Mälardalen University
School of Innovation Design and Engineering
Västerås, Sweden

Thesis for the Degree of Master of Science (60 credits) in Computer Science with Specialization in Software Engineering - 15.0 credits

TEST ORACLE AUTOMATION WITH MACHINE LEARNING: A FEASIBILITY STUDY

Nermin Imamovic
nic17001@student.mdh.se

Examiner: Mobyen Uddin Ahmed
Mälardalen University, Västerås, Sweden

Supervisor: Adnan Causevic
Mälardalen University, Västerås, Sweden

Company Supervisor: Ola Sellin,
Bombardier Transportation, Östra Ringvgen 2,
722 14 Västerås

June 19, 2018

Abstract

Machine Learning is increasingly being used in different industries. It also found a place in software testing, what we show in this thesis. First, it is needed to show motivation for solving different types of the test oracle problem, what is in the most cases related to cost reduction. The thesis focuses on the using different methods to examine the feasibility of using machine learning approach for solving the test oracle problem and use of the most appropriate for the implementation. Through background, we present main terminology and basics related understanding the work proposed in answering research questions such as test oracle, machine learning, and multivariate time series classification.

For the solving this type of the test oracle problem, we chose certain use case and implemented solution following the feature-based multivariate time-series classification. The case study is based on the analyzing real data from industry mixed together with simulated data in consultation with industry experts. Results are shown as the answers of the research questions as the best methods needed for solving this type of the test oracle and accuracy of the machine learning algorithm together with confusion matrix, sensitivity, and specificity. We presented different results when we use different features for making machine learning model. For the end, this thesis proposes possible future work related to this or similar topics.

Keywords*: Test oracle automation, the oracle problem, machine learning, classification, feature engineering, signal classification, time-series analysis, time-series classification, multivariate time-series classification*

Acknowledgements

First I want to show my gratitude to my supervisor Adnan Causevic for unselfish and continuous support and the opportunity to get in the new field of computer science. Special thanks are reserved of Ola Sellin, Magnus Bergman and Kent Ohrn at Bombardier Transportation, Västerås for the opportunity of working thesis project on this wonderful place and resources for the organization of my case study. I would like to thank my examiner Mobyen Uddin Ahmed for showing me approach to answer my research questions. I would also to show my gratitude to professors from my home and host universities for the opportunity to study in Sweden. For me, the most important gratitudes are reserved for my parents Nermina and Ismet, for unconditional love and support in achieving my dreams, my sister Nasiha, brother Adnan, and friends for motivation and support. The foremost, I need to show my gratitude to God, without Who everything will be impossible, with my favorite quote:

"*O my Lord, increase me in knowledge!*" (The Qur'an Taha 20:114)

Table of Contents

List of Figures

1 Introduction

Software verification and validation represent one of the most important phases of software development process. The software verification process is used to determine whether development products of given activities satisfy the expected requirements of that activities, while the software validation represents process used to determine whether the product satisfies user requirements and needs [1]. One of the most important software verification and validation techniques is called software testing and it is used to identify correctness and quality of software. Traditionally, software testing has been done by a human tester who has chosen test cases and has analyzed results. As the number and size of programs have increased, there have produced a lot of work for human testers, and there are chance and need for new automated methods.

Usually, automated methods are used for executing test cases on predefined actions and compare them with expected results, what represents the oracle problem. These methods often use predefined test scripts for the execution written by the human tester, but there are many situations where it is impossible because a human expert should give the final verdict. The best example of this situation is a regression testing. A regression testing is a phase of software testing where re-testing of the previous system has to be conducted in order to ensure that previously working functionalities have not been corrupted without any of additional features and that this program works regularly with new changes [53]. It is a normal part of the software development process and it is usually done by code testing or quality assurance specialists. During other phases of software verification process such as unit and integration testing, regression testing can help to catch defects early and reduce the cost to resolve them. It is also known that regression testing can affect two-thirds of software verification costs [53, 27].

The human oracle is one of the most popular types of un-automated oracles. Specialists use their knowledge and experience from different fields to give a final verdict, what cannot be completely reliable with thousands of test cases [43]. The classical way of automation can be a long-lasting and expensive process. Example of this problem is presented in the signal classification. The expert should have experience and knowledge to classify signal for certain problem. The most commonly used techniques for signal classification are based on machine learning.

The human expert spent a lot of time resources for regression testing due to this we want to investigate the possibility of using artificial intelligence for machines for solving this problem. Branch of artificial intelligence what has the ability to learn from the previous experience is called machine learning. Machine learning is used in different industries for different problems, what represents a possibility of using machine learning in software testing for solving test oracle problem. The main challenge lays in investigating whether it is possible to automate the process of examining the behavior of different types of verdict processes, such as analog and digital signals. The goal of this master thesis is to examine whether it is possible to create system what can approximately make decisions about examining different types of verdicts.

Machine learning represents one of today's most rapidly growing technical fields, where we have the intersection of computer science and statistics at the core of artificial intelligence and data science [20]. Due to this reason, we want to investigate the feasibility of using machine learning in solving test oracle problem based on different types of signals.

1.1 Problem Formulation

The main purpose of this thesis is expressed in a specific phase of software testing, namely regression testing, where re-testing of a system has to be conducted in order to ensure that previously working functionalities have not been corrupted with any of additional features. Research questions objective is related to usage of machine learning approach for test oracle automation.

1.2 Research Objective

After we formulated problem we can specify research goals. The main goal of this research is to show the most common way of solving test oracle problem with machine learning approach where

[1] Ieee approved draft standard for system, software and hardware verification and validation.IEEE P1012/D17, August 2015, pages 1262, Jan 2015.

different signals are part of the verdict process. Other research goals are related to presenting the most important phases of the test oracle automation such as data pre-processing, feature extraction, feature selection, and classification.

1.3 Expected outcome

The expected outcome of this research is to implement the certain use case for the solving test oracle problem. The use case is implemented using Support Vector Machine classification. For generating the machine learning model we use Classification Learner application integrated into the Matlab.

1.4 Report outline

The report is organized as follow:

- Section 2 presents the background, main challenges for the solving the test oracle problem, and describes different methods for time-series and multivariate time-series classification.
- Section 3 propose the methodologies used for the thesis. We did a literature review on the topics of the test oracle problem and machine learning to explore relations between them. Also, we defined the design of the case study used in this thesis.
- Section 4 is reserved for the implementation of the case explained in 4.2 with methods proposed in 2.
- Section 5 presents the final evaluation of collected data in the case study with a short discussion.
- Section 6 shows limitation what we have during the working of this thesis.
- Section 7 is reserved for the conclusion and future work.

2 Background

In this section, we explain main terminology and concepts of this thesis. We will get acquainted with terms in regression testing, test oracle problem, machine learning, feature engineering, time-series analysis, etc.

2.1 Software testing

Software testing is the process of analyzing a software item to detect the differences between existing and required conditions and to evaluate the features of the software system [2]. According to ISTQB(an internationally recognized software testing certification), System Under test represent test object (component or system), what is validated for correct operation. ISTQB defines the test case as a document of a set of conditions or actions performed to verify the expected functionality of the software application [25]. A collection of the test cases used for test execution process of the software application is called a test suite, while the executing of the test suites is called the test run. That can be unit, integration, system and acceptance testing. Also, there are a lot of different types of the software testing, such as alpha testing, beta testing, specification-based testing, implementation based testing, regression testing, etc.

2.2 Regression testing

Regression testing is the process of ensuring that system works normally after modifications. The formal definition of regression testing from IEEE Standard Glossary of Software Engineering Terminology is:

”*Selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements.*”

It is a normal part of the software development process and it is usually done by code testing or quality assurance specialists, what requires a lot of time. During other phases of software verification process such as unit and integration testing, regression testing can help to catch defects early and reduce the cost to resolve them. The main motivation for executing regression testing is improving quality of system reducing costs of system maintainability. More than 80% of software testing cost is reserved for regression testing, while software testing wastes more than 50% of software maintainability costs [12]. As it is defined in IEEE Standard for Software Test Documentation, several iterations of the system will be executed in order to test changes in the program caused during the system test period, what indicate that regression testing will be performed for each new version of the system to detect unexpected impact resulting from program modifications, where the results are generated by comparing results from new and previous versions of tests [3]. Software testing is often divided into two groups: manual and automated testing. Automated testing is often applied in situations where we have repetitive tasks, where test cases are executed after every changes [24]. One of these situations is the regression testing, where is needed to be ensured that system works regularly after new changes. The process of the automation of regression testing was executed in the way of writing testing scripts and verifying test results [24]. There are situations where is not possible to automate test process by the execution of testing scripts,

2.3 Test Oracle Problem

Under the term software testing, it is often considered examining the behavior of a system for certain inputs in order to discover potential faults. For successful automation of execution of the test processes, we are using test oracles to asses if the test case has passed or not. The main aim of the test oracle is to check whether the result test execution is as expected [13]. With regard to different inputs of the system, the challenge of the corresponding desired, correct behavior from potentially incorrect behavior is called the ”test oracle problem” [5], while verdict represents the result of test case execution where expected and actual outputs are compared [31]. The comparison of those verdicts task where human testers should rerun old tests with a new version of the program

[2] Ieee standard glossary of software engineering terminology.IEEE Std 610.12-1990, pages 184,Dec 1990.

[3] Ieee standard for software test documentation.IEEE Std 829-1998, pages 164, Dec 1998.

that increases test execution what requires extra time and resources. One of the biggest challenges of automating test oracles is to reduce the involvement of human testers and in this way reduce costs. A special challenge in this master thesis lies in research on how to automate the process of the corresponding new and different test types. Using test oracles there are a lot of simple and reliable resources that guide testers to undertake the testing process and detect faults [58].

Oracles can be classified into specified test oracles, derived test oracles, implicit test oracles and no automatable test oracle [5]. Verdicts in specified test oracles are based on all behavioral aspects of the system with respect to a given formal specification. Specification languages define a mathematical model of a system's behavior, using formal semantics to define the meaning of each language construct in terms of model [5]. In these cases where the verdicts of the System Under Test (SUT) are based on the specification, we can use formal specification language to define the oracle problem [13]. We define "test oracle" using formal specification language on next way.

Definition (Test Oracle)"*A test oracle* $D : T_A \rightarrow B$ *is a partial function from a test activity sequence to true or false* [5]."

Derived test oracles are used when specified systems are unavailable [13]. They are using different artifacts such as documentation or previous versions of systems [13, 5]. An implicit oracle is used for identification and detection of the system defects. Last category, a no automatable test oracle is reserved for handling the lack of an automated test oracle, where a human tester must verify whether the behavior of SUT is correct [5].

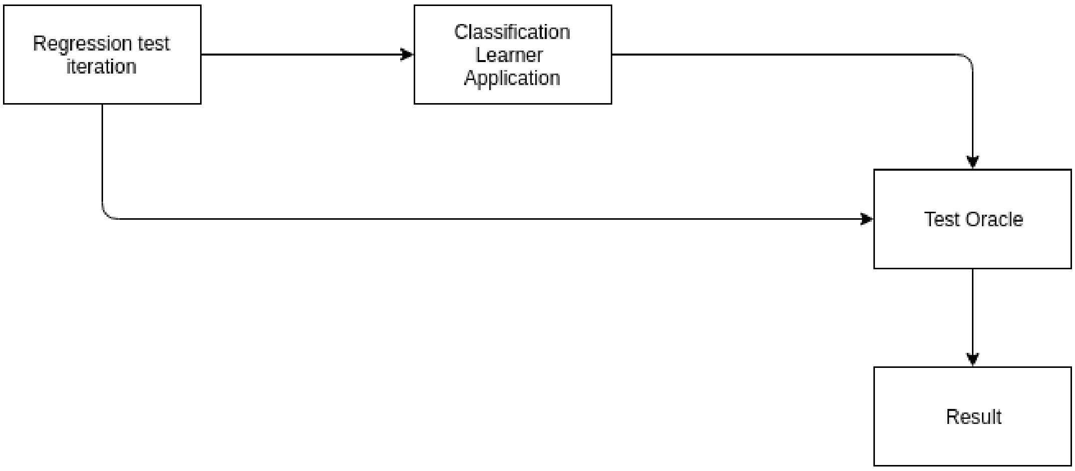


Figure 1: Schema of the test oracle system for automated testing

In most cases, the test oracle observes system behavior and return verdicts, what can be pass or fail. The pass means that observed behaviors match the specification of the SUT, while the fail means that SUT is not consistent with specification[13].

2.3.1 The Human Oracle Problem

In the past, the oracle problem was reserved only for human experts, where they had the responsibility of examining the behavior of the system[13]. The human experts are using knowledge and previous experience gained in solving similar problems. The main challenge in the human oracle problem is labeled as the Human Oracle Cost. It represents a way of the effort needed that human expert need to create test cases or in evaluating them[5, 29]. Reducing the human oracle cost is classified in next ways as quantitative and qualitative reduction[5]. Quantitative Cost Reduction is displayed in the reducing the amount of work that the tester has to do for the same amount test coverage, while Qualitative Cost Reduction represents reducing the work needed to understand and evaluate test case [5]. Quantitative Human Oracle Cost is the most impressive in the reduction of a test case and test suite. All characteristics of the Qualitative Human Oracle problem lays in the incorporation of human knowledge for better understanding of test cases. Where is needed to simulate knowledge and experiences from the human expert we cannot use automated oracle based on script execution. In this situations, it is necessary to use other techniques based on artificial intelligence, machine learning, and fuzzy experts system.

