Weld Process Free Burn Estimator

- Weld Process State Classification -

Master Thesis

CA10-1036

Aalborg University
Control and Automation

AALBORG UNIVERSITY
STUDENT REPORT

Control and Automation
Aalborg University
http://www.aau.dk

Title:
Weld Process Free Burn Estimator

Project Period:
Spring Semester 2020

Project Group:
CA10-1036

Participant(s):
Peter Aurelius Munk
Emil Færgemand Bøgh

Supervisor(s):
Kirsten Mølgaard Nielsen, AAU
Tom Søndergaard Pedersen, AAU

Copies: 1

Page Numbers: 118

Date of Completion:
June 3, 2020

Abstract:

This thesis investigates the problem of predicting free burn of the short-circuit phase of a conventional GMAW Short-Circuit welding process. The problem is formulated as a binary classification problem where data samples in the last period of the short-circuit phase should be classified differently from the prior samples of the short-circuit phase. In addition to the measurement data provided from the welding process, a set of features is extracted from the measurement data to support the classification. A state observer is designed for the welding process in the short-circuit phase, and the residual for the control voltage is added to the feature set. Several classification models are compared in terms of binary classification performance measures and the K-Nearest Neighbor m preformed the best. In testing, following optimisation of model hyperparameters, performance is acceptable on test data of the same origin as the training data, however, the model deliver an unacceptable performance on test data from different variations of the welding process. For acceptable performance on variations of the welding process, the classification model must be trained on the same variations of welding measurement data.

Contents

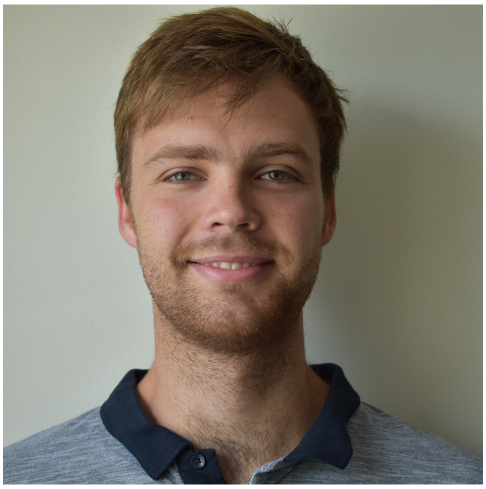
Preface

The thesis was written at Aalborg University as a 4^{th} semester master thesis in Control and Automation, in collaboration with the company Migatronic A/S, by the authors listed below:

Aalborg University, June 3, 2020

Peter Aurelius Munk

Peter Aurelius Munk
<pmunk15@student.aau.dk>



Emil Færgemand Bøgh
<ebagh15@student.aau.dk>

Chapter 1

Introduction

This chapter contains a short introduction to welding followed a statement of the problem for the work behind this thesis, and a thesis outline. The problem statement is one proposed by the company Migatronic A/S. Migatronic A/S is a danish company which produces a variety of metal working equipment, such as Metal Inert Gas, Metal Active Gas and Tungsten Inert Gas welders [1].

1.1 Introduction to Welding

Welding is a process which fuses metal pieces by applying heat, and in some cases pressure and/or a filler metal, creating what is known as a weld [2]. The definition of a weld is:

A localized coalescence of metals or nonmetals produced either by heating materials to the welding temperature, with or without the application of pressure, or by the application of pressure alone and with or without the use of filler metal [2].

If the welding process is successful, then the resulting metal piece can be considered as one solid piece instead of two joint pieces. There exists several different types of welding techniques, all with different benefits and drawbacks. The most common of these techniques can be categorised as [2]:

- **Arc welding with non-consumable electrode**
  - Gas Tungsten Arc Welding (GTAW)
  - Plasma Arc Welding (PAW)
- **Arc welding with consumable electrode**
  - Shielded Metal Arc Welding (SMAW)

  - Gas Metal Arc Welding (GMAW)
  - Flux-Cored Arc Welding (FCAW)
  - Submerged Arc Welding (SAW)

- **Other welding techniques**
  - Oxyfuel Gas Welding (OFW)
  - Thermit Welding (TW)
  - Solid-State Welding (SSW)
  - Resistance Welding (RW)
  - Electron Beam Welding (EBW)
  - Laser Beam Welding (LBW)

This thesis will focus on the Gas Metal Arc Welding technique, and any other welding technique is therefore not described further.

1.1.1 Gas Metal Arc Welding

In Gas Metal Arc Welding (GMAW) a consumable electrode is used. An electrical arc is established between the electrode, often the anode, and the workpiece, acting as the cathode, which melts the electrode causing drop growth and detachment from the electrode tip. The consumed electrode is continuously replaced by feeding new electrode material as a wire, pushed forward by a wire feeding system. To protect the weld pool from contamination, the process is protected from the ambient air by a shielding gas [3]. An overview of the GMAW process is shown in Figure 1.1.

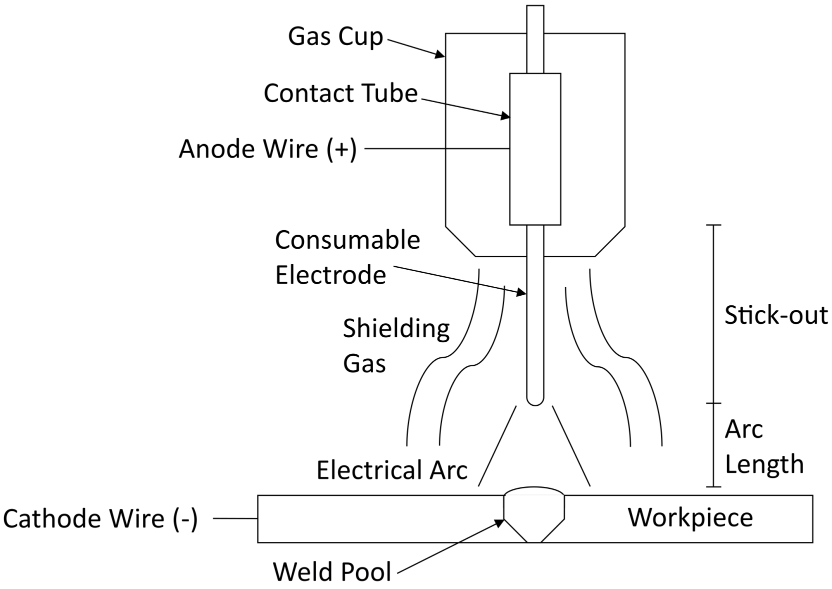


Figure 1.1: The GMAW or MIG/MAG process

Two subtypes of GMAW exists, which are defined by the type of shielding gas used. Both methods use gas bottles to provide the shielding gas, which is fed through the welding gun, to protect the weld area.

Metal Inert Gas (MIG) welding uses an inert gas or gas mixture to shield the weld area, such as argon and helium. Metal Active Gas (MAG) welding uses an active gas or gas mixture to shield the weld area, often a mixture between carbon dioxide, argon and oxygen. The shielding gas is important to achieve a good weld, as it affects the arc stability, metal transfer and degree of spatter. Furthermore, shielding gases can also impact the penetration of the weld [4]. Figure 1.2 shows the basic equipment components required in a gas metal arc welding setup.

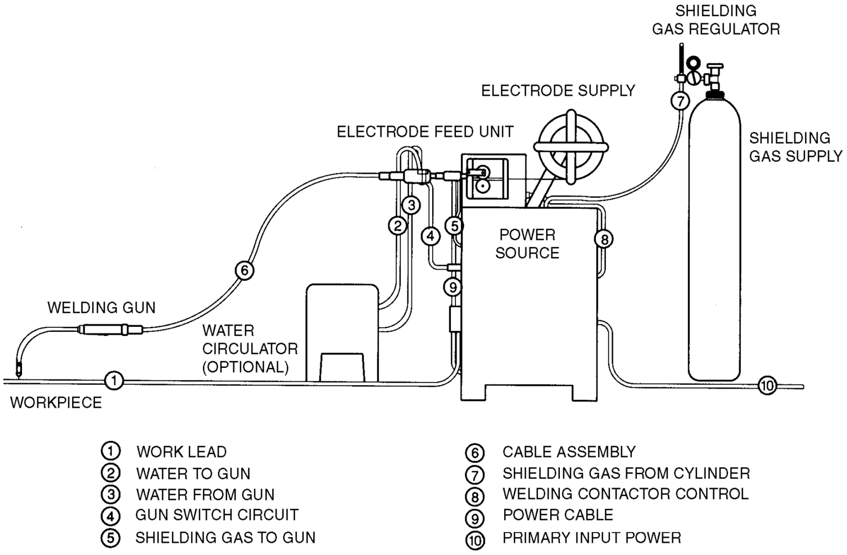


Figure 1.2: Basic equipment components in a gas metal arc welding (GMAW) setup [5].

