [3]. The reason why this method was not employed in this thesis is due to the nature of the chosen gradient boosting ensemble technique that, at least to the extent of what has been discussed in previous chapters, could not handle time-sequence data more effectively. In future works and given more relapse patient data is rendered available, another interesting learning technique could be researched and employed—that of classification of time-series images using convolutional neural networks [5].

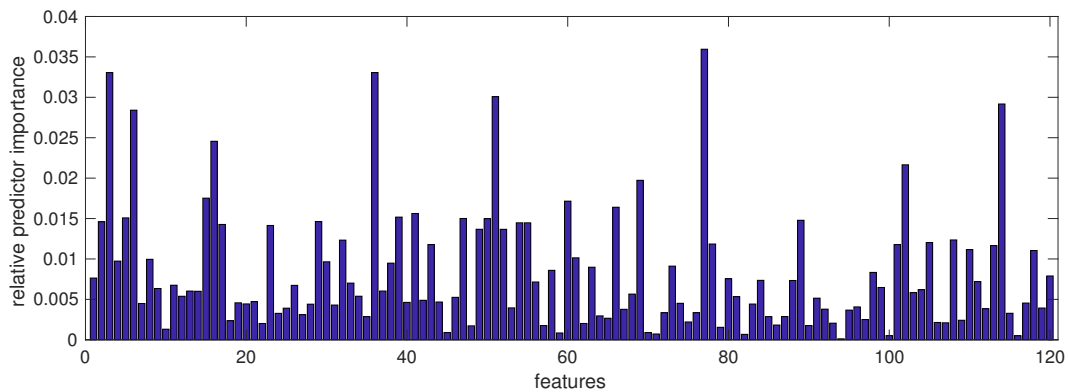


Figure 6.1. Relative predictor (feature) importance for partial sums. The ten partial sums with the highest relative feature importance are sums of the following EWSQ scores: 4,6,10; 6,7,8; 4,9,10; 3,7,10; 3,5,6; 2,7,8; 2,4,6; 3,4,10; 1,3,5; 3,6,10. The most represented combinations in all partial sums are 3, 5, 10 with five counts each.

Another feature extraction technique that could be further researched in the future and which might be of interest to the architects of the ITAREPS study, pertains to partial sums of scores and their relative importance in improving class separability. For the purpose of demonstrating this, a feature set is constructed where, for each EWSQ score submission a feature is generated from the sum of scores of all question combinations of three, of which there are $\binom{10}{3} = 120$. Then, a gradient boosting ensemble is trained on the generated feature set and the resulting predictor importance values might resemble those in Figure 6.1. It is evident that some partial sums contribute more towards separability than others, and the ten topmost ones are enumerated in the caption below Figure 6.1. Unfortunately, this feature extraction technique was formulated too late in the process of working on the task and writing the thesis and so it was not implemented.

References

- [1] Leo Breiman. *Classification and regression trees*. Routledge, 2017.
- [2] Vincent Lepetit Carlos Becker, Roberto Rigamonti and Pascal Fua. Supervised feature learning for curvilinear structure segmentation. *CVLab, Ecole Polytechnique Fédérale de Lausanne, Switzerland*, 2013.
- [3] James B Elsner and Anastasios A Tsonis. *Singular spectrum analysis: a new tool in time series analysis*. Springer Science & Business Media, 2013.
- [4] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.
- [5] Nima Hatami, Yann Gavet, and Johan Debayle. Classification of time-series images using deep convolutional neural networks. In *Tenth International Conference on Machine Vision (ICMV 2017)*, volume 10696, page 106960Y. International Society for Optics and Photonics, 2018.
- [6] Martin Hollý. Psychiatrická léčebna bohnice jde vstříc komunitní péči. *Espirit*, Feb 2009.
- [7] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.
- [8] Alisa Housková. Individualizovaná detekce relapsu schizofrenních pacientů v programu itareps. Master’s thesis, January 2017.
- [9] Gibbs Y Kanyongo. Determining the correct number of components to extract from a principal components analysis: A monte carlo study of the accuracy of the scree plot. *Journal of Modern Applied Statistical Methods*, 4(1):13, 2005.
- [10] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.
- [11] Emil Kraepelin. *Psychiatrie; ein Lehrbuch für Studierende und Ärzte, 5th edn.* JA Barth, 1896.
- [12] Emil Kraepelin. *Psychiatrie; ein Lehrbuch für Studierende und Ärzte, 6th edn.* JA Barth, 1899.
- [13] Rokach Lior et al. *Data mining with decision trees: theory and applications*, volume 81. World scientific, 2014.
- [14] Feng Pan, Tim Converse, David Ahn, Franco Salvetti, and Gianluca Donato. Feature selection for ranking using boosted trees. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 2025–2028. ACM, 2009.
- [15] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

- [16] David Martin Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2011.
- [17] Yun Qian, Yanchun Liang, Mu Li, Guoxiang Feng, and Xiaohu Shi. A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, 143:57–67, 2014.
- [18] Michael Shepherd, David Watt, Ian Falloon, and Nigel Smeeton. The natural history of schizophrenia: a five-year follow-up study of outcome and prediction in a representative sample of schizophrenics. *Psychological Medicine Monograph Supplement*, 15:1–46, 1989.
- [19] F Španiel, P Vohlídka, J Kožený, T Novák, J Hrdlička, L Motlová, J Čermák, and C Höschl. The information technology aided relapse prevention programme in schizophrenia: an extension of a mirror-design follow-up. *International journal of clinical practice*, 62(12):1943–1946, 2008.
- [20] Filip Španiel, Jan Hrdlička, Tomáš Novák, Jiří Kožený, Cyril Höschl, Pavel Mohr, and Lucie Bankovská Motlová. Effectiveness of the information technology-aided program of relapse prevention in schizophrenia (itareps): a randomized, controlled, double-blind study. *Journal of Psychiatric Practice*, 18(4):269–280, 2012.
- [21] Filip Španiel, Pavel Vohlídka, Jan Hrdlička, Jiří Kožený, Tomáš Novák, Lucie Motlová, Jan Čermák, Josef Bednařík, Daniel Novák, and Cyril Höschl. Itareps: information technology aided relapse prevention programme in schizophrenia. *Schizophrenia Research*, 98(1):312–317, 2008.
- [22] Martin Stefan, Mike Travis, Robin Murray, and Matcheri S Keshavan. *Atlas of Schizophrenia*. CRC Press, 2002.
- [23] Shirli Werner, Dolores Malaspina, and Jonathan Rabinowitz. Socioeconomic status at birth is associated with risk of schizophrenia: population-based multilevel study. *Schizophrenia Bulletin*, 33(6):1373–1378, 2007.
- [24] Tomáš Werner. Optimalizace. 2018.
- [25] Filip Španiel, Eduard Bakštein, Jiří Anýž, Jaroslav Hlinka, Tomáš Sieger, Jan Hrdlička, Natálie Görnerová, and Cyril Höschl. Relapse in schizophrenia: Definitely not a bolt from the blue. *Neuroscience letters*, 2016.



Appendix A

Abbreviations

- EWSQ ■ Early Warning Sign Questionnaire
- ITAREPS ■ Information Technology Aided Relapse Prevention Programme in Schizophrenia
- NaN ■ Not a Number
- PCA ■ Principal Component Analysis
- SMS ■ Short Message Service