2.4 Machine Learning

Machine learning has been one of the most popular research areas for last few decades and it is getting more and more used in different parts of software engineering. It was born from the pattern recognition and computational learning theory [4]. It is the most effective data analysis method used in order to predict something by devising some models and algorithms [4]. Programs or computers learn on the way by studying different training data sets in order to predict next behavior. Now machine learning is used in different branches of computer science where is needed to make predictions or decisions based on sample inputs [35]. With the prediction of next behavior, machine learning is employed for automation a lot of different processes.

As machine learning has become used in different parts of software engineering, it is possible to research its usage in software testing. Using machine learning system it will be possible to predict the result of test case execution. It is important to understand differences between automation and machine learning as artificial intelligence. As it is explained in the previous section, automation test compares actual outputs with predicted executed on predefined test scripts. The problem arises in the situations when it is needed to test actual and desired outputs without a possibility to create test scripts. These situations are when is important to use knowledge and experience from the human expert. The solution to this problem is in the simulating human experience and thinking what is the main purpose of artificial intelligence. There are different machine learning methods used to solve a different kind of problems. The most used machine learning techniques are supervised learning, where for training are used labeled examples with known inputs and desired outputs, and unsupervised learning, where is the task to explore data and find some structure within [35]. In this research, the most are taken for supervised learning algorithms, where is needed to classify the different type of data. The goal of using machine learning is to create a model that can be implemented in the system. A model represents an output generated by using certain machine learning algorithm on the training set. Generally, this is machine learning workflow.

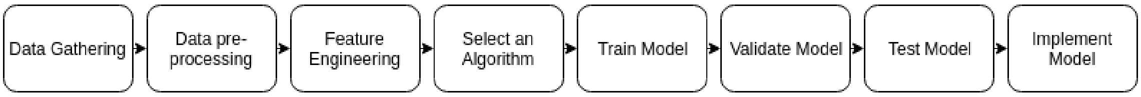


Figure 2: Machine Learning workflow

2.5 Supervised Learning

Supervised learning is the most popular problem in machine learning. It is known for the problem of classification of pictures such as dog or cat or classification of handwritten digits. It is called supervised learning because it is needed to label training data by the expert that they can be used for classification or prediction of unlabeled data. Supervised learning can be defined as the machine learning technique based on mapping a set of input variables X and output variables Y, using this mapping to predict outputs for unseen data [10]. After a set of input is labeled with correct outputs, the algorithm learns by comparing its actual outputs with correct outputs to find errors.

Two main problems of supervised learning are called regression and classification. Regression problem is reserved for predicting a continuous value of output for different input. A classification problem is shown in situations where the output variable is some category or class. The most known classification is binary classification, where we have two class of output such as "Yes" or "No", "1" or "0", "Good" or "Bad". If we have more than two classes for output this problem

[4] A dictionary of computer science. Oxford reference online premium. Oxford University Press, Oxford], seventh edition, 2016.

is called multiclass classification problem. How the main occupation of this research is to classify signals, we explain the most used classification algorithms.

General steps in supervised learning:

- Gathering data
- Preparing the data (Pre-processing)
- Feature Engineering
- Selecting Algorithm
- Training of the model
- Validation of the model
- Testing of the model
- Prediction

An existing dataset is always divided into three parts training, validation, and test sets. The training set is used for training of a model after we selected certain algorithm. The validation set is used to minimize the overfitting of the model. The overfitting represents model's learning the noise and detail in the training data that can cause mistakes in the model performances. The test set is used to get final performances of the model. Final performances are shown through accuracy, sensitivity, specificity and confusion matrix. A confusion matrix is a table on a set of data with known true values to show the final performance of the model. In binary classification, values in table are marked as True Positive (data what has positive value and for what is predicted to be positive), True Negative(negative data for what is predicted to be negative), False Positive (negative data for what is predicted to be positive) and False Negative(negative data for what is predicted to be positive). Accuracy presents the correctness of the algorithm (sum of the True Negative and True Positive over the set of data). Sensitivity is the percentage of true values in the date for what is predicted to be true(percentage of the True Positive over the sum of True Positive and False Negative). Specificity represents a percentage of the false values in the values what are predicted to be false (True Negative over the sum of True Negative and False Positive).

The most common used supervised algorithms for classification are Logistic Regression, K-Nearest-Neighbour(KNN), Support Vector Machine(SVM), Artificial Neural Network(ANN). In this thesis, we explain two algorithms proposed for time-series classification k-Nearest Neighbour (kNN) and Support Vector Machine (SVM).

2.5.1 The k-Nearest Neighbor (kNN)

k-Nearest Neighbor is one of the simplest classification algorithms. kNN algorithm uses a database with previous data points to predict the class of new data point [48]. Classification is done on the way by finding k closest data points to a new data point.

Steps in kNN (formulated in [48]):

1. Specifying positive integer k with new data point
2. Selecting k entries in databases closest to the new data point
3. Finding the most common classification of these entries
4. Get class for new data point

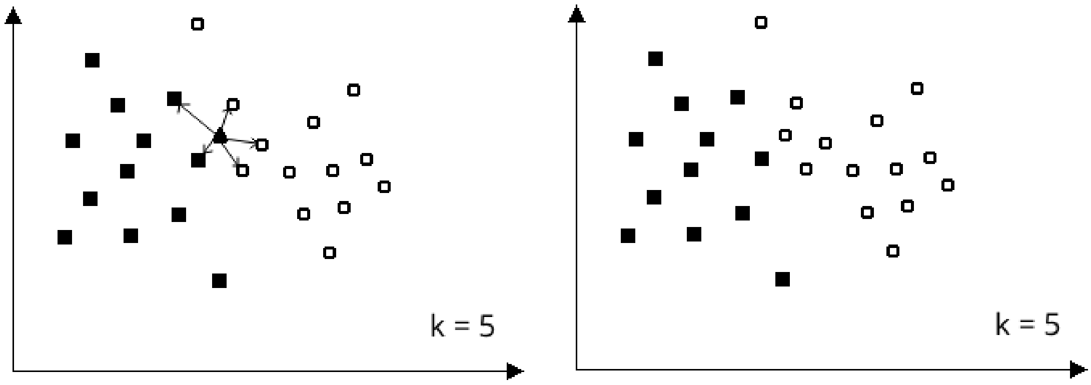


Figure 3: kNN classification algorithm (k = 5)

2.5.2 Support Vector Machine (SVM)

A Support Vector Machine is one of the most commonly used classification algorithms. This algorithm was introduced by Boser, Guyon, and Vapnik in 1992. Formally, SVM constructs an optimal N-dimensional hyperplane that separate data in N categories. There are two different types of SVM, linear and non-linear. For linear SVM, in N-Dimensional Euclidean space, hyperplane represent a flat that consists of the N-1-Dimensional subset that separates the space in two parts. In SVM, the main task is to find appropriate margin, what means to maximize the distance between hyperplane an and nearest data points of different classes.

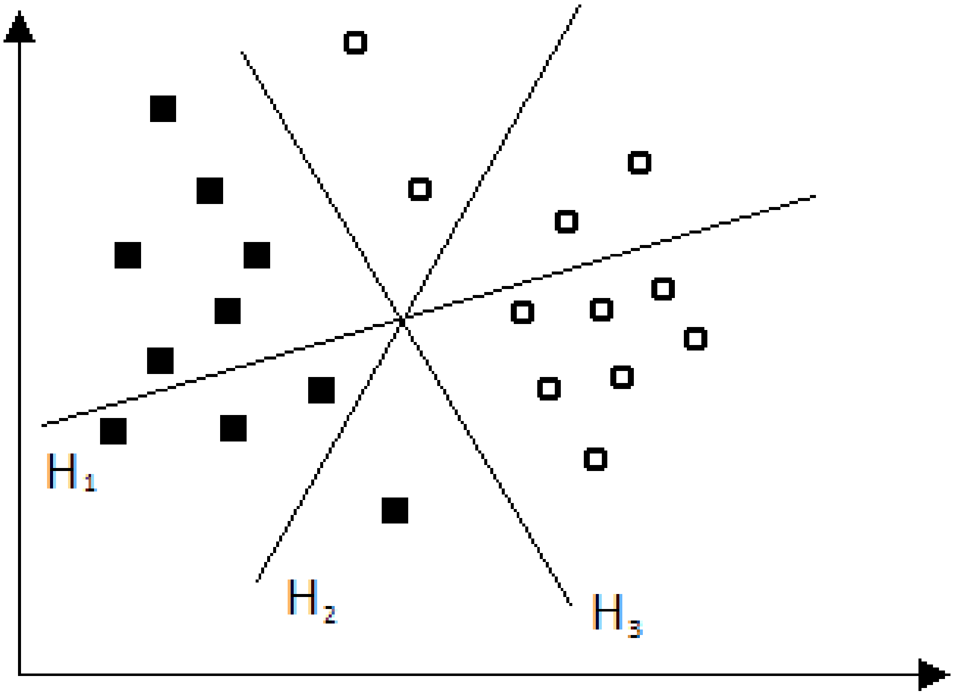


Figure 4: Finding the hyperplane for SVM. (H_3 is the best hyperplane)

There are two steps in SVM classification (proposed in [59]):

1. Transform inputs into a high-dimensional feature vectors
2. Finding the hyperplane of maximal margin in feature space

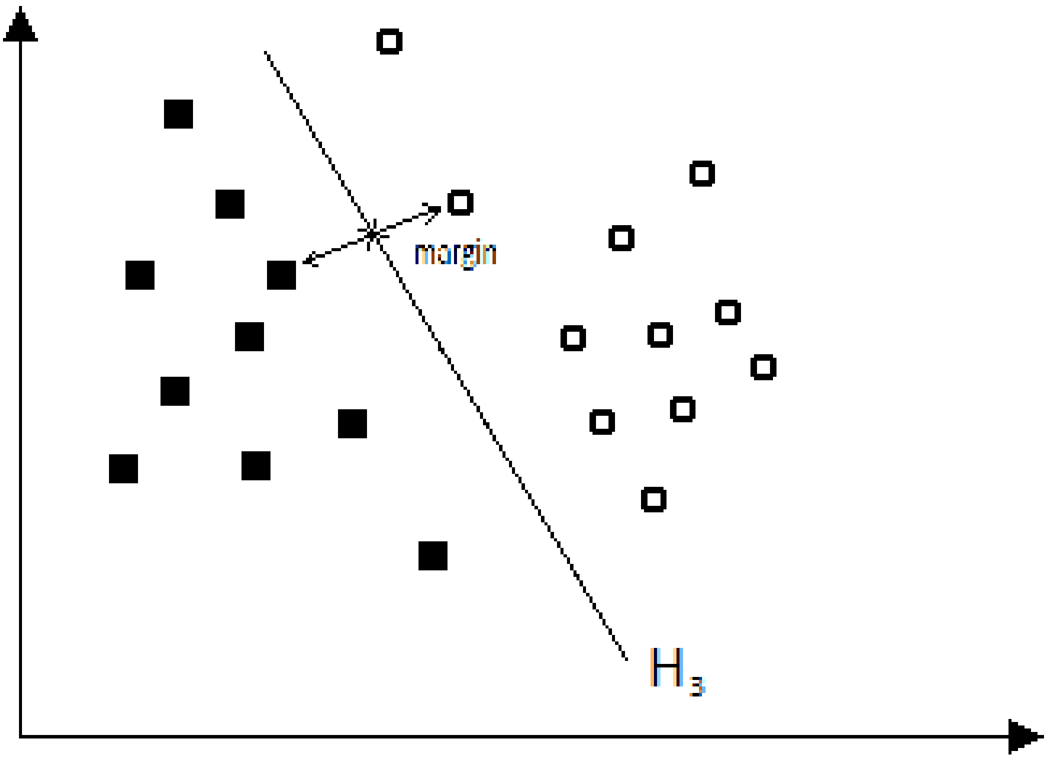


Figure 5: Maximal margin of the hyperplane

2.6 Feature Engineering

Machine learning algorithms are not applied to the raw data, but rather on extracted features [50]. Feature engineering is the most important process in machine learning. It represents using domain knowledge of the analyzed data to construct suitable features that lead to improved predictive performances of machine learning [33]. A feature can be defined as an attribute that contains a characteristic property useful for gathering useful information for machine learning. In the other words, features are an abstraction of the original output data used as input of machine learning system [50].