There exists different transfer modes for both MIG and MAG welding. The different modes each offers different benefits and drawbacks. The different modes are often achieved by regulation of the current of a constant-voltage power supply, but in some cases a special power supply is needed.

- **Spray transfer:** Some times also called axial spray, is a high current transfer mode, where small drops of molten metal is transferred at a relatively high frequency. No short circuits between the electrode and workpiece occurs during the transfer. This results in a large weld pool and a high amount of heat on the workpiece. Spray transfer is limited to horizontal positions and thicker metals due to the high amount of heat [2].

- **Globular transfer:** Some times also called drop transfer or repelled transfer, is a medium to low current transfer mode, where normally larger drops of molten metal is transferred at a relatively low frequency. Short circuits between electrode and workpiece do occur during the welding. The resulting weld is not as smooth as for spray transfer and produces more spatter. Globular transfer is normally used in horizontal position for steel welding [2].

- **Pulsed-Spray transfer:** This is a variation of the Spray transfer mode, that uses a pulsing power supply. Molten metal is transferred as small droplets at a fixed frequency. The transfer frequency is determined by the pulse duration of the power supply. By pulsing the current a lower amount of heat is applied to the workpiece, allowing Pulsed-Spray transfer to weld thinner material than with regular Spray transfer, this however results in an increased number of settings to regulate for a new workpiece, reducing the ease of use [2].

- **Short-Circuit transfer:** Some times also called short arc or dip transfer, is a low current transfer mode. The molten tip of the electrode periodically makes contact with the weld pool creating a short circuit. Metal is then transferred to the weld pool by surface tension, disconnecting the electrode from the weld pool, (also called free burn) and starting the process all over again. This cycle repeats many times a second. This creates a distinctive current and voltage waveform pattern, as seen on Figure 1.3.
  At stage **A**, the electrode drop makes contact and short-circuits with the weld pool, the arc voltage approaches zero and the current level starts to increase. At **B**, the electrode necks or pinches, the voltage slowly increases and the current increases towards a peak value. At **C**, the drop detaches from the electrode and the short-circuit ends, and the welding process enters the arc phase, the current reaches a peak value and the voltage almost immediately raises to a peak too. **D** is at the tail-out of the short-circuiting waveform, the electrode drop reforms. At **E**, the electrode drop once again short-circuits with the weld pool. If the current is to high at the time of the free burn, then spatter will occur, see below. Short-Circuit transfer is normally used for out of position welding, e.g. when the workpiece positioned such that the weld is vertical, and for welding of thinner ferrous metals [2].

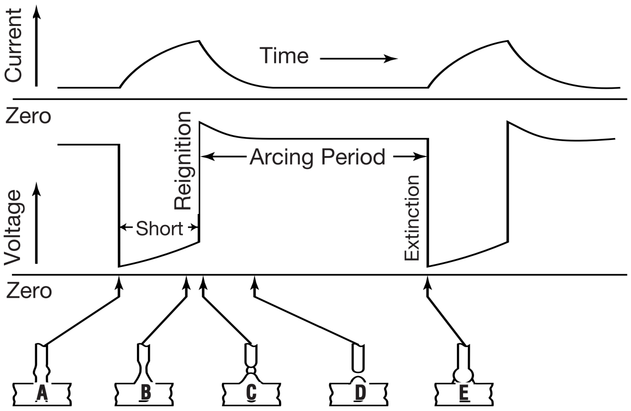


Figure 1.3: Current and voltage during the GMAW short-circuit transfer process [6].

Welding Performance Improvement

Spatter is the metal particles expelled during fusion welding which do not form a part of the weld bead [2]. Welding spatter is generated in fusion welding by unbalanced metal transfer forces, caused by an inadequate welding environment, relating to the welding current and voltage, shielding gas and the welding filler. Welding spatter is detrimental to the welding process as the spatter reduces the quality of the welded product, and necessitates inefficient cleaning processes to remove the spatter [7].

Figure 1.4 illustrates a weld bead with spatter from the weld process on a welded product.

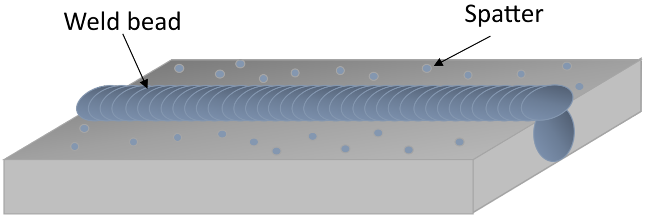
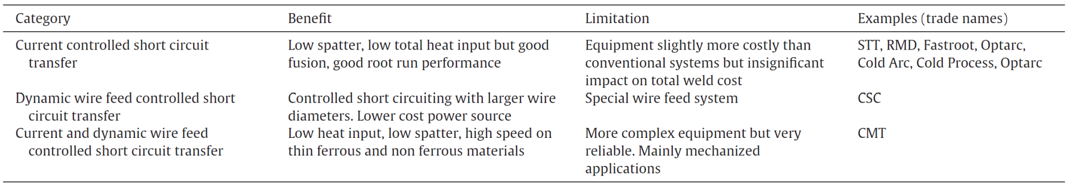


Figure 1.4: Illustration of a weld bead and spatter on a welded product

There already exist some methods for improvement of the performance of GMAW short-circuit welding, namely the Surface-Tension transfer (STT) method [8]. STT works by modulating the current so that the current is low just before a free burn. This minimizes spatter, but it also fixes the transfer rate. Similar as for the Pulse-Spray mode, the number of settings needed is also increased, thereby reducing the ease of use. Figure 1.5 categorises several industrial controlled Short-Circuit transfer methods, including STT, and makes a brief comparison of their features.



Category	Benefit	Limitation	Examples (trade names)
Current controlled short circuit transfer	Low spatter, low total heat input but good fusion, good root run performance	Equipment slightly more costly than conventional systems but insignificant impact on total weld cost	STT, RMD, Fastroot, Optarc, Cold Arc, Cold Process, Optarc
Dynamic wire feed controlled short circuit transfer	Controlled short circuiting with larger wire diameters. Lower cost power source	Special wire feed system	CSC
Current and dynamic wire feed controlled short circuit transfer	Low heat input, low spatter, high speed on thin ferrous and non ferrous materials	More complex equipment but very reliable. Mainly mechanized applications	CMT

Figure 1.5: Categories of controlled short circuit transfer and their features [9].

Migatronic A/S [1], has developed a solution for Short-Circuit transfer control called Intelligent Arc Control (IAC). In IAC, every welding cycle is registered, and the current waveform is controlled when the electrode drop shorts with the weld pool. The waveform is then designed such that the current is lowered just before the drop detaches from the electrode resulting in a rupture. Outside the short-circuit period the system is voltage controlled. IAC is laborious to set up for new electrode wire and shielding gas combination, as the system requires approximately 26 welding parameters to be finely tuned. As such, IAC is only available on a few select welding programs, see the Sigma Select Program Table in Appendix B. Furthermore IAC requires additional hardware in operation of the welding system, to enable adjusting the current fast enough as ruptures cannot be identified early enough to allow for the regulation of current time in without this extra hardware.

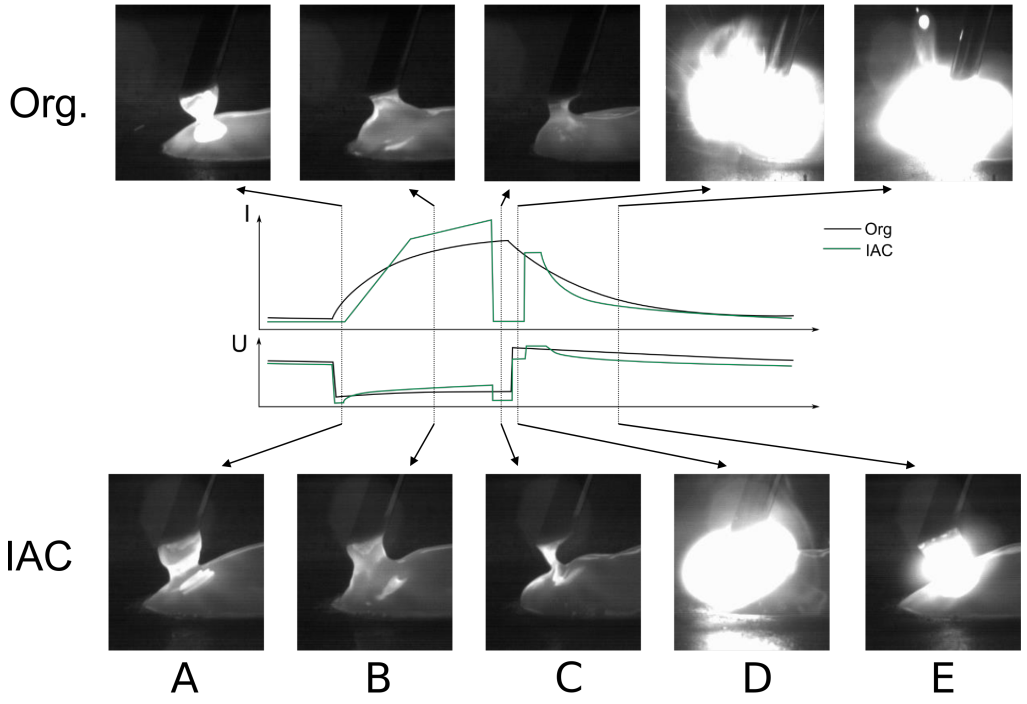


Figure 1.6: The IAC waveform design (denoted IAC) vs. a conventional Short-Circuit welding waveform (denoted Org.) [1]

Figure 1.6 shows an IAC designed current waveform, along with the corresponding voltage waveform versus the conventional Short-Circuit welding current waveform and its corresponding voltage waveform. The effect of the waveform is showcased by images from a welding process with IAC current control at various stages in the welding cycle respectively from a conventional Short-Circuit welding process welding process at corresponding welding cycle stages.