The dimensionality of data involved in machine learning has increased explosively, what causes a large number of features for data and results in overfitting of learning model and degeneration in their performances [49]. In this situation, it is necessary to reduce the dimensionality of data. Two most useful dimensionality reduction techniques are feature extraction and feature selection. Feature extraction maps the original feature set to a new feature set with lower dimensions. In our case where data are presented as time-series, feature extraction is a process where a signal is characterized into numerical representation [22]. Formally, feature selection is a process of selecting a subset of predominant features from original feature set and remove the feature that is irrelevant without any transformation [49, 22]. The most important task of feature extraction is to keep characteristics of data, needed for classification [8]

Features from time-series can be categorized into statistical and structural features [34, 50]. Statistical represent quantitative values of data, while structural represent organization of subpatterns(e.g shapes) and their interrelationships in the data [34]. Structural features are based on human perception and cognition, while statistical features are based on statistical decision theory [34]. When we collect iterations of regression testing, it is needed to extract features and generate feature vectors from them. After that, it needs to feed this vectors to an SVM what is used for classification of the new regression test iteration.

2.6.1 Digital Signal Processing

Generally, signal represent any time-varying physical quantity [46]. Digital Signal Processing is applied in different industry fields such as data compression, speech recognition, power distribution and, as the term suggests, it processes the signal with proper algorithms, mathematics, and specialized techniques [15]. In most cases, signals from the power system analysis are coming in analog form, and they should be converted into digital form. Signals can be presented in different domains such as time and frequency domain. In time domain it is shown how the signal changes with its amplitudes over time, while in frequency domain signal is described by the sinusoidal signal of which it is composed [23]. Pre-processing of the signal is the most important thing in DSP, to remove irrelevant and redundant values of the signal and to get the most important information and features from it. There are different techniques for changing the domain of different signals such as DFT (Discrete Fourier Transformation) and DWT (Discrete Wavelet Transformation).

2.7 Time-series analysis

The main aim of time-series analysis is to create mathematical models that provide possible descriptions for sample data [44]. It is used to provide a statistical setting for the character of different data types that fluctuates in random fashion over time and we can say that time-series represents an ordered sequence of data points recorded at specific time interval [44]. Applications of the time-series data can be expressed in different fields of science such as industry, where signals are taken from sensors, biological data, speech, sounds, economics, etc.

Time-series is variable what represents a collection of data made sequentially in time. It can be univariate or multivariate, what is classified in the way of observations [50], if observed variable consists of more than one component it is multivariate time-series [8]. Univariate time-series represents an observation of one variable such voltage signal in PowerLineLimitation during some period of time, while multivariate time-series represents an observation of all signals in Powerline limitation during some period of time. Conventionally time-series data are displayed graphically by plotting the values on the vertical axis (ordinate), with a time scale on the horizontal axis (abscissa). Of course for the human it is more easily and clearly visible and understandable.

Univariate time-series X_T represent a finite sequence of N data points ordered by time, what can be written as :

$$X_t = \begin{pmatrix} x_{t_1}, & x_{t_2}, & \cdots & x_{t_n} \end{pmatrix} \text{[50].}$$

Multivariate time-series represents a generalization of univariate time-series instances, where they are composed of different streams measured at the same time. Multivariate time-series data are often used in medical and industrial applications where they are generated from several sensors showing different component of this data [8]. A multivariate time-series Y_t consists of at least two univariate time-series, which are denoted by Y_u with 1 ≤ u ≤ M, where the number of data points N is same for all univariate time-series in multivariate one [50]. It can be written as:

$$Y_t = \begin{pmatrix} y_{1,t_1}, & y_{1,t_2}, & \cdots & y_{1,t_n} \\ y_{2,t_1}, & y_{2,t_2}, & \cdots & y_{2,t_n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m,t_1}, & y_{m,t_2}, & \cdots & y_{m,t_n} \end{pmatrix}.$$

2.7.1 Time-series as the test oracle problem

It is possible to observe problems of signal classification such as ECG and EEG signals as the oracle problem. In the past, this process was always done by a human expert, and this process is automated.

Drawbacks of manual signal classification(time-series classification) are described in [55]:

1. Number of human experts with certain experience is limited.
2. Inspection of a different signal is time-consuming for the human expert and it requires a high level of concentration, what increases chances for mistake after long inspection.
3. Inter-reader variability in the manual inspection and monitoring by experts is necessary.

2.7.2 Time-series classification

Time-series classification (TSC) is a specific challenge for classification algorithms: how to measure the similarity between series and examine belonging to a certain class. In this type of classification, a class label is applied to an unlabelled set of ordered data. The data doesn't need to be ordered temporally; any logical ordering is sufficient. Every time-series classification process is proceeded by data pre-processing, what means removing redundant and irrelevant information from data [8]. In traditional classification problems, the order of the attributes is unimportant, and interaction between variables is considered to be independent of their relative positions. The most used types of time-series classification are instance-based and feature-based [14]. Instance-based classification methods predict labels of test time-series based on their similarity to the training instance [14].

The classification is done on the way by measuring the distances between pairs of time-series represented as an ordered and then classified using certain classifier. Instance-based classification is also known as distance or similarity-based classification because it measures similarity using distance functions [45]. The most appropriate classification algorithm used in instance-based time-series classification is KNN (K-Nearest-Neighbor). It uses a distance function *d=(t,q)*, between two time-series, labeled training instance t and query q, to find the k most similar training instances t_1, t_2,...t_k to predict the label for a query instance q [45]. Drawback of the Euclidean distance is that it requires same length of time-series.

Dynamic Time Warping is originally developed for speech recognition and it represent extension of Euclidean distance used to compare time-series what don't have same length. DTW provides warping path, in the other words the smallest distance obtained by allowing a nonlinear matching of the observations of two time-series sequences. Warping path $p := (p_1, p_2, ..., p_N)$ defines an alignment between two sequences X = $(x_1, x_2, ..., x_N)$ and Y = $(y_1, y_2, ..., y_M)$ by assigning the element x_nt of X to the element y_mt of Y.

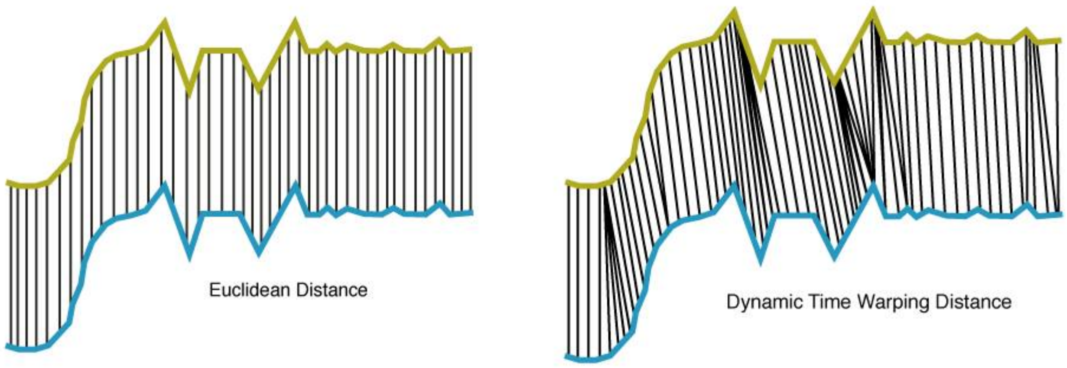


Figure 6: Differences between Euclidean distance and Dynamic Type Warping

A drawback of instance-based classification it is only applicable on short time-series.

Feature-based time-series classification are typically applied to logger time-series data. First, it is necessary to extract and select features for certain time-series data to feature vectors. This type of time-series classification uses time-series vectors as input for classification. Then it is possible to apply different machine learning algorithms for classification. It is important to have a different approach for feature extraction of different types of time-series data. In these situations, we use statistical and structural features for creating feature vectors for certain time-series data. Next phase is to select certain features to reduce dimensions of the feature vectors.

This is a procedure for classification of univariate time-series data. From time-series X_t of N data points ordered by time:

$$X_t = \begin{pmatrix} x_{t_1}, & x_{t_2}, & \cdots & x_{t_n} \end{pmatrix}$$

we extract and select certain features, then generate feature vector F with K features.

$$\vec{F} = \begin{pmatrix} \vec{f_1}, & \vec{f_2}, & \cdots & \vec{f_3} \end{pmatrix}$$

Also, it is needed to create feature vectors from other time-series what will be used in classification and generate feature matrix, what will be used for classification. From feature matrix, we generate training set used for training of model with a certain algorithm and testing set used for testing it. When we find the most appropriate model, we can make a classification for new data.

2.7.3 Multivaraite time-series classification

As it is mentioned multivariate time-series data consists of more than one univariate time-series data. Traditional time-series classification algorithms. cannot be directly applied to the classification of multivariate time-series data [8]. Feature-based representations of multivariate systems can include both features of individual time-series, and features of inter-relationships between (e.g., pairs of) time-series.

For this type of multivariate time-series classification, it is needed to extract features from every-time series in multivariate time-series data. This features will be placed in the same feature vector, from which the training table is generated.

This is a procedure for classification of multivariate time-series data. From time-series Y_t of M univariate time-series with N data points ordered by time:

$$Y_t = \begin{pmatrix} y_{1,t_1}, & y_{1,t_2}, & \cdots & y_{1,t_n} \\ y_{2,t_1}, & y_{2,t_2}, & \cdots & y_{2,t_n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m,t_1}, & y_{m,t_2}, & \cdots & y_{m,t_n} \end{pmatrix}$$

features are extracted and selected, then we generate feature vector F with K features.

$$\vec{F} = \begin{pmatrix} \vec{f_1}, & \vec{f_2}, & \cdots & \vec{f_3} \end{pmatrix}$$

Also, we create feature vectors from other time-series what will be used in classification and generate feature matrix, what will be used for classification. From feature matrix, we generate training set used for training of model with certain algorithm and testing set used for testing it. When we find the most appropriate model, we can make a classification for new data.

3 Research Methods

This research has different phases what summarize different research methods. The first phase is the systematic literature review about test oracle problem and machine learning followed by industrial case study.

3.1 Literature review

In early stages, it is important to make a literature review to identify related work and main challenges in test oracle automation with machine learning approach. Through this literature review, we will get acquainted with existing approaches to using machine learning techniques for solving test oracle problem and time series classification. The systematic literature review helps us to find different techniques in time series classification.

We used next databases for the systematic literature review:

- IEEE Xplore
- ACM Digital Library
- Springer Link
- Google Scholar

We used next terminology for the search queries in different combinations:

- Automated testing
- Regression test
- Test Oracle
- Machine Learning
- Signals
- Time Series
- Multivariate Time Series
- Feature Extraction
- Feature Selection
- Classification

As results, we have different research papers, scientific journals, books, etc. Through literature review, we will answer the first research question and find the base for the RQ2.

3.1.1 Related work

As the main topic of the thesis is the test oracle problem and signals are parts of a verdict process, we will explain two topics in related work. The first topic is intended for software testing and machine learning where we explain utilization of machine learning in software testing generally. Another topic is reserved for the signal classification and the time series classification generally. It is possible to imagine train as a human. Propulsion Control (PPC) system is represented as a heart of the train, while Train Control Management System (TCMS) represents a brain of the train. Regression tests iterations consist of several different signals to represent different output on simulation of the system. We can imagine it as usage of medical applications to measure dynamics of brain or heart regions through time, ElectroEncaphalogram (EEG) or ElectroCardiogram (ECG) applications, where different signals represent data generated from different sensors in some time period [8].

A huge amount of software testing applications and research papers can be found in different types of literature, and it represents one popular research area. First time when the term ”test oracle” is appeared is in the William Howden’s seminal work in 1978 [5, 16]. In the paper ”The Oracle Problem in Software Testing: A Survey”, authors list and explain main challenges in the oracle problem and analysis of software testing research and practice [5]. Test oracle by itself shows new challenges of reducing costs and increasing benefits [5], what represent main motivation of writing this thesis. That is also stated in the research ”Reducing Qualitative Human Oracle Costs associated with Automatically Generated Test Data”, where authors made simple case study where they explained how automatic test data generation reduce qualitative human oracle costs [29], what we also want to do in our research. How main aim of this research lies in regression testing, it is important to label the most important work about regression Testing. In their work, Engström and Runeson gave a quality analysis of the testing phase expressed in this research, regression testing in academia and different industries [12].In 2003, Lin wrote paper ”Regression Testing in Research And Practice”, where he specified challenges of regression testing in industry and research work [6]. Machine learning has become popular research area since the middle of 20 the century. Of course, it has been changed during the time. It was born from pattern recognition and the theory that computers can learn without being programmed to perform specific tasks [4]. At the beginning researchers interested in artificial intelligence wanted to see that computers can learn from the data. Angra and Ahuje in their research [4] made a review of using machine learning in different applications. In 2017, Ongsulee wrote paper [35], where he made an analysis about artificial intelligence, machine learning, and deep learning. Now, machine learning has a wide spectrum of utilization in software testing in order to solve testing problems such as the completeness of test suites, localization of faults causing failures, risk-driven testing, and the automation of test oracle [7].

Wang, Yao and Wo in their paper ”Intelligent Test Oracle Construction for Reactive Systems without Explicit Specifications” presented InTOL, a library for a convenient and flexible collection of test traces, where they used machine learning techniques for automatically construction of test oracle [54], what represent good base for the creation of the system prototype implemented in section 4.2. In our case, results of the regression testing are signals plotted on a graph where the human expert should determine is it correct. This process takes a lot of time, so it is necessary to use artificial intelligence as the human minds, so that machine mimics ”cognitive” functions such as ”learning” and ”problem-solving” [35], what will reduce qualitative human oracle explained in [29]. A branch of artificial intelligence, that explores study and constructions of algorithms that can learn from and make predictions on the data is called machine learning [35]. As an example of artificial intelligence, machine learning techniques have application for solving oracle problem is research paper [1], what shows that the best examples of artificial intelligence to solve test oracle problem are artificial neural networks and info-fuzzy systems. In scientific researches [1] and [58] are made comparison on how to use solve oracle problem using artificial neural networks (ANNs). In 1995, Cheatham, Yoo, Wahl in their research ”Software testing: A Machine Learning Experiment” made an analysis about using machine learning techniques to identify the most important attributes in prediction software testing costs and execution time of testing process in certain company [9]. This research [9] shows good basis of using machine learning for solve our problem. Mathur, Miles, Du in their research ”Adaptive Automation: Leveraging Machine Learning to Support Uninterrupted Automated Testing of Software Applications” made a framework of using leverage self-adaptive technologies and machine learning methods for the automation of the test process [28].

M. Polo, P. Reales, M. Piattini explain in their paper that testing activities take up between 30 and 60 percent of all software life-cycle costs, depending on critically and complexity of the product and they suggest that it is important to control and reduce test costs through test automation [37]. Almagharibe and Roper in their research ”Automatically Classifying Test Results by Semi-Supervised Learning” proposed new approach to solving test oracle problem with the use of the semi-supervised machine learning algorithms [3]. Also, in [3] authors specified other machine learning techniques used in different types and phases of software testing, such as supervised algorithms (classification algorithms) in regression testing, what we used in implementation of the system prototype. In [51] authors proposed new automated oracle system to reduce costs of the testing process. This automated oracle system is based on the artificial neural networks, based on a black-box approach where presented only inputs and outputs[51]. The artificial neural is trained by backpropagation algorithm with test cases from the original system, and it can be used as an oracle for evaluating the correctness of the system. We will also use black-box approach where we need to get outputs for certain inputs. In their research ”Classification of Software Behaviors for Failure Detection: A Discriminative Pattern Mining Approach” authors proposed a method for classification method to detect potential failures [26]. In [43] authors provide a method of using the artificial neural networks as a multi-networks automated test oracle. In [13] authors made a case study on automatic analysis of the non-functional test results.

Time-series analysis now represents one of the biggest challenges in computer science. Generally, time-series analysis is presented in many different types of industries and sciences. Time-series data occurs in different context such as signal, biological or financial data, speech etc. At the beginning the main research topics in time series analysis was finding potential patterns in them. In their book ”Time Series Analysis and Its Applications: With R Examples”, Shumway and Soffer explained the usage of time series with easy examples in R programming language [44]. Volna, Kontriba and Janosek in their book ”Pattern Recognition and Classification in Time Series Data” gives the basics about time series analysis, pattern recognition and classification of time series data [52]. In the book ”Data Classification: Algorithms and Applications” are explained classification of different data types, where Kostakos and Gunopolos wrote a section about time series analysis and classification [2]. The main role of time series classification lays in feature extraction and selection. There are a lot of research papers based on feature extraction and selection for speech recognition or music classification, such as Kalaptapu, Goli, Arthum, Malapi wrote paper about feature selection and classification of Indian music [22]They analyzed different feature extraction techniques, feature selection algorithms, and different classifiers, which showed a good way for feature selection in our research. Ozlewski wrothe his Ph.D. thesis on the topic ”Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data” [34], where are proposed types of features what we used for our research. Theissler wrote his Ph.D. thesis on the topic ”Detecting anomalies in multivariate time series from automotive systems” where he paid attention to the multivariate time series classification and proposed new methods [50]. From [50] we used technique how Theissler used machine learning approach for detecting anomalies in multivariate time-series, for classification of multivariate-time series, using same steps.

Research and industry about digital signal processing have started since 60ś and 70ś when digital computers became popular [47]. A lot of scientific literature and papers were written about digital signal processing. In 2006, Christina Gherasim wrote her Ph.D. thesis on the topic ”Signal Processing for Voltage and Current Measurements in Power Quality Assessment” where she made an analysis about phases in industrial digital signal processing, extracting features from them, and made a relationship between power quality and artificial intelligence [15]. In 2011, Ariella D. Richardson wrote a Ph.D. thesis on the topic ”Mining and Classification of Multivariate Sequential Data” where she explained previous methods for multivariate time series classification with drawbacks and benefits of each [40]. In his Ph. D. thesis Mohammed Waleed Kadous made a survey about time series classification algorithms and proposed new techniques for multivariate time series classification[21]. These Ph. d. theses propose different techniques and steps what we can use for solving our problem. In his research [30], Monte proposed different signal pre-processing methods for signal analysis, what are good basis for the time-series analysis.

In 2007, Chakratborty summarized the current approaches of feature selection and classification of multivariate time-series data [8]. Fulcher and Jones explained differences between Feature-Based and Instance-Based Time-Series classification and specified challenges related to [14]. Silva, Souza, and Batista in their research [45], explained instance-based time series classification using different techniques, what we proposed in section 2.7.2. Ravikumar and Devi in their research present new methods for time series classification and compared instance-based and feature-based classification using 1-NN (Nearest Neighbour) classification algorithm [38]. In [17] authors proposed the use of dynamic SVM (Support Vector Machine) algorithm for classification of multidimensional time series. Schäfer and Leser proposed WEASEL, a novel method for time series classification [42]. In [57] authors proposed numerosity reduction for instance-based classification where they used dynamic time wrapping (DTW) as similarity measures and 1-NN as classification algorithm.

Biomedical application results can be observable as the oracle problem. An output of these applications are often different signals or set of different signals for what is needed to make a classification. This work was always reserved for the human expert who should give a final verdict if the output is good or bad. Machine learning found its place in this topic, where researchers wrote a lot of papers specifying the biggest challenges, proposing new methods in classification and feature engineering. Nanopoulos, Alcock, and Manolopoulus wrote research about proposing the use of statistical features for-time series classification [32]. They used Multi-layer perceptron (MLP) neural network as classification algorithm. In [55] authors explained drawbacks of using manual signal classification, made a comparison of different types of time series classification and proposed new method, what confirmed main motivation of this thesis. In [19] is presented a survey of ECG classification, where authors provide different techniques for pre-processing, feature extraction, ANN-based classification measures and different performance measures. Wang, Yan, and Oates in their research [56] proposed using deep neural networks for time series classification, what can be used in future work. In [36] authors proposed a technique called Neural Bag-of-Features for time series classification, what also can be investigated during future work for solving our problem.

3.2 Case study

The most appropriate method to use in this research is case study since it is commonly used to explore, describe, explain and improve upon some technique. In the book Guidelines for conducting and reporting case study research in software engineering is explained how to create a case study in software engineering [41].

1. Case study design - defining the objectives and planning the case study
2. Preparation for data collection - defining procedures and protocols for data collection
3. Collecting evidence - executing collected data on the studied case
4. Analysis of collected data
5. Reporting

3.2.1 Case company

Bombardier is one of the world's leaders in manufacturing trains and planes. In Bombardier, they develop transportation solutions as high-speed trains, business and commercial aircrafts giving attention to safety, efficiency and performance [11]. Business area of this company is split as Bombardier Transportation and Bombardier Aerospace. Bombardier Transportation provides different solutions in developing rail control solutions, transportation systems, rail vehicles, propulsion and controls, manufacturing and services, etc. In Västerås Bombardier Transportation develop different solutions for TCMS (Train Control Management System) and PPC (propulsion and controls). TCMS can be imagined as a brain of the train responsible for controlling most train functionalities, while PPC represents hearth of the train in charge for converting energy source into a form the train can use. Propulsion and Controls (PPC) is responsible for developing and developing propulsion drive systems with high reliability and low losses, what consists propulsion converters (traction and auxiliary converters) and drives (motors and gears) and TCMS [11].

3.2.2 Case and subject selection

Propulsion and Controls provided four different cases of regression testing. They are results of testing LinePowerLimitation, ProtectingActions, StartUpShutDown, SystemStability systems. Results are presented as multivariate time series data, what will be part of verdict process in developing the test oracle mechanism. We chose case Power Line Limitation to explain and implement. Testing of Power Line Limitations is used to check that line power is limited in case of too low or too high voltage. On next figure is shown the behavior of the line voltage.

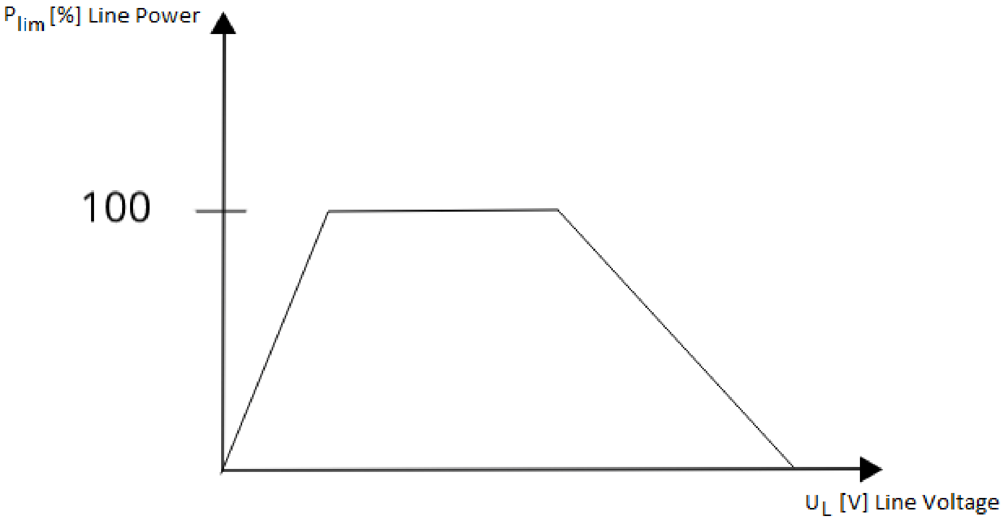


Figure 7: The behavior of the line voltage to get maximum line power

In real-time it looks like:

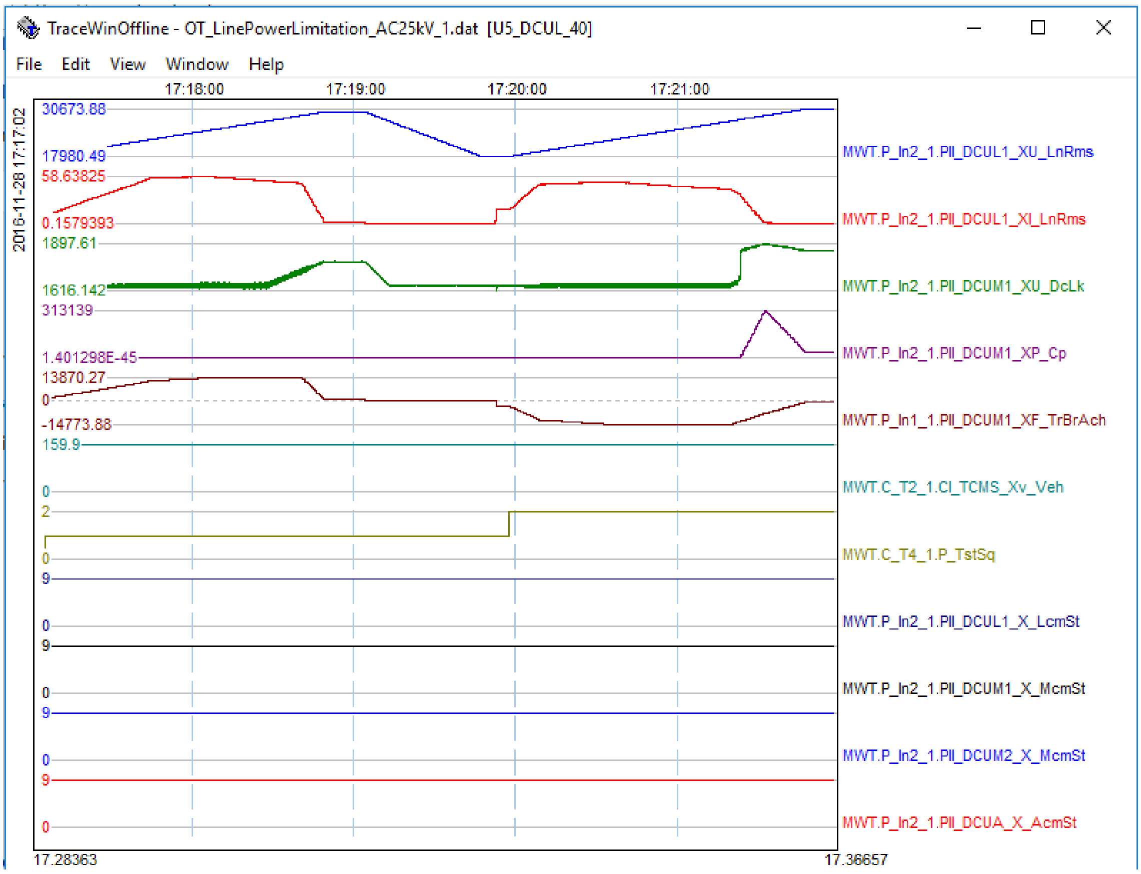

TraceWinOffline - OT_LinePowerLimitation_AC25kV_1.dat [U5_DCUL_40]
File
Edit
View
Window
Help
17:18:00
17:19:00
17:20:00
17:21:00
2016-11-28 17:17:02
30673.88
17980.49
58.63825
0.1579393
1897.61
1616.142
313139
1.401298E-45
13870.27
0
-14773.88
159.9
0
2
0
9
0
9
0
9
0
9
0
17.28363
17.36657
MWT.P_In2_1.PlI_DCUL1_XU_LnRms
MWT.P_In2_1.PlI_DCUL1_XI_LnRms
MWT.P_In2_1.PlI_DCUM1_XU_DcLk
MWT.P_In2_1.PlI_DCUM1_XP_Cp
MWT.P_In1_1.PlI_DCUM1_XF_TrBrAch
MWT.C_T2_1.CI_TCMS_Xv_Veh
MWT.C_T4_1.P_TstSq
MWT.P_In2_1.PlI_DCUL1_X_LcmSt
MWT.P_In2_1.PlI_DCUM1_X_McmSt
MWT.P_In2_1.PlI_DCUM2_X_McmSt
MWT.P_In2_1.PlI_DCUA_X_AcmSt