Describing the IAC process: At **A** the electrode short-circuits with the weld pool and the voltage sharply decreases while the current is allowed to increase towards a peak value higher than in conventional Short-Circuit welding. At **B** the voltage is slowly increasing, and the current increases towards a peak value higher than in conventional welding, to allow the pinch effect on the electrode drop. At **C**, after reaching the peak current, the current and voltage both are suddenly decreased to allow a cold transfer of the molten drop. At **D**, the short-circuit phase has ended and the voltage increases, however to allow a stable re-ignition of the arc the current remains low. Finally, at **E** the arc has re-ignited and the arc phase begins, the current is at first increased and then steadily reduced to a low level while a new electrode drop forms [10].

If instead the time of start of free burn is predicted, based on process current and voltage measurements, the current can be adjusted just in time to minimize spatter. This could increase the performance without reducing the ease of use and without fixing the transfer rate. Such an estimator should be robust to account for varying welding settings and working conditions.

1.2 Problem Statement

The project work presented in this thesis is made in collaboration with Migatronic A/S and it focuses on predicting the start of the free burn phases in a conventional GMAW Short-Circuit welding process. The full project proposal suggested by Migatronic A/S can be seen in Appendix A.
The extent of the collaboration is such that Migatronic A/S has provided measurement data sets for project work, see Chapter 2.

The aim of the project is to develop a system, which based on estimation and classification of 50kHz sampled current and voltage data, can predict when free burn is starting to happen, at least 5 sample-times before it happens.
From Migatronic A/S, it is specified that the free burn phase start is to be predicted atleast 5 samples before it occurs, which with a sampling frequency of 50kHz corresponds to 0.1ms. Focus is put on being able to correctly classify data samples to determine the current phase of the welding process for the individual measurement data sample, and whether free burn is going to occur. It is especially important to not misclassify the phase of the welding process, as this is detrimental to the welding process.
Features are extracted from the obtained current and voltage data, and modelled states are observed for use in classification of the system state.

Problem statement: *"How should a classification algorithm be, which, based on, features extracted from the system measurements, can reliably predict when the GMAW Short-Circuit welding process is about to free burn, in the rupture period, in the ending of the shor-circuit phase."*

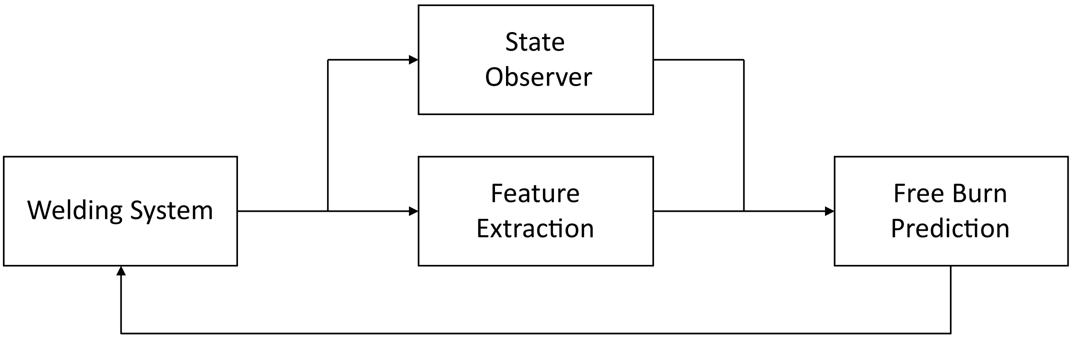


Figure 1.7: Overview of the system outlined in this project

The diagram shown in Figure 1.7 depicts the structure of a free burn estimation system proposed in the work of this thesis. From the welding system, measurements of current and voltage are passed on to the feature extraction where the data is processed and additional information is retrieved from the measurements. In parallel, the measurements are used for observing states of the welding system based on models of the welding process as additional features. The extracted features are then passed on to the free burn predictor, where the state of the welding process in predicted based on a classification model, utilising the extracted and estimated features.

The following is an outline of the work described in this thesis, relating the system diagram in Figure 1.7:

Chapter 2 presents the welding system component of the system in Figure 1.7. The welding system setup used in the work of this thesis for acquisition of the GMAW welding data is described. Models of several parts of the welding system are introduced, including the electrical system and arc, and the mechanical system, along with the dynamics of the electrode drop detachment.

Chapter 3 discusses the feature extraction part of the system in Figure 1.7. The stochastic properties of segments of interest in the time series of welding data measurement samples are examined. The classification problem for prediction of free burn is outlined in preparation for extraction of features from the welding data. A set of sample-by-sample as well as statistical features are extracted, and the relevance of these features are quantified.

Chapter 4 relates to the state observer part of the system in 1.7. State observation, in the form of a extended kalman filter in implemented, to achieve additional features for solving the classification problem is introduced.

Chapter 5 addresses the free burn prediction part of the system in Figure 1.7. Several methods of classification are introduced and the performance of these are compared for the classification problem of predicting rupture samples in the short-circuiting phase of the system.

In **Chapter 6** a conclusion is drawn on the project work and it is evaluated whether the problem stated was solved.

In **Chapter 7** the results achieved in this project are discussed. The strengths and shortcomings of the results of the project work are addressed and the approach taken to solve the task is discussed, along with suggestions for future work.

Chapter 2

Welding System Overview

The goal of the work behind this thesis is, as mentioned in section 1.2, to develop a classifier able to predict when the GMAW short circuit process is about to free burn/rupture. This prediction has to be made at least 5 measurement samples in advance, in order for the system to have time to regulate the current in order to minimise spatter. The current cannot be changed instantaneously due to system inductance, inherent in the power supply and the power cables. This chapter goes into more detail about the general welding system setup used for GMAW short circuit. This includes a closer look at both the electrical and the mechanical system, along with the data acquisition method.

2.1 Welding System Diagram and Data Acquisition

Figure 2.1 shows a simplified diagram for the GMAW system in Figure 1.2. The system can be divided into two parts: an electrical and arc system, and a mechanical system. Different parameters can be used to model each of these two components, some of which are shown on the diagram.

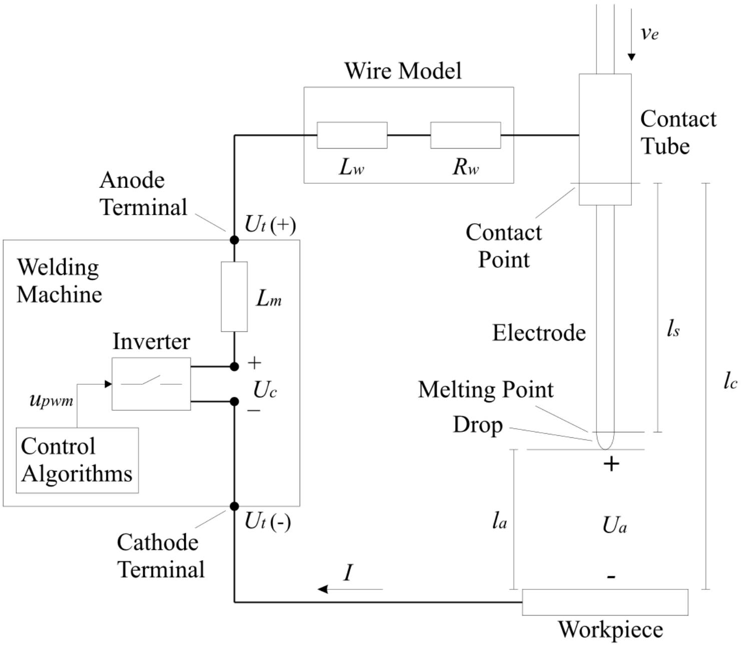


Figure 2.1: Simplified diagram for the GMAW system [3]. For legend for the parameter symbols, see Eq. (2.1)-(2.2)

As mentioned in section 1.2, the work behind this thesis is carried out in collaboration with Migatronic A/S. They have a robotic welding setup that can be used to produce consistent welds for testing purposes. The resulting measurement data obtained during the welding processes can then be used to train and/or test possible free burn predictors. To simplify the development of the initial estimator, the welding parameters are kept the same between different welds. The welding parameters here used for data generation for welding training/test can be see in Table 2.1. For a detailed overview of the different welding programs, see Appendix B.