Figure 8: Power line limitation in real time.

what presents test cases for the oracle mechanism.
Meaning of each signal in multivariate time-series:

- MWT_P_In2_1_PII_DCUL1_XU_LnRms - Line voltage
- MWT_P_In2_1_PII_DCUL1_XI_LnRms - Line current
- MWT_P_In2_1_PII_DCUM1_XU_DcLk - DClink voltage power
- MWT_P_In2_1_PII_DCUM1_XP_Cp - Chopper power
- MWT_P_In1_1_PII_DCUM1_XF_TrBrAch - Achieved effort
- MWT_C_T2_1_CI_TCMS_Xv_Veh - Speed
- MWT_C_T4_1_P_TstSq - Test sequence number
- MWT_P_In2_1_PII_DCUL1_X_LcmSt - LCM state
- MWT_P_In2_1_PII_DCUM1_X_McmSt - MCM1 state
- MWT_P_In2_1_PII_DCUM2_X_McmSt - MCM2 state
- MWT_P_In2_1_PII_DCUA_X_AcmSt - ACM state

3.2.3 Research questions

Through research objective in 1.2, we can state research questions in a structured way:

- **RQ1:** What are current approaches of using machine learning techniques for solving test oracle problem where time series data represent verdicts?
- **RQ2:** How can the test oracle automation with machine learning approaches be implemented?

3.2.4 Data Collection

For data collection, we used direct methods, such as informal interviews with experts from PPC. Collected data is real test cases, what represent instances of different regression test iterations. In personal communication with experts, we got results of the test case execution what will represent labels or classes used in classification. A human expert from PPC provided us 20 test cases, where 13 were labeled as pass and 7 as fail. We generated 180 more artificial data with different possible cases for pass and fail tests.

3.2.5 Analysis procedures

Analysis procedures are described in Section Use Case implementation. After the case subject is selected, test cases are exported from TraceWinOffline program to .MAT files. In consultation with experts test cases are mentioned as inputs for classification and they are labeled with certain classes. In Matlab, we implemented system described in section 4.2, where we imported necessary data for classification. We used Support Vector Machine (SVM) classification algorithm. We created necessary functions for extracting and selecting important features. Classification algorithms are integrated into Matlab.

3.2.6 Evaluation

The final evaluation of the collected data is shown in section 5. We proposed different types for time-series classification and presented system prototype created in section 4.

4 Use case implementation

In this section we will make the system prototype for the case PowerLineLimitation explained in 3.2.2. In this section, we explain the architecture of the system used for the implementation of the certain use case.

4.1 System architecture

We used an approach where we created a feature vector for regression test iteration and put important features from all important signals. The oracle system will be based on black-box approach, where we need to get outputs for certain inputs.

Train your model (in effect decide hyperparameters, perhaps in your SVM the soft constraint parameter C and the kernel parameter) by looking at how it performs with different hyperparameters when you run it on the combination four parts and validate against the fifth part - each run doing this five times, against each of the folds. Choose the hyperparameters which give the optimal results in this cross-validation.

Run your model, using the chosen hyperparameters, on the whole training set. This is your final model. Test (once only) your final model on the test set to see how accurate/sensitive/specific it is on what is designed to represent out-of-sample data.

The overall architecture of the system prototype:

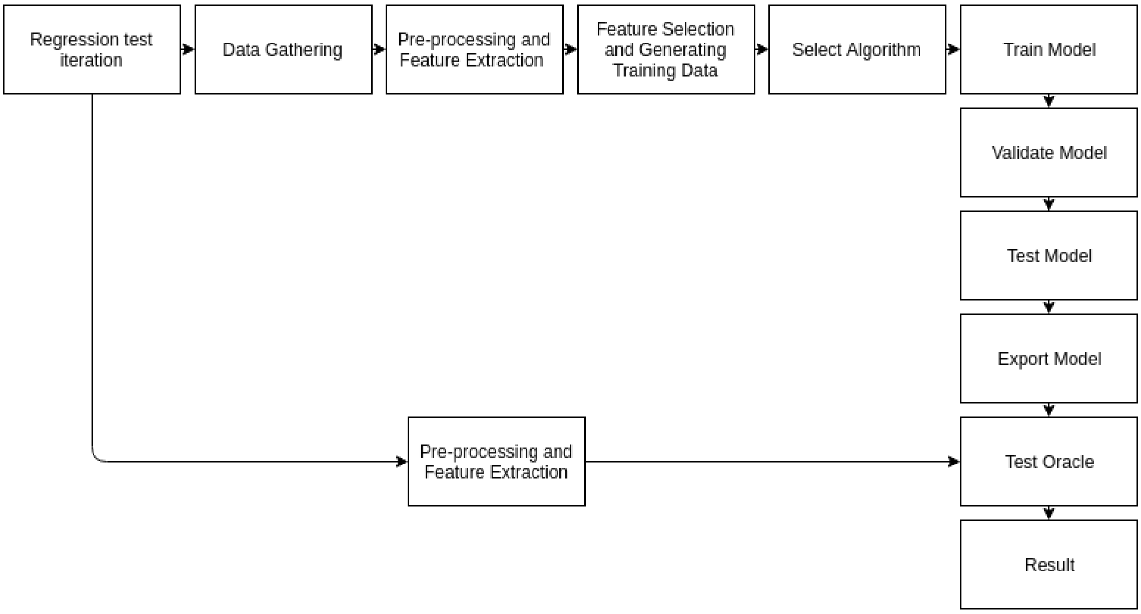

Regression test iteration
Data Gathering
Pre-processing and Feature Extraction
Feature Selection and Generating Training Data
Select Algorithm
Train Model
Validate Model
Test Model
Export Model
Pre-processing and Feature Extraction
Test Oracle
Result

Figure 9: Architecture of the system prototype for automated test oracle

4.2 System Prototype Implementation

System consists of next phases:

1. Data gathering
2. Pre-processing of the signal and feature extraction
3. Feature selection and generating feature vectors
4. Training of a model
5. Evaluation of the model
6. Prediction

4.2.1 Data gathering

Twenty regression test iterations were run by an expert from Propulsion department. They are labeled as pass or fail regression tests. Regression test iterations are shown in program TraceWinOffline, then exported as .MAT file and run in the Matlab. In consultation with the expert, we generated 180 more regression test iterations as artificial data what can be possible results of regression testing. Now we have set of 200 regression test iterations, from what is needed to extract training and test set.

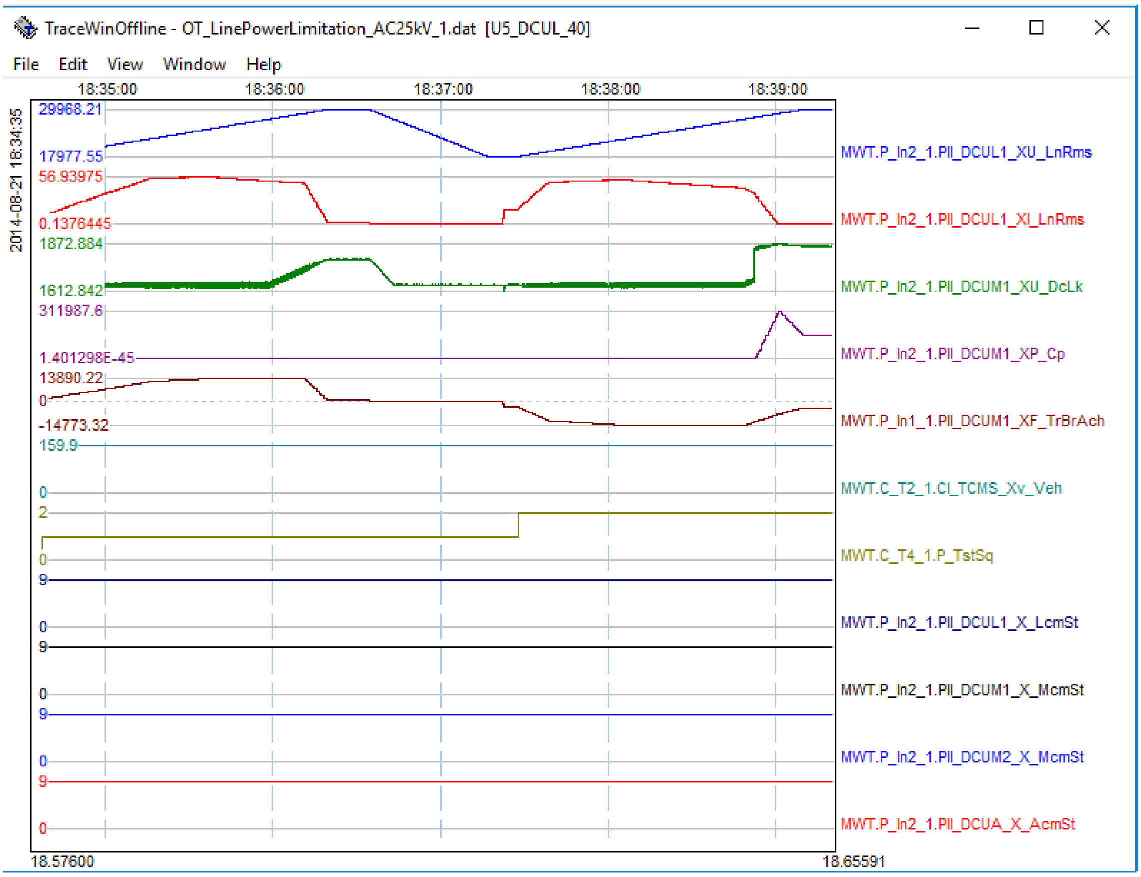

TraceWinOffline - OT_LinePowerLimitation_AC25kV_1.dat [U5_DCUL_40]
File
Edit
View
Window
Help
18:35:00
18:36:00
18:37:00
18:38:00
18:39:00
2014-08-21 18:34:35
29968.21
17977.55
MWT.P_In2_1.PII_DCUL1_XU_LnRms
56.93975
0.1376445
MWT.P_In2_1.PII_DCUL1_XI_LnRms
1872.884
1612.842
MWT.P_In2_1.PII_DCUM1_XU_DcLk
311987.6
1.401298E-45
MWT.P_In2_1.PII_DCUM1_XP_Cp
13890.22
0
-14773.32
MWT.P_In1_1.PII_DCUM1_XF_TrBrAch
159.9
0
MWT.C_T2_1.CI_TCMS_Xv_Veh
2
0
MWT.C_T4_1.P_TstSq
9
0
MWT.P_In2_1.PII_DCUL1_X_LcmSt
9
0
MWT.P_In2_1.PII_DCUM1_X_McmSt
9
0
MWT.P_In2_1.PII_DCUM2_X_McmSt
9
0
MWT.P_In2_1.PII_DCUA_X_AcmSt
18.57600
18.65591

Figure 10: Iteration of regression testing for PowerLineLimitation

4.2.2 Pre-processing of the signal and feature extraction

Pre-processing represents removing redundant features and noise from the raw data, what is presented as signals in our case. In consultation with the expert, we understood important features for different signal in regression test iteration. For feature extraction of different signals, we need different approaches. For signals:

- MWT_P_In2_1_PII_DCUL1_X_LcmSt
- MWT_P_In2_1_PII_DCUM1_X_McmSt
- MWT_P_In2_1_PII_DCUM2_X_McmSt
- MWT_P_In2_1_PII_DCUA_X_AcmSt

it is needed to extract statistical features, such as minimum, maximum and average value. Constant signals:

- MWT_P_In2_1_PII_DCUL1_XU_LnRms

- MWT_P_In2_1_PII_DCUL1_XI_LnRms
- MWT_P_In2_1_PII_DCUM1_XU_DcLk
- MWT_P_In2_1_PII_DCUM1_XP_Cp
- MWT_P_In1_1_PII_DCUM1_XF_TrBrAch

needs pre-processing to remove noise from the signal. We use Matlab function *smoothData()*. In general, we use statistical features such as minimum, maximum, average value of some signal, and number of peaks in some time period.