Current	185 [A]
Voltage	18.9 [V]
Eletrode diameter	1.0 [mm]
Eletrode material	ER 316 LSi
Gas mix [ratio]	ArCO$_2$ [98/2]
Program number	202

Table 2.1: Welding settings used by the welding robot for welding training/test data generation. The Current is I and the voltage is U_c in Figure 2.1

The data measured from the welding system are voltage, current, and reference current. The reason for two current measurements, is that the power supply is inverter based and the system is current controlled, emulating an old transformer based power supply. A digital signal processor (DSP) is used to sample the data, with a sampling frequency of 50kHz. The DSP used for data acquisition also controls the welding system symbolised by the box "Control Algorithms" in Figure 2.1. Beside the current and voltage measurement data, the data accessible for the project work of this thesis also contains a flag signal, showing an estimate of when in the welding process time line the short circuits occur. This flag signal is generated by some post-processing and has the nature more of a guide line than that of an exact categorisation or evidence, and is not available on the DSP. A short example section of the given data provided can be seen in Figure 2.2 The data, and how it will be used, are discussed in further detail in Chapter 3.

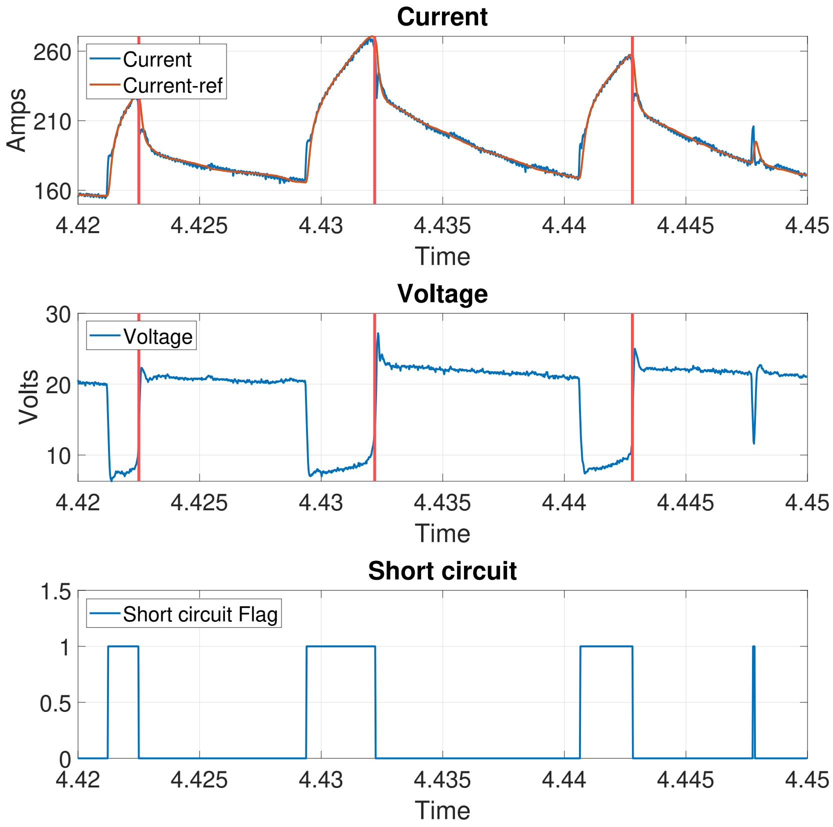


Figure 2.2: Short example section of the data provided by Migatronic A/S. Note the vertical red lines in the current and voltage data denoting when a free burn/rupture occurs.

2.2 Electrical System

The electrical system of the welding system consists, in the arcing phase of the welding process of the power supply, the power cables and the arc. In the short-circuit phases no arc exists; instead a metal bridge is present, as described in Section 2.3.1. For the arcing phases, the electrical system can be modeled as a series of resistances and inductances, with the arc as a voltage drop, described by Eq. (2.2). Eq. (2.1) below states a model for the control voltage, $U_c(t)$.

$$U_c(t) = (L_m + L_w)\dot{I}(t) + (R_m + R_w + R_e(t))I(t) + U_a(t) \tag{2.1}$$

Where, with reference to denotations to the diagram in Figure 2.1:

L_m = power supply inductance	$[H]$
L_w = power cable inductance	$[H]$
I = current	$[A]$
R_m = power supply resistance	$[\Omega]$
R_w = power cable resistance	$[\Omega]$
R_e = welding wire resistance	$[\Omega]$
U_a = arc voltage drop	$[V]$

The resistance in the system is split into 3 parts: the resistance of the power cables including the workpiece, R_w, the output resistance of the power supply R_m, and the resistance of the welding wire R_e. For the resistance of the welding wire, only the part from the contact point to tip is considered, as the current only runs though this length of the welding wire. Both R_w and R_m stays constant during welding, while R_e varies, due to changes in wire lengths and temperature. R_e also changes depending on whether the welding process is arcing or short-circuited [11]. R_w does not change during an ongoing welding process, but may change in between welds, if the lengths of the power cables changes. Similar for the power cable inductance L_w, which changes depending on the shape and position of the cable. For example if the cable is coiled up the inductance is higher than if it is not.

2.2.1 The Arc

The electrical behaviour of the arc is a non linear process, where the voltage drop over the gap between the wire tip and workpiece depends on the welding current, arc length, wire material, gas mix and several other conditions [3]. This is a very complex process, it is simplified in the modelling using a linear approximation [3],[11]:

$$U_a(t) = U_0 + R_a I(t) + E_a l_a(t) \tag{2.2}$$

Where, with reference to denotations to the diagram in Figure 2.1:

R_a = arc current coefficient	$[\Omega]$
E_a = arc length coefficient	$[\frac{V}{m}]$
l_a = arc length	$[m]$
U_0 = arc voltage constant	$[V]$

I = current $[A]$

This linear approximation only depends on arc length and current. The current is known as the system is current controlled. The arc length is harder to measure precisely, as distance from the wire tip to the workpiece varies, significantly throughout the progressing of the welding process, even dropping to zero through the short circuiting phases.

The mechanical properties of the arc will not be modeled in this thesis, as the arc phase of the welding process in not of interest in this work, it is only necessary to know if the welding process is in the arc phase or not. For further information please refer to [3] and [11].

2.3 Mechanical System

Part of the mechanical system is the drop of molten metal at the end of the welding wire. This drop forms during each arcing period of the GMAW short-circuit welding process, and is affected by a multitude of forces. While these forces are not the prime factor for metal transfer, they still affects the size and shape of the drop. A diagram of the forces acting on the drop can be seen in Figure 2.3. The directions of the arrows indicate the directions of the forces when welding on a horizontal workpiece.

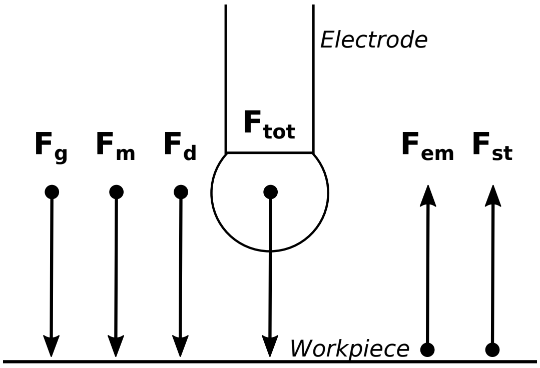


Figure 2.3: Forces acting on the molten metal drop, while welding on a horizontal workpiece. For legend for the forces, see Eq. (2.3).