Signals:

- MWT_C_T2_1_CI_TCMS_Xv_Veh
- MWT_C_T4_1_P_TstSq

are always same and they will not use in classification.

On Github repository repository [18] are displayed Matlab methods for feature extraction of different signals.

4.2.3 Feature Selection

After features are extracted it is needed to select them and generate a feature vector for every regression test iteration. Feature vectors are inserted in a feature matrix used for training of classification model. Generating of the feature matrix is displayed also on Github repository [18]. Feature matrix is exported to a table, what represent input for the classification model. We will add a new column to feature matrix what will represent output labels for each input. Feature selection has a lot of benefits related to machine learning model, such as faster training of the model, reducing its complexity and overfitting, and improving its accuracy.

After feature selection and generating a table, it is needed to create training, validation and test set. For test set, we prepared 20 regression test iterations, what represent 10% of the dataset. Another 90% percent of the dataset is reserved for training and validation. For the demonstration, we will use all 28 features what are extracted from the raw data. Also, we made the comparison of the results in the cases where we used different features. Features are selected before the training.

For the comparison, we will select different features. First feature selection is based to get only one feature from every signal and use them to train the model. Next feature selection is based to use features from the signals:

- MWT_P_In2_1_PII_DCUL1_X_LcmSt
- MWT_P_In2_1_PII_DCUM1_X_McmSt
- MWT_P_In2_1_PII_DCUM2_X_McmSt
- MWT_P_In2_1_PII_DCUA_X_AcmSt.

Results are shown through section 4.2.6 and 5, where we used same test sets to get a final accuracy of the model.

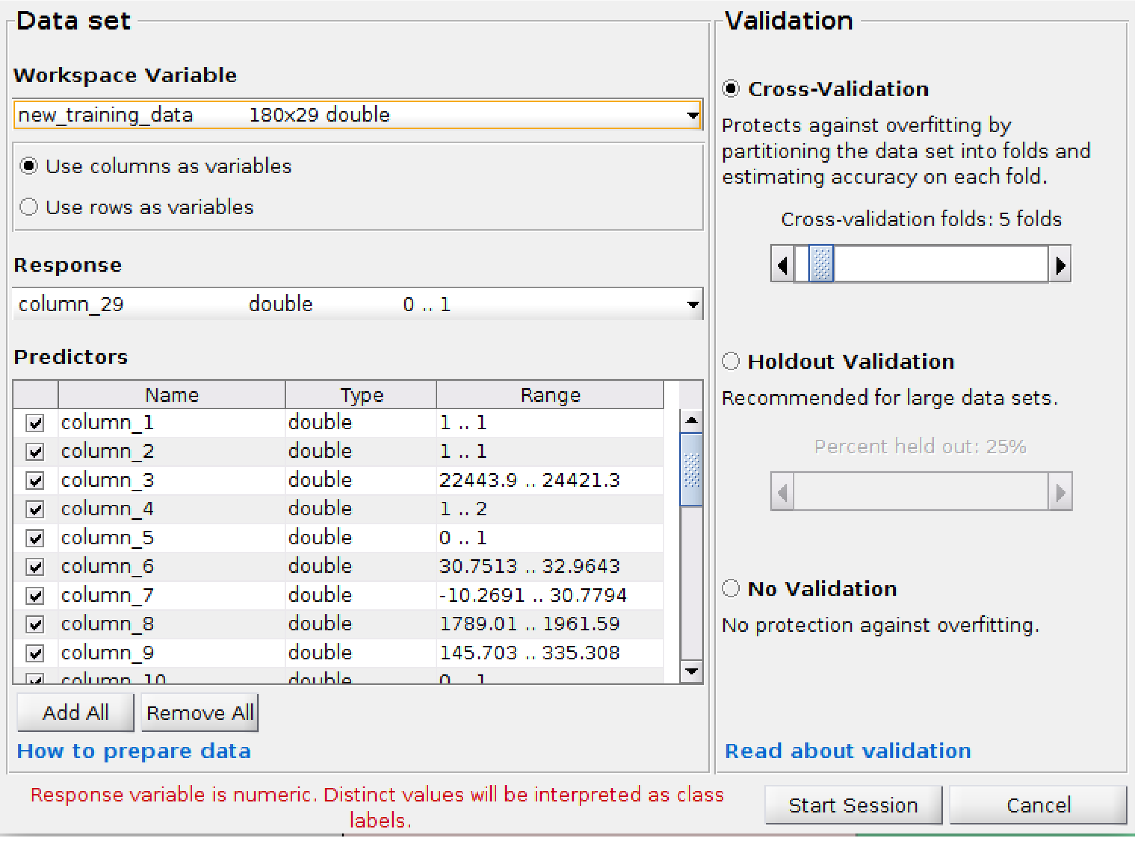
4.2.4 Training of a model

After the train set is generated, it is needed to have set of outputs(labels) what show whether regression test iteration passes or fails. Next part of this phase is to choose classification algorithm, where is needed to pay attention to next characteristics (explained on Matlab website):

- seed of training
- predictive accuracy on new data

- memory usage
- transparency and interpretability

Matlab has Classification Learner Application what we will use to create machine learning model. We should select input data for the training of the model. We will choose a table where we extracted feature matrix together with labels for each input.



Data set

Workspace Variable

new_training_data 180x29 double

● Use columns as variables

○ Use rows as variables

Response

column_29 double 0 .. 1

Predictors

☐	Name	Type	Range
☑	column_1	double	1 .. 1
☑	column_2	double	1 .. 1
☑	column_3	double	22443.9 .. 24421.3
☑	column_4	double	1 .. 2
☑	column_5	double	0 .. 1
☑	column_6	double	30.7513 .. 32.9643
☑	column_7	double	-10.2691 .. 30.7794
☑	column_8	double	1789.01 .. 1961.59
☑	column_9	double	145.703 .. 335.308
☑	column_10	double	0 .. 1

How to prepare data

Validation

● **Cross-Validation**

Protects against overfitting by partitioning the data set into folds and estimating accuracy on each fold.

Cross-validation folds: 5 folds

○ **Holdout Validation**

Recommended for large data sets.

Percent held out: 25%

○ **No Validation**

No protection against overfitting.

Read about validation

Response variable is numeric. Distinct values will be interpreted as class labels.

Figure 11: Classification Learner Application after selected training set

First 28 columns represent input data for classification, while column 29 is the label for each input. For the validation of the system, we used 5-fold cross-validation.

For classification, we will use Linear Support Vector Machine (SVM) classification algorithm, what means that we use linear kernel function.

4.2.5 Validation of the model

Evaluation of the model shows us the final performances and accuracy of the output model. Before we get final accuracy we should validate the model originated on the training data. Validation represents the process of verifying if the model is good enough to make predictions. After successfully training from training data set it is needed to check the accuracy of the model on a labeled dataset, what is not used for training of the model. Cross-validation is a technique for evaluating and comparing machine learning algorithms. It is done by dividing the remaining 90% labeled data on training set and validation set [39]. The most common used cross-validation example is k-fold cross-validation. In k-fold cross-validation, input data are divided into k folds (subsets of data). k - 1 folds are used for training and 1 fold is used for validation. This process is repeated k times, where we use different folds for evaluation each time. The process of the cross-validation is used to remove the overfitting of the model. For the training and validation of our model we used 5-fold cross-validation.

4.2.6 Testing of the model

Testing of the model is reserved to get final performances of our model. We apply the model to the data that are not used in the process of training and validation. 10% of the data (20 regression test iterations) was reserved for the testing machine learning model, from which half is marked as ”pass”, while another half is ”fail”. The result is shown as the confusion matrix.

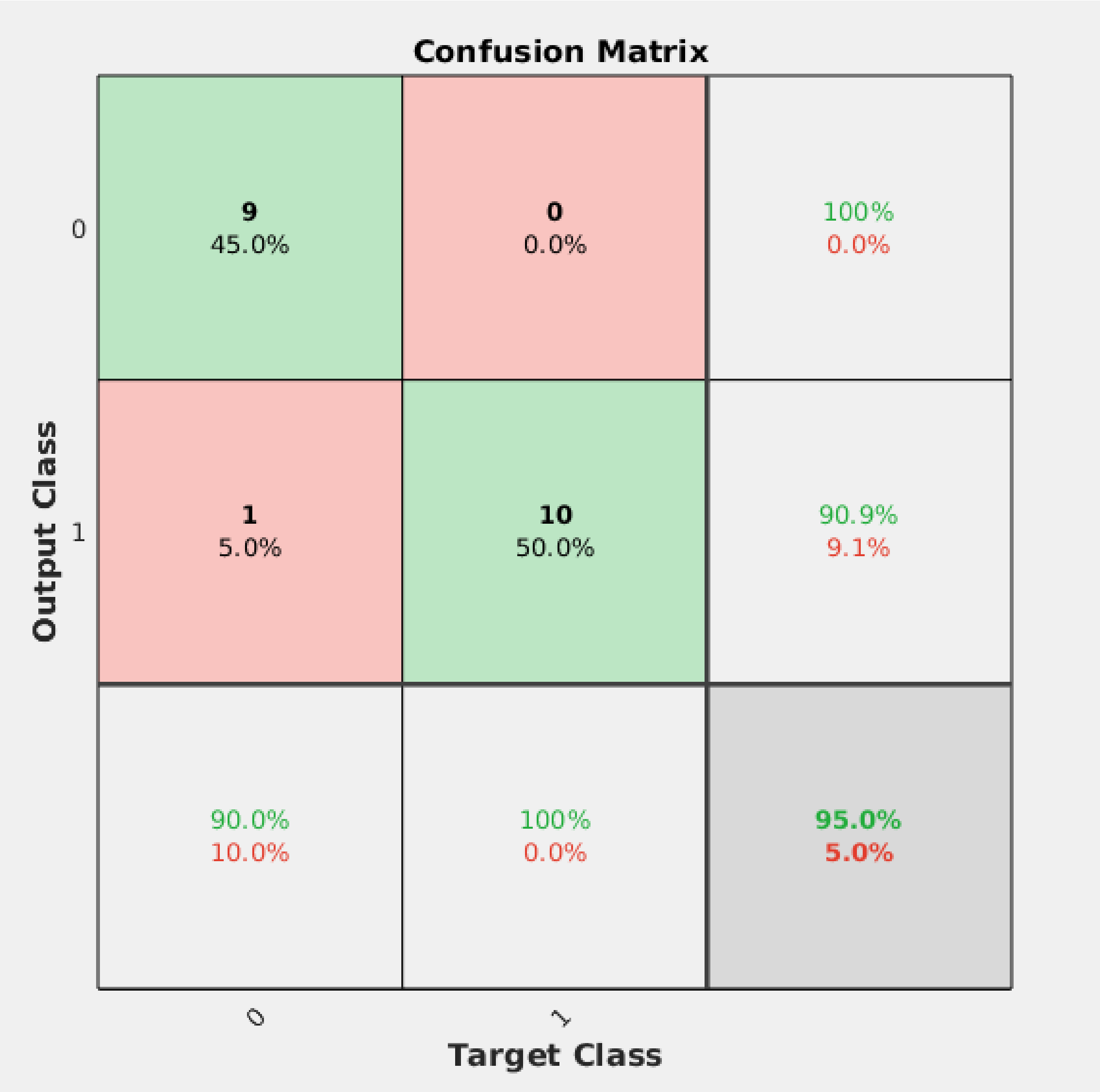


Figure 12: Confusion matrix

On the confusion matrix, Target Class represent real values of the test iterations, while Output class represents the result of the prediction of our model. Final accuracy of our model is 95%. Sensitivity is 90.9 % (percentage of positive values in values for what is predicted to be positive). Specificity is 100 % (percentage of negative values in values for what is predicted to be negative).

Now we present results for the case were we select one feature for every signal. We selected 9 different features. The accuracy of our model is 75 %. Sensitivity is 66.7 %, while specificity is 100 %.