The direction and magnitude of the forces, relative to the electrode, may change during the weld, due to changes in drop mass, air flow, welding gun position, etc. Eq. (2.3) shows the total force acting on the drop in vector notation.

$$\mathbf{F_{tot}} = \mathbf{F_g} + \mathbf{F_m} + \mathbf{F_d} + \mathbf{F_{em}} + \mathbf{F_{st}} \tag{2.3}$$

Where:

$\mathbf{F_{tot}}$ = total force acting on drop [N]
$\mathbf{F_g}$ = gravitational force [N]
$\mathbf{F_m}$ = momentum force [N]
$\mathbf{F_d}$ = aerodynamic force [N]
$\mathbf{F_{em}}$ = electromagnetic force [N]
$\mathbf{F_{st}}$ = surface tension [N]

The different forces and their impact on the drop may be described as follows [3]:

- **Gravitational force:** This force represents the effect of gravity on the drop. The magnitude is proportional to the mass of the drop, while the direction relative to the electrode and workpiece, depends on the orientation of the weld.
- **Momentum force:** This force is due the increase in drop mass, coming from the electrode melting. The force magnitude changes depending on the melting rate and feed speed of the electrode. The force direction follows the wire direction (away from the welding gun).
- **Aerodynamic force:** This force is caused by the drag produced by the flow of the shielding gas. While the drop radius is smaller than the electrode radius, the force magnitude is insignificant, but when the drop radius grows larger than the electrode radius, the force magnitude increases. The force direction follows the direction of the gas flow.
- **Electromagnetic force:** This force due to the electromagnetic field, created by the current flowing in the drop. If the current is increased, the force magnitude also increases. The direction of the force changes depending on the current path though the drop.
- **Surface tension:** This force is the dominant force that is keeping the drop attached to the electrode. When the magnitude of the components of the sum of all forces, $\mathbf{F_{tot}}$, in the direction opposite the direction of the surface tension, $\mathbf{F_{st}}$, the drop is detached from the electrode. However, in GMAW short circuit welding, this rarely happens, as the drop at the tip of the electrode usually makes contact with the workpiece before $\mathbf{F_{tot}}$ becomes larger than $\mathbf{F_{st}}$.

2.3.1 Metal Bridge

In the short circuiting phase of the welding process, no arc exists, instead a molten metal bridge, that forms when the molten tip of the electrode makes contact with the workpiece [11]. This is when the majority of the metal transfer takes place. An example of this bridge can be seen on Figure 1.6, where **A**, **B**, **C**, **D** and **E** shows the formation, development and collapse of the molten metal bridge, when the rupture occurs. This cycle repeats many times a second, as mentioned in Section 1.1.

Appendix C takes a closer look at the bridge, using high speed footage. Different approaches to modeling the bridge characteristics are also discussed. The conclusion is to model the metal bridge as a piece of wire with a varying cross-sectional area, leading to Eq. (2.4).

$$R = \frac{\rho l}{A} \tag{2.4}$$

Where:

R = approximate resistance of the metal bridge	$[\Omega]$
ρ = resistivity of the wire material	$[Wm]$
l = length of the wire piece	$[m]$
A = cross-sectional area of the wire	$[m^2]$

The approximate metal bridge resistance for the short circuiting phase of the welding process, be in Eq. (2.4), can be added to the electrical model, Eq. (2.1), for the welding system, expanding the model. The bridge and the arc are mutually exclusive, meaning that only one of the two can exist at a given time. So if, in the model, a voltage drop is present over the arc, no voltage drop can be over the bridge, the bridge is non-existing and vice versa. This results in two separate equations for the electrical system: one for the when the welding process is arcing, given by Eq. (2.1), and one for when the welding process is short circuiting, given by Eq. (2.5).

$$U_c(t) = (L_m + L_w)\dot{I}(t) + (R_m + R_w + R_e(t))I(t) + I(t)\frac{\rho l}{A} \tag{2.5}$$

2.4 Chapter Summation

This chapter has presented the general characteristics of the GMAW welding system and its different subsystems, looking into the functionality and dynamics of the subsystems in order to develop adequate models. Chapter 4 will look into state estimation, based on the models developed in this chapter. The next chapter will look into feature extraction based on the system measurements.

Chapter 3

Data Feature Extraction

This chapter introduces the processing of the welding measurement data sets obtained.
First the data is pre-processed and certain segments of data are discarded. The remaining data samples are then divided into classes, in preparation for use in training and testing of various classification algorithms. Secondly, additional features are extracted from the data sets, and an analysis is conducted to determine the relevance of the extracted features. Feature extraction is necessary, as the original welding data does not contain enough information to accurately predict the start of free burn of the weld short-circuit. Hence the aim of feature extraction is to obtain new feature data in which the differences between the various phases of the welding process are more prominent.

3.1 Data Pre-processing

A welding measurement data set consist of corresponding time series of samples of current and voltage, measured from the welding system, see Section 2.1. Additionally a time series of flag samples is included which provide and estimate of whether the welding process is arcing or short-circuiting. In Figure 3.1 the first 2 seconds of a typical data set obtained, as provided by Migatronic A/S, is shown. In the start of the welding data sets, the current and voltage measurements indicate the welding process has not yet begun These measurements do not belong to any phase in the welding process, and they are therefore discarded. In between the different data sets provided, the point of time varies, where the welding process actually beings. But in no data set indication is found for the data set not to be ongoing 1 second after start of the data set. Therefore, the first second of each data set is discarded, such that only data belonging to the welding process is used for training and testing of classifiers.

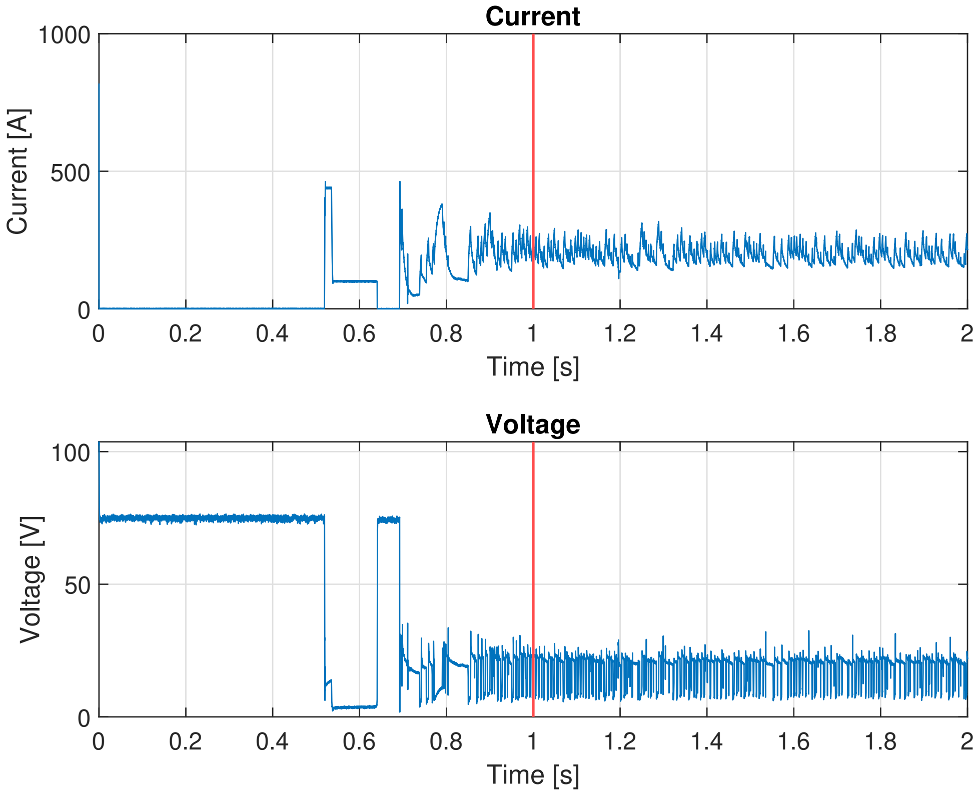


Figure 3.1: Current and voltage for the first 2 seconds of a dataset. The red line indicates the cut-off, at 1 second, data before this point of time is discarded.

Following the discarding of the first second of data, the measurement data samples of current and voltage are separated into classes, utilising the short-circuit flag data. The short-circuit flag data are only an estimate of when the short-circuitings occur, and will therefore not be used for classification purposes, but serves to help divide the data samples into classes of interest, as a starting point for this, the 4 welding data classes, introduced in this work are:

$c_1 = Arc$

$c_2 = Short$

$c_3 = False\ Short$

$c_4 = Rupture$

The set of targeted classes is then $C = \{c_1, c_2, c_3, c_4\}$. Data samples belonging to the Arc class are data sampled during the arc phases of the welding process, likewise data samples belonging to the Short class are data samples from the short-circuit phases

of the welding process, before rupture begins between the electrode and workpiece. The False Short class is for data samples belonging to short-circuit periods that are deemed to short to be a proper short-circuit period and hence are not suited to provide proper training or testing data samples, as samples from such periods are suspected to be more irregular. Lastly, the Rupture class is for data samples obtained in the last segment of a short-circuit period, where the electrode rupture more and more from the workpiece just before free burn occurs. In this manner any data samples belong to one class only.
The short-circuit Flag has a value of "0" when it is estimated that the welding process is in the arcing phase, and a value of "1" when it is estimated to be in short-circuit phase, see Figure 2.2 for and example. Based on these Flag values, the data samples are divided into classes by the algorithm below. In the algorithm *shortLength* refers to the minimum length of a short-circuit phase, for it to be deemed a proper short-circuit period, *ruptLength* refers to the length of the rupturing period, Figure 3.2 illustrates these period lengths. Unless otherwise specified, given in number of samples, $shortLength = 30$ and $ruptLength = 10$. The *shortLength* is picked to allow the welding process to properly settle into the short-circuit phase, while the *ruptLength* is picked due to the requirement of predicting the start of free burn 5 samples early, by then setting the $ruptLength = 10$ it can be investigated how early the prediction is possible. Algorithm for class-division by short-circuit Flag:

1. Find next short-circuit period in the short-circuit Flag, a data sample with Flag value of "1".

2. Determine the length of the short-circuit period, the length of the sequence of "1".

3. If the short-circuit period is longer than *shortLength*, then label the last *ruptLength* samples in the short-circuit period with class label c_4, signifying a rupture period, then label the prior samples in the short-circuit period with class labe c_2. Otherwise label samples in the short-circuit with class label c_3, signifying a false short.

4. Go back to Step 1. until no more short-circuit periods are found.

5. When no more short-circuit periods are found: Label the unlabelled samples with class label c_1, signifying arcing period.

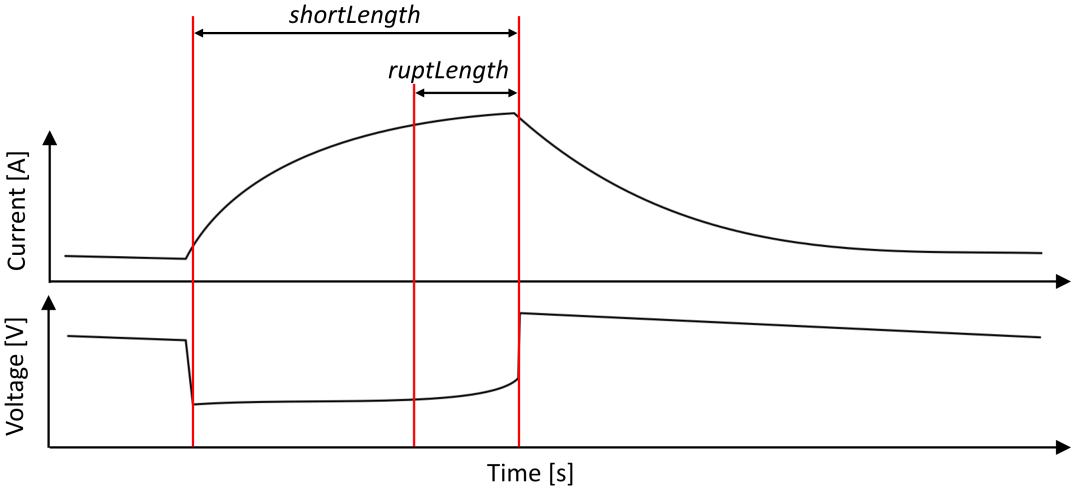


Figure 3.2: Illustration of a short-circuit period of length *shortLength*, with a rupture period of length *ruptLength*

For one data set Figure 3.3 shows the distribution of data samples of the various classes upon current and voltage measurement values, while Table 3.1 shows the proportion of each class within the data set. From Figure 3.3 it is evident that while the measurement samples of each phase of the welding process is somewhat grouped, there is a large overlap in the measurement samples meaning the data belonging the various phases are not easily separated.

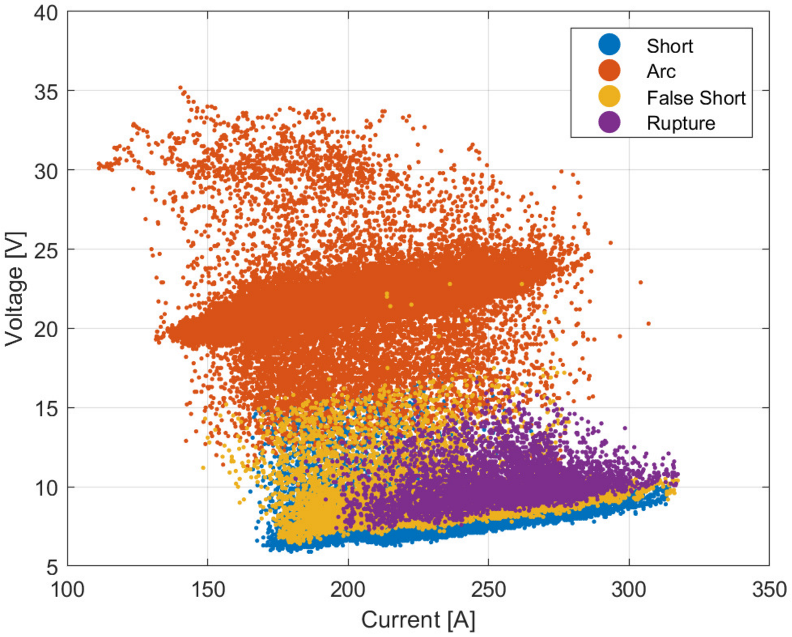


Figure 3.3: Distribution of the various classes for one data set upon current and voltage measurement values

Class	Proportion - Samples	Proportion - %
Arc	245062	81.8371
Short	45375	15.1527
False Short	4074	1.3605
Rupture	4940	1.6497

Table 3.1: Proportion of the various classes for one data set

In the project work presented, focus lies on distinguishing the *Short* class from the *Rupture* class, as there already are good means at available to determine whether the welding process has entered the short-circuit phase. The difficulty lies in being able to predict the free burn of the welding process, marking the end of the short-circuit phase, based on being able to determine the welding process having entered the rupture period within the short-circuit phase. Therefore features extracted, described in this chapter, are evaluated only in terms of their relevance to the classes *Short* and *Rupture*, and any classification of data will be concerning classification to these classes.

Following data pre-processing and division into classes, additional features for the purpose of classification, can be extracted from the data.

At first expected feasibility for extraction of features suitable as base for classification from the data samples belonging to the period when rupture is ongoing, i.e. the last part of the short-circuit phase before free burn, is analysed by looking into the stochastic properties of those data samples.

3.1.1 Stochastic Properties

Before extracting features from the data, it is worth looking into the randomness of the data leading up to a rupture to ascertain whether they exhibit kinds of patterns making it seem feasible from them to extract feature on which to base a classification. This is done by first identifying when a rupture occurs, and then looking at the samples obtained in the last time interval leading up to the rupture. For the data sets provided by Migatronic A/S, this creates more than 500 small rupture data sets for each measurement type, i.e. voltage, current. By examining at both the auto- and cross-covariance of these rupture data sets, it is possible to determine whether they contain any repeating features or patterns or whether they resemble white noise with no dominant pattern in the signal leading up to a rupture. The Matlab function *xcov()* is used to calculate both the auto- and cross-covariance.

By computing the auto-covariance for the rupture data sets and comparing it to the auto-covariance of white noise, it is possible to determine whether there is any correlation between the samples in each data set. The auto-covariance is calculated for the voltage data sets. Here rupture data sets with a length of 25 samples are used, meaning the data sets each comprises the last 25 samples before rupture/free burn occurs, ending the short-circuit period. Figure 3.4 shows the auto-covariance of 4 random data sets. By comparing the auto-covariance of the data sets, it is clear there is some correlation between the samples in the data sets, indicating there is some degree of pattern in the voltage signal leading up to a rupture. The reasoning behind comparing the stochastic properties of a period 25 samples here, compared to the *ruptLength* of 10 samples in Section 3.1, is to determine how early a pattern develops, while in the prior section the class division is done to avoid an abundance of non-rupture samples being labelled as rupture samples.

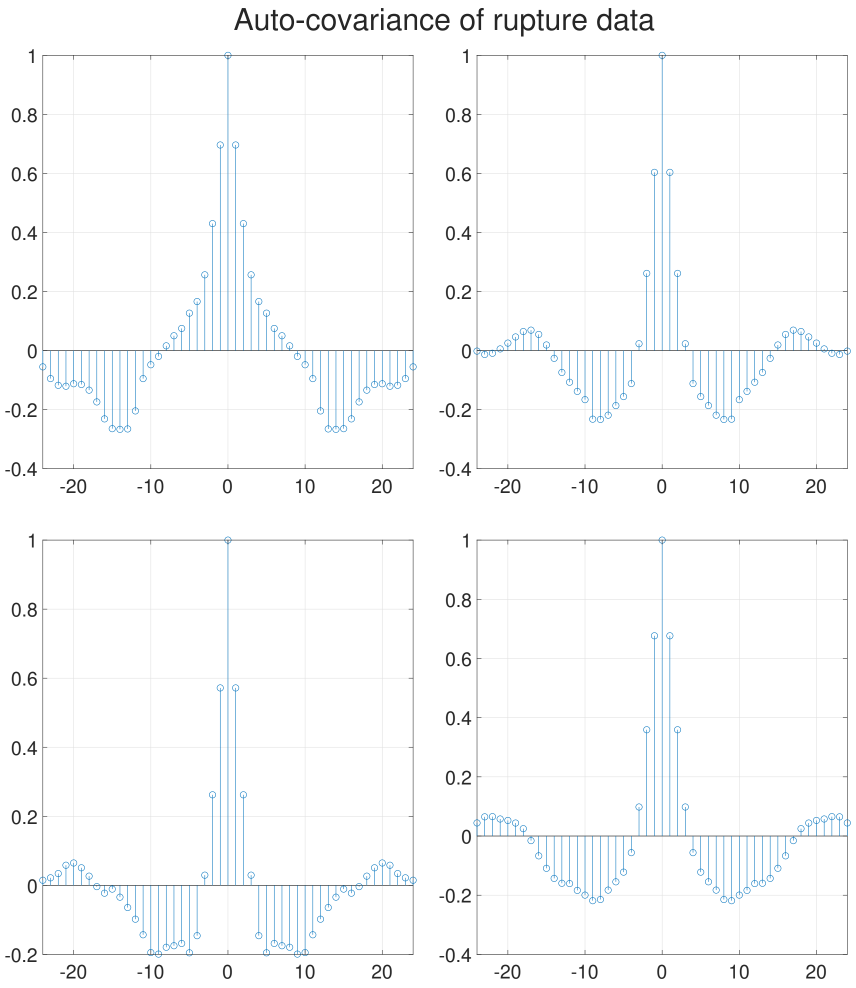


Figure 3.4: Normalised auto-covariance for 4 voltage rupture data sets from random ruptures, all with a sample lengths of 25 samples

There is however some differences between the auto-covariance of the different data sets, indicating there is some variance in the pattern of the voltage signal leading up to a rupture. One way to analyse this, is to look at the cross-covariance for pairs of the data sets.

By computing the cross-covariance for pairs of the data sets, it is possible to determine whether there is any correlation between the samples in one data and the samples in another data set. Here the cross-covariance is calculated for voltage data sets with a length of 25 samples. Figure 3.5 shows the cross-covariance for 8 random data sets, randomly paired. By comparing these with the auto-covariance, it can be seen that some correlation exits between the different sets.

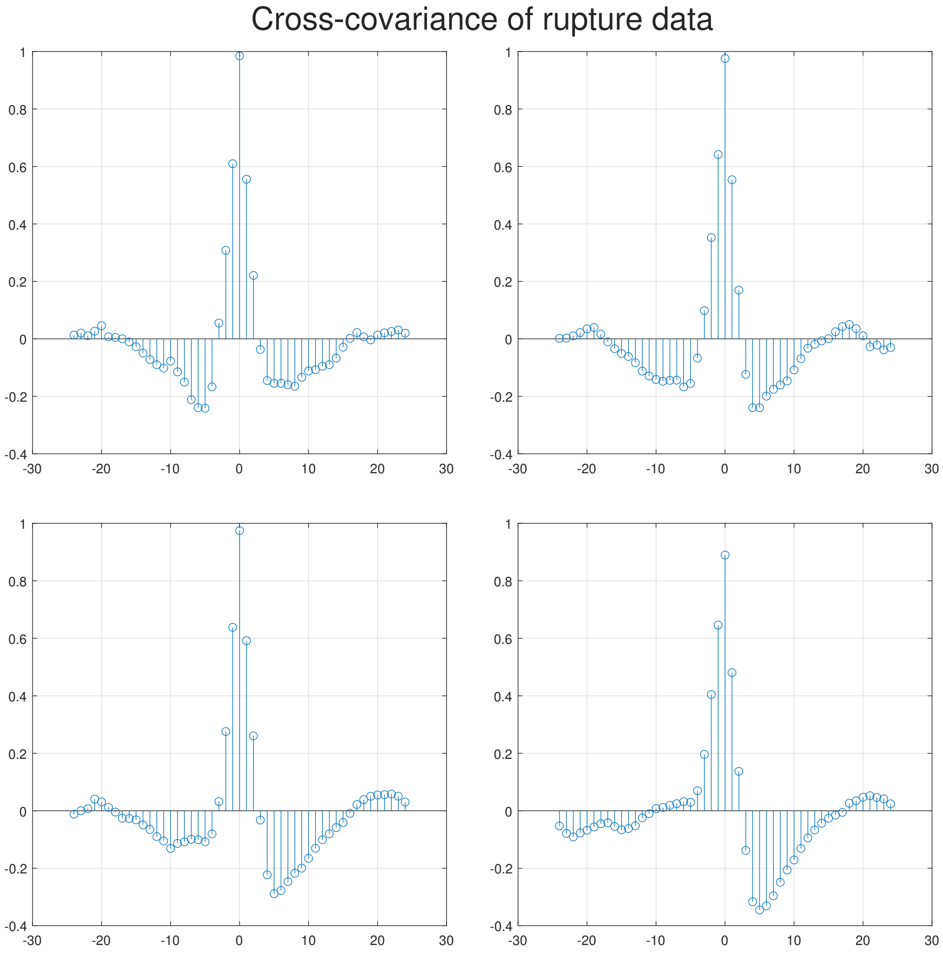


Figure 3.5: Normalised cross-covariance for random pairs of 8 voltage rupture data sets from random ruptures, all with a sample length of 25 samples.

Again, even though there is some difference between the cross-covariance of the different pairs of data sets, there is still a clear tendency, due the similarity to the auto-covariance of the rupture data sets.

Both the auto- and cross-covariance has been calculated for multiple data sets, respectively pairs of data sets, and the results all show the same tendency. The repeating tendencies indicate that the samples leading up to a rupture contain some degree of fixed or common features or patterns and are not completely random. Hence it make senses to it makes sense to search or these features by extracting a number of candidate features, and to look at the importance of these for the prediction of ruptures.

3.2 Feature Extraction

In the first phase of the feature extraction process, features that can be extracted from the current and voltage measurements sample-by-sample are investigated. The first two features are the current and voltage measurements, used following the the aforementioned pre-processing in Section 3.1. The total list of features here investigated is:

- Current, I
- Voltage, U
- $I^2 + U^2$
- Power, $P = I \cdot U$
- Voltage Time Gradient, $\frac{dU}{dt}$
- Resistance, $R = \frac{U}{I}$

The above features are all obtained by simple operations on the current and voltage measurements. Figure 3.6, shows the various features plotted for a short example section. Estimated rupture periods in the data is indicated by the red highlighting, determined by the labelling of data samples in Section 3.1. As can be seen, several of these features seem visually to be just scaled versions of the original current or voltage signals, however it is visible that some parts of the signal have been scaled differently than other parts. It is assumed, these features will be relevant to class separation and classification. An analysis is later conducted to determine the relevance of the extracted features, and whether they are each redundant in relation to other features.

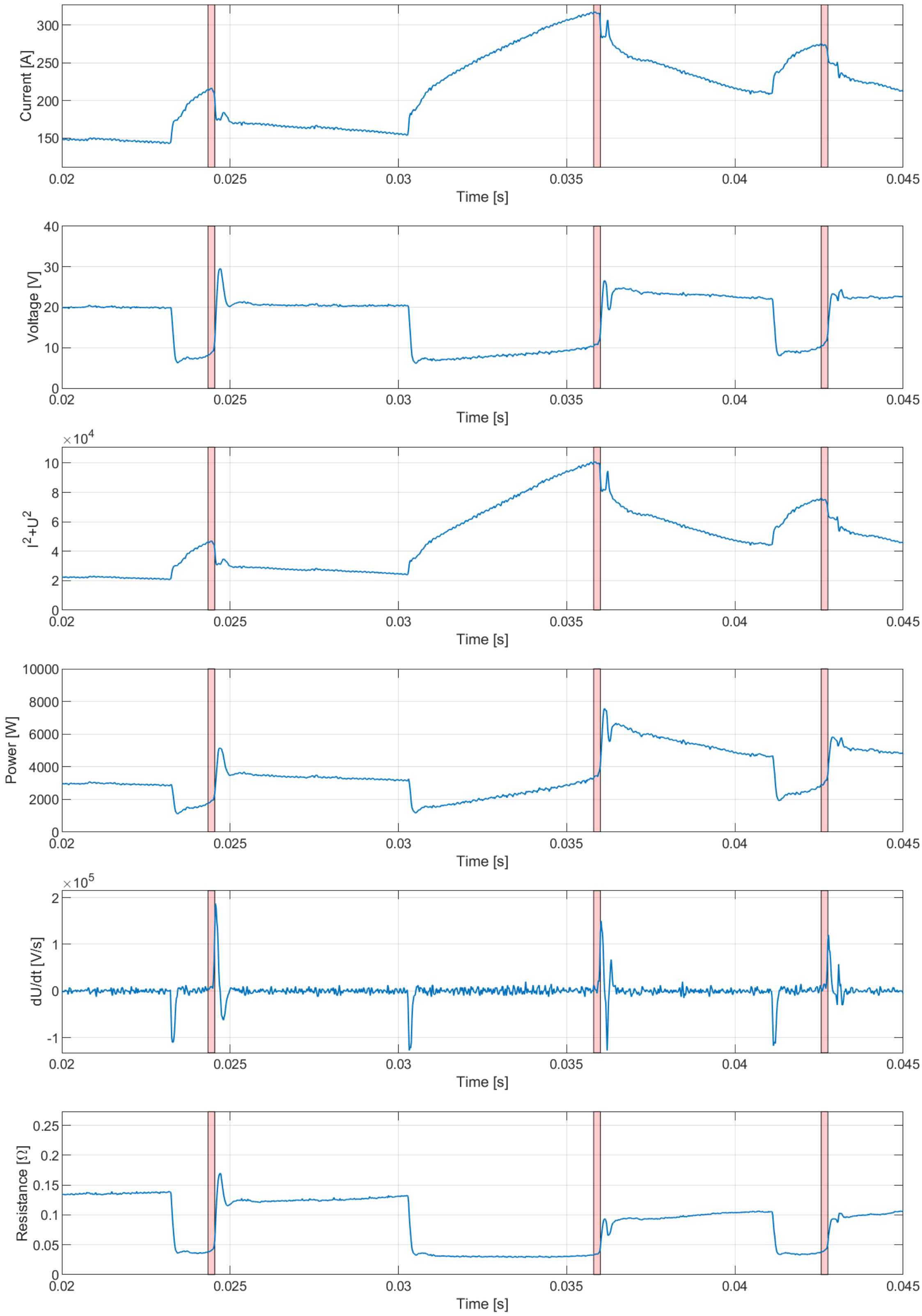


Figure 3.6: Plots of the extracted sample-by-sample features for a short example section. Rupture periods are highlighted in red

3.2.1 Statistical Feature Extraction

In addition to the in Section 3.2 aforementioned sample-by-sample features, several statistical features are also extracted from the measurement data, in particular from the voltage data: Further on in the report all discussions of statistical features in relation to welding data sets are based on voltage data In [12], 11 statistical features are extracted from an electrical signal for identification of classes of stimuli on plants. These features can be relevant to the the stimuli resulting in patterns or features in the electrical signal, although the sequences of data samples investigated are much longer. Several of these statistical features are extracted from the electrical signals, that is, the voltage measurements, in the data sets available for this project. The voltage signal is chosen as base instead of the current signal, since the signal signature in the transition from short-circuit to free burn is more abrupt. The following is a list of the extracted statistical features.

- Sample Mean
- Sample Variance
- Sample Skewness
- Sample Kurtosis
- Hjorth Parameters
    - Mobility
    - Complexity
- Interquartile Range

Contrary to the first set of sample-by-sample features extracted, these statistical features are computed by evaluating the respective measures for the features over a sliding window of samples, rather than based on one sample at a time.
E.g. the set of available voltage measurements in a data set is denoted as:

$$U = (u_1, ..., u_n)$$

By positioning the sliding window so that in the first position, for computation of the first feature value, the window includes only the first data sample, and for each new position, for each computation of a subsequent feature value, moving the window one data sample forward in time, the first iterations of the feature computation will not be based on a full window of samples. In general, the set of data samples included, here voltage measurements, included in the computation of the ith feature value is:

$$U_N = \begin{cases} (u_1, ..., u_i), & \text{if } i \leq N \\ (u_{i-(N-1)}, ..., u_i) & \text{otherwise} \end{cases} \tag{3.1}$$

The length of the window is the N samples, and the overlap between window positions as outlined in Eq. (3.1), is $N-1$ samples, such that a feature value will be computed for every measurement sample.

The following gives a short description of the various features and how they are computed. In the computation formulas Eq. (3.2)-(3.8), x_i is the ith sample in the sliding window.

Mean

The mean is the sample mean of the N samples in the window. The sample mean is computed as:

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3.2}$$

Variance

The variance is the sample variance of the N samples in the window. The sample variance is computed as:

$$\mathrm{Var}(x) = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2 \tag{3.3}$$

Skewness

The sample skewness [13], is a measure of the asymmetry of the value of the data samples around the sample mean. If the sample skewness is negative, the distribution of data values is spread out more to the left of the sample mean, than to the right. If the skewness is positive, the distribution of data values is spread out more to the right side of the sample mean , covering more values above the mean value, than to the left side. If the distribution is normal, then the skewness is 0. The unbiased sample skewness is defined as Eq. (3.4).

$$\mathrm{Skew}(x) = \frac{\sqrt{N(N-1)}}{N-2}\frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^3}{\left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2}\right)^3} \tag{3.4}$$

Kurtosis

The sample kurtosis [13], is a measure of how prone the distribution is to outliers.The sample kurtosis of a normal distribution is 3. Distributions more prone to outliers have a kurtosis larger than 3, while distribution less prone to outliers has a kurtosis less than 3. The unbiased sample kurtosis is defined as Eq. (3.5).

$$\mathrm{Kurt}(x) = \frac{N-1}{(N-2)(N-3)}((N+1)\frac{\frac{1}{N}\sum_{i=1}^{N}(x_i-\overline{x})^4}{\left(\frac{1}{N}\sum_{i=1}^{N}(x_i-\overline{x})^2\right)^2} - 3(N-1)) + 3 \tag{3.5}$$

Hjorth Parameters

The Hjorth parameters, activity, mobility, and complexity, are three parameters originally defined to describe general characteristics in an EEG trace [14], however they could be suitable as features in other signal processing tasks as well. The parameters quantify characteristics of a signal based on variance of the signal amplitude, and variance of the first and second derivative of the signal.
The activity of a signal is the variance of the amplitude, which is already included as a feature and accounted for above.
The mobility is the square root of the ratio between the variance of the first derivative and the variance of the amplitude, see Eq. (3.6). This ratio is a measure of the average curve slope.
The complexity is the ratio between the mobility of the first derivative of the signal, and the mobility of the signal itself, see Eq. (3.7). The parameter is a measure of the similarity of the signal to a pure sine wave, where the minimum value of the parameter, 1, corresponds to a pure sine wave.

$$H_{mob}(x) = \sqrt{\frac{Var(\frac{dx}{dt})}{Var(x)}} \tag{3.6}$$

$$H_{comp}(x) = \frac{H_{mob}(\frac{dx}{dt})}{H_{mob}(x)} \tag{3.7}$$

Interquartile Range

The interquartile range (IQR), is the range or difference between the 75th and 25th percentiles or upper and lower quartiles. Given a set of en even $2n$ values, or a set of an odd number of values, $2n+1$ valuues, then the first quartile Q_1 equals the median of the n smallest values and the third quartile Q_3 equals the median of the n largest values. Then the interquartile range IQR equals the difference between Q_1 and Q_3, i.e. Eq. (3.8) [15].

$$IQR = Q_3 - Q_1 \tag{3.8}$$

Finding Optimal Window Length N

It is necessary to determine the optimal length N of the sliding window, in terms of relevance to classification, for the computation of the aforementioned statistical features. In order to determine the best value for N, the relevance of each feature computed for various value of N is quantified. Additionally a simple classification model is fitted to the computed feature values, and performance measures quantifying the classification performance are computed, using a tenfold cross-validated feature set for each window length N. Section 3.2.2 outlines a method of evaluating the relevance of features to a classification problem.

3.2.2 Minimum Redundancy Maximum Relevance

To evaluate the relevance of a feature set for a classification, it is necessary to determine a method of quantifying this relevance. On such method is the Minimum Redundancy Maximum Relevance (mRMR) [16] feature selection method. The mRMR method aims to find the subset of features S, from the set of features F, that contains the features with maximum relevance for the targeted classes and the minimum redundancy with respect to each other. For two variables, x and y, the mutual information, Eq. (3.9), is based on joint probabilities, $p(x, y)$ and marginal probabilities, $p(x)$ and $p(y)$.

$$I(x,y) = \sum_{i,j} p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(x_i)p(y_j)} \tag{3.9}$$

For features and targeted classes, if a feature is strongly differently expressed for different targeted classes, the mutual information is large, while if the expression of the feature is random or uniform for the different targeted classes, the mutual information is zero. Recall that C is the set of targeted classes. The mutual information condition to maximize, to maximize relevance for