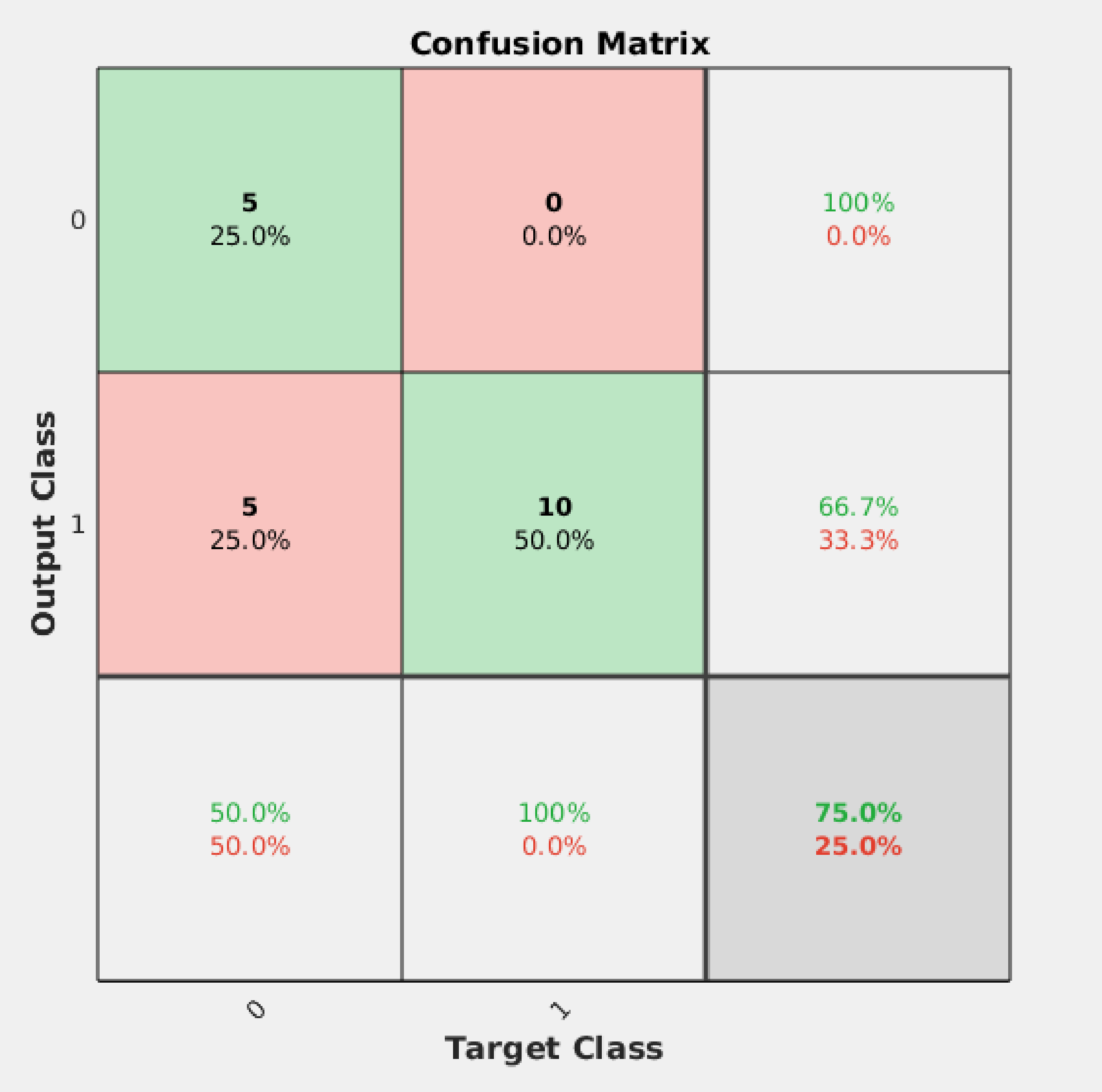
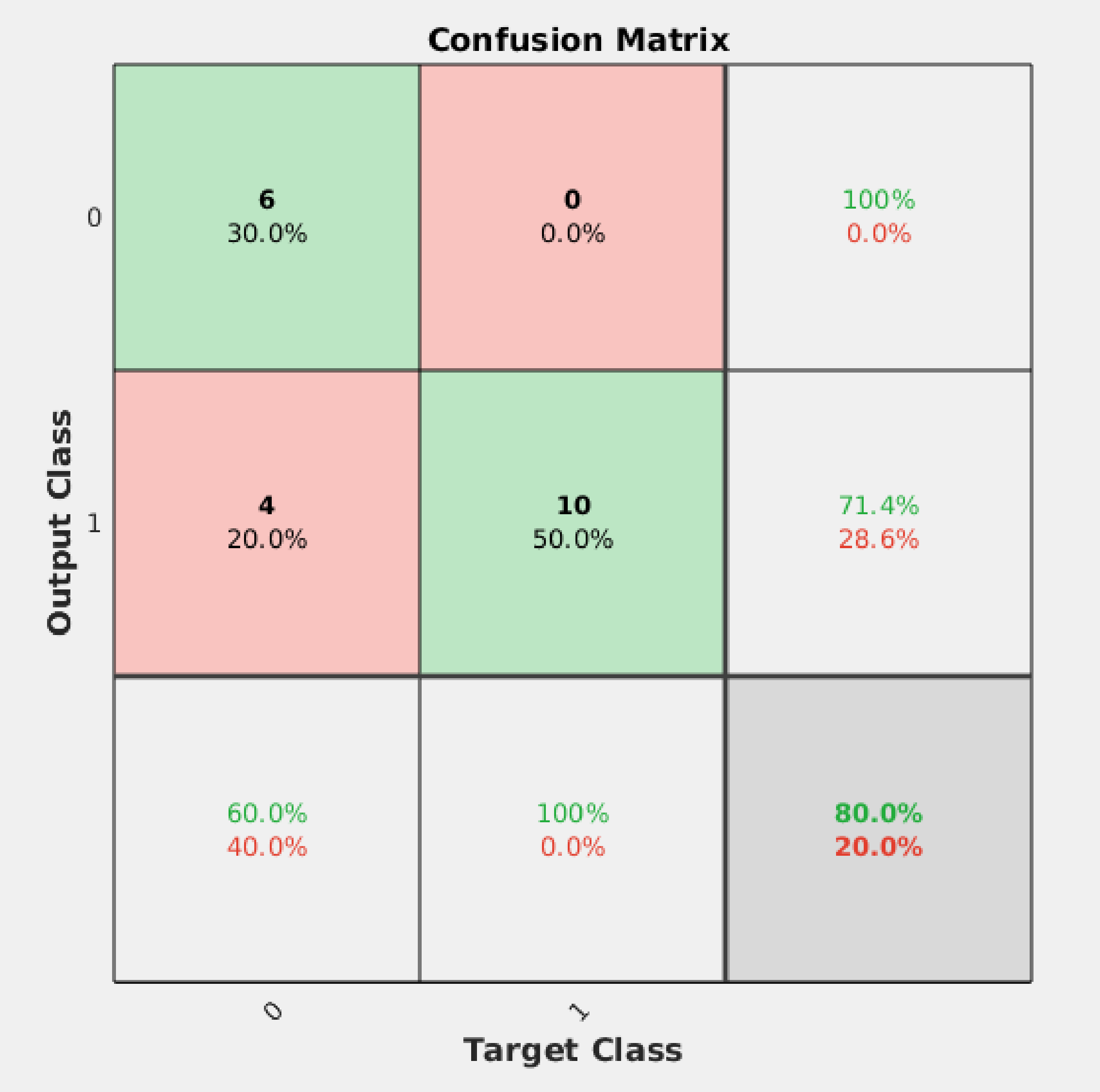


Figure 13: Confusion matrix with selected one feature for every signal

Next comparison is reserved for the model trained on the features from the next signals:

- MWT_P_In2_1_PII_DCUL1_X_LcmSt
- MWT_P_In2_1_PII_DCUM1_X_McmSt
- MWT_P_In2_1_PII_DCUM2_X_McmSt
- MWT_P_In2_1_PII_DCUA_X_AcmSt.

Final accuracy is 80%, while sensitivity is 71.4% and specificity is 100%. Results are shown on the confusion matrix.



	0	1	
0	6 30.0%	0 0.0%	100% 0.0%
1	4 20.0%	10 50.0%	71.4% 28.6%
	60.0% 40.0%	100% 0.0%	80.0% 20.0%

Figure 14: Confusion matrix with selected one feature for the first four signals

4.2.7 Prediction

After the model is evaluated successfully and we got wanted accuracy it is possible to deploy our model. In our case, it means that human expert can use this model for prediction of execution of next regression testing iteration. We can export a model from Classification Learner Application and use it for prediction. We should import certain regression test iteration and extract same features what we used for training of the model.

Machine learning model is exported and displayed in Github repository [18]. We made simple Matlab Graphical User Interface to present the oracle problem. For testing of our system, we took one instance of regression testing.

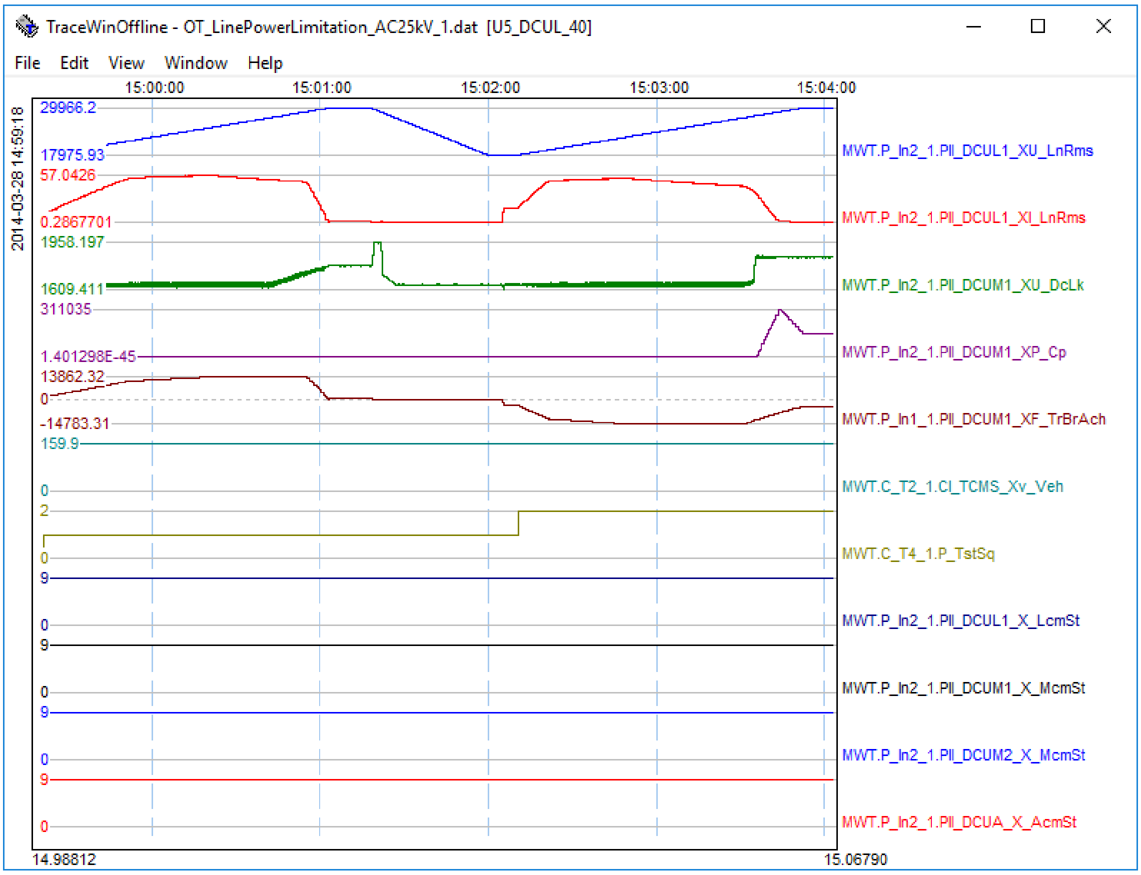
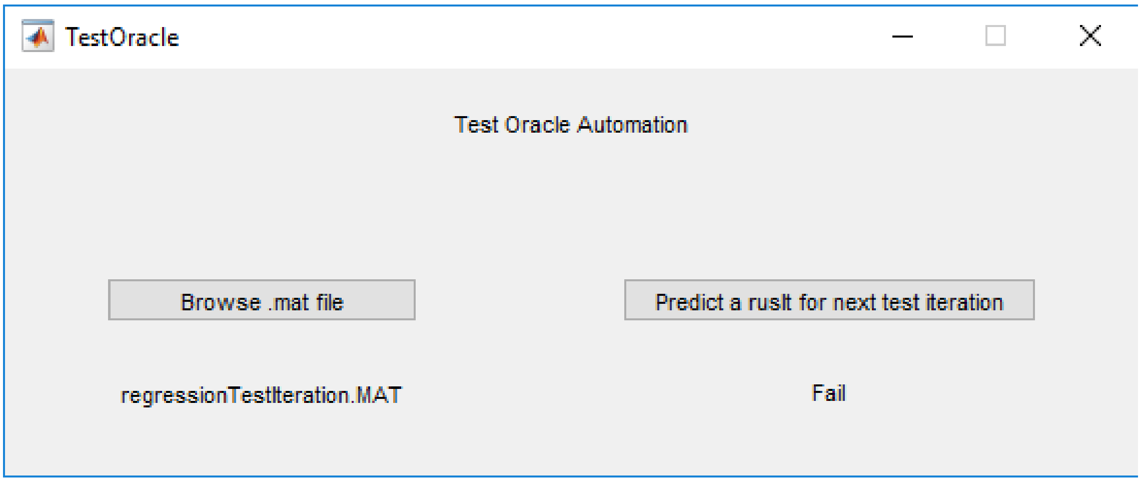


Figure 15: Failed regression test iteration

In discussion with the domain expert, this iteration is labeled as failed. We exported this iteration to a .mat file. In our simple GUI, we tested this iteration. The code for prediction data is presented in the Github repository [18].



regressionTestIteration.MAT

Fail

Figure 16: Result of the automated oracle system

Our system showed us that this test iteration is failed.

5 Results and Discusions

This section presents results obtained in this research. The aim of this thesis is to examine the feasibility of the using machine learning approach for solving the test oracle problem.

According to RQ1 (current approaches to solving the test oracle problem where time series data represent parts of verdict process), we presented different techniques used for time series classification. This RQ is answered through related work in Section 3.1.1. We explained relationships between test oracle automation and signal classification. Signal classification can be observed as the test oracle problem because it is always needed to be classified by the domain expert. In different types of industries, this process had some chances to automate at least to help domain expert in solving the problem of signal classification. In 2 we presented main challenges and steps of using machine learning approaches. We specified different methods for specific and extract the best for solving our use case. For the different use cases from case company, we had problems where multivariate time-series data is needed to be classified. We proposed to feature-based multivariate-time series classification for solving this type of problems.

With respect to RQ2, we created system prototype in section 4. In this prototype, we proposed the best way of using the machine learning techniques for solving test oracle problem. We presented necessary steps for successful multivariate time series classification. For the demonstration it is extracted 28 different features from the multivaraite time-series data. After generating feature matrix and importing it to training data. We used Matlab application called Classification Learner to create our machine learning model. After *5*-fold cross-validation, we have 95% accuracy of the machine learning model. For the demonstration, we created simple GUI application to show the classification of the new regression test iteration. Results are presented through confusion matrix in section 4.2.6. Sensitivity of the our model is 90.1%, while specifity is 100%. We also created two different models what are trained using different features. First, where we had one feature for every signal accuracy is 75 %, sensitivity is 66.7 % while specificity is 100 %. Model what is trained by features related for the first four signals has the accuracy 80%, while sensitivity is 71.4% and specificity is 100%. As the model is trained with a relatively small amount of the data, overfitting is handled with cross-validation.

6 Limitations

The limitations with short discussion are listed as following:

- We don't always have a guarantee that machine learning algorithms will work properly.
- Generally, machine learning algorithms often require a lot of training data, what can be a challenge to collect them.
- It is not possible to apply machine learning algorithms to the general problem, every problem needs to have an individual approach.
- Gathered data cannot be applied directly to machine learning algorithms, they need to be clean and prepared properly.
- For the successful application of machine learning algorithms, it is necessary to understand the problem.
- Only twenty iterations of regression testing are provided by the PPC, what induced to create new 180 artificial data.
- Provided multivariate time series contains just one signal that causes failing test. In consultation with the expert, we generated possible failing test cases caused by another signal in the multivariate time series.
- We extracted only 28 features what are used for generating data used for training.

7 Conclusion and Future work

Machine learning approach for solving the test oracle can never replace a domain expert completely, buy it can facilitate this process. In this thesis, we examined the feasibility of using machine learning approach for solving the test oracle problem. As main research goal, we showed how machine learning approach can be used for the solving the test oracle problem. Through literature review, we showed a relationship between test oracle problem and time series classification. In the background, we proposed two different methods for time-series classification, instance-based and feature-based, and feature-based multivariate time-series classification. We explained different steps in multivariate time-series classification and implemented the prototype of the system for solving the test oracle problem.

This feasibility study can be a good base for the future research on the similar topics. It is possible to investigate adding one more class of the test results what can be called "indefinite", where expert always should give a final verdict. For the future implementation of the system, it will be good to have minimum 200 iterations of regression testing with all possible cases what can cause that test iteration is failed. It will be good to extract new different and more appropriate features of the different signals in test iterations. It is also possible to use different learning algorithms, what can have better accuracy. As the final challenge for future work, it will be to investigate and implement case-based reasoning for solving this type of problems.

References

[1] D. Agarwal, D. E. Tamir, M. Last, and A. Kandel. A comparative study of artificial neural networks and info-fuzzy networks as automated oracles in software testing. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 42(5):1183–1193, Sept 2012.

[2] C. C. Aggarwal. *Data Classification: Algorithms and Applications*. Chapman & Hall/CRC, 1st edition, 2014.

[3] R. Almaghairbe and M. Roper. Automatically classifying test results by semi-supervised learning. In *2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE)*, pages 116–126, Oct 2016.

[4] S. Angra and S. Ahuja. Machine learning and its applications: A review. In *2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC)*, pages 57–60, March 2017.

[5] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo. The oracle problem in software testing: A survey. *IEEE Transactions on Software Engineering*, 41(5):507–525, May 2015.

[6] A. Bertolino. Software testing research and practice. In *Proceedings of the Abstract State Machines 10th International Conference on Advances in Theory and Practice*, ASM'03, pages 1–21, Berlin, Heidelberg, 2003. Springer-Verlag.

[7] L. C. Briand. Novel applications of machine learning in software testing. In *2008 The Eighth International Conference on Quality Software*, pages 3–10, Aug 2008.

[8] B. Chakraborty. Feature selection and classification techniques for multivariate time series, 10 2007.

[9] T. Cheatham, J. Yoo, and N. J. Wahl. Software testing: A machine learning experiment., 01 1995.

[10] P. Cunningham, M. Cord, and S. J. Delany. *Supervised Learning*, pages 21–49. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[11] E. Ekblad. *Continuous Improvements During Project Based Production: A Case Study Executed at Bombardier Transportation*. Lule University of Technology, 2015.

[12] Emelie Engström and Per Runeson. A qualitative survey of regression testing practices. In *Proceedings of the 11th International Conference on Product-Focused Software Process Improvement*, PROFES'10, pages 3–16, Berlin, Heidelberg, 2010. Springer-Verlag.

[13] M. Fagerström, E. E. Ismail, G. Liebel, R. Guliani, F. Larsson, K. Nordling, E. Knauss, and P. Pelliccione. Verdict machinery: On the need to automatically make sense of test results. In *Proceedings of the 25th International Symposium on Software Testing and Analysis*, ISSTA 2016, pages 225–234, New York, NY, USA, 2016. ACM.

[14] Ben D. Fulcher and Nick S. Jones. Highly comparative, feature-based time-series classification. *CoRR*, abs/1401.3531, 2014.

[15] C. Gherasim. *Signal Processing for Voltage and Current Measurements in Power Quality Assessment*. Katholieke Universiteit Leuven, 2006.

[16] W. E. Howden. Theoretical and empirical studies of program testing. *IEEE Transactions on Software Engineering*, SE-4(4):293–298, July 1978.

[17] R. Huerta, S. Vembu, M. Muezzinoglu, and A. Vergara. Dynamical svm for time series classification, 08 2012.

[18] N. Imamovic. Test oracle automation with machine learning: A feasibility study. `https://github.com/NerminImamovic/testOracleAutomation`, 2018.

[19] S. H. Jambukia, V. K. Dabhi, and H. B. Prajapati. Classification of ecg signals using machine learning techniques: A survey. In *2015 International Conference on Advances in Computer Engineering and Applications*, pages 714–721, March 2015.

[20] M. Jordan and T.M. Mitchell. Machine learning: Trends, perspectives, and prospects. 349:255–60, 07 2015.

[21] M. W. Kadous. *Temporal Classification: Extending the Classification Paradigm to Multivariate Time Series*. The University of New South Wales, 2002.

[22] P. Kalapatapu, S. Goli, P. Arthum, and A. Malapati. A study on feature selection and classification techniques of indian music. *Procedia Computer Science*, 98:125 – 131, 2016. The 7th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2016)/The 6th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2016)/Affiliated Workshops.

[23] U. Karrenberg. *Signals in the time and frequency domain*, pages 33–64. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[24] P. Laplante, F. Belli, J. Gao, G. Kapfhammer, K. Miller, W. E. Wong, and D. Xu. Software test automation. 2010, 01 2010.

[25] ISTQB Level, Agile Tutorial, 2018 Dates, ISTQB Tests, Contact Us, Privacy Policy, Terms Use, About Us, and Write us. What is a test case?, 2018.

[26] D. Lo, H. Cheng, J. Han, S. Khoo, and C. Sun. Classification of software behaviors for failure detection: a discriminative pattern mining approach. In *KDD*, 2009.

[27] P. D. Machado and W. L. Andrade. The oracle problem for testing against quantified properties. In *2007 7th International Conference on Quality Software(QSIC)*, volume 00, pages 415–418, 10 2007.

[28] R. Mathur, S. Miles, and M. Du. Adaptive automation: Leveraging machine learning to support uninterrupted automated testing of software applications. *CoRR*, abs/1508.00671, 2015.

[29] P. McMinn, M. Stevenson, and M. Harman. Reducing qualitative human oracle costs associated with automatically generated test data. In *Proceedings of the First International Workshop on Software Test Output Validation*, STOV '10, pages 1–4, New York, NY, USA, 2010. ACM.

[30] G. Monte. Sensor signal preprocessing techniques for analysis and prediction. In *2008 34th Annual Conference of IEEE Industrial Electronics*, pages 1788–1793, Nov 2008.

[31] F. A. Muhammed. *An Introduction to UMTS Technology : Testing, Specifications, and Standard Bodies for Engineers and Managers*. BrownWalker Press, 2008.

[32] A. Nanopoulos, R. Alcock, and Y. Manolopoulos. Feature-based classification of time-series data. 10:49–61, 01 2001.

[33] F. Nargesian, H. Samulowitz, U. Khurana, E. B. Khalil, and D. Turaga. Learning feature engineering for classification, 08 2017.

[34] R. T. Olszewski. *Generalized Feature Extraction for Structural Pattern Recognition in Time-series Data*. PhD thesis, Pittsburgh, PA, USA, 2001. AAI3040489.

[35] P. Ongsulee. Artificial intelligence, machine learning and deep learning. In *2017 15th International Conference on ICT and Knowledge Engineering (ICT KE)*, pages 1–6, Nov 2017.

[36] N. Passalis, A. Tsantekidis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis. Time-series classification using neural bag-of-features. *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 301–305, 2017.

[37] M. Polo, P. Reales, M. Piattini, and C. Ebert. Test automation. *IEEE Software*, 30(1):84–89, Jan 2013.

[38] P. Ravikumar and V. S. Devi. Weighted feature-based classification of time series data. In *2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pages 222–228, Dec 2014.

[39] P. Refaeilzadeh, L. Tang, and H. Liu. Cross-validation. 532538:532–538, 01 2009.

[40] A. D. Richardson. *Mining and Classification of Multivaraite Sequential Data.* Bar-Ilhan University, 2011.

[41] P. Runeson and M. Hst. Guidelines for conducting and reporting case study research in software engineering. 14(2):131.

[42] P. Schäfer and U. Leser. Fast and accurate time series classification with weasel. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pages 637–646, New York, NY, USA, 2017. ACM.

[43] S. R. Shahamiri, W. M. N. Wan-Kadir, S. Ibrahim, and S. Z. M. Hashim. Artificial neural networks as multi-networks automated test oracle. *Automated Software Engineering*, 19(3):303–334, Sep 2012.

[44] R. H. Shumway and D. S. Stoffer. *Time Series Analysis and Its Applications: With R Examples.* Springer Texts in Statistics. Springer International Publishing, 4 edition.

[45] D. F. Silva, V. M. A. D. Souza, and G. E. A. P. A. Batista. Time series classification using compression distance of recurrence plots. In *2013 IEEE 13th International Conference on Data Mining*, pages 687–696, Dec 2013.

[46] P. Sinha. *Speech Processing in Embedded Systems.* 01 2010.

[47] S. W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing.* California Technical Publishing, San Diego, CA, USA, 1997.

[48] O. Sutton. Introduction to k nearest neighbour classification and condensed nearest neighbour data reduction, 2012.

[49] J. Tang, S. Alelyani, and H. Liu. Feature selection for classification: A review.

[50] A. Theissler. *Detecting Anomalies in Multivariate Time Series from Automotive Systems.* Brunel University, 2013.

[51] M. Vanmali, M. Last, and A. Kandel. Using a neural network in the software testing process. 17:45–62, 01 2002.

[52] E. Volna, M. Kotyrba, and M. Janosek. *Pattern Recognition and Classification in Time Series Data.* IGI Global, Hershey, PA, USA, 1st edition, 2016.

[53] F Wang, S. Yang, and Y. Yang. Regression testing based on neural networks and program slicing techniques. In Y. Wang and T. Li, editors, *Practical Applications of Intelligent Systems*, pages 409–418, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[54] F. Wang, L. W. Yao, and J. H. Wu. Intelligent test oracle construction for reactive systems without explicit specifications. In *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, pages 89–96, Dec 2011.

[55] J. Wang, P. Liu, M. She, S. Nahavandi, and A. Kouzani. Bag-of-words representation for biomedical time series classification. 8, 12 2012.

[56] Z. Wang, W. Yan, and T. Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1578–1585, May 2017.

[57] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana. Fast time series classification using numerosity reduction. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 1033–1040, New York, NY, USA, 2006. ACM.

[58] M. E. Yousif, S. R. Shahamiri, and M. B. Mustafa. Test oracles based on artificial neural networks and info fuzzy networks: A comparative study. In *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, pages 467–471, June 2015.

[59] H. Yu and S. Kim. Svm tutorial - classification, regression and ranking. In Grzegorz Rozenberg, Thomas Bck, and Joost N. Kok, editors, *Handbook of Natural Computing*, pages 479–506. Springer, 2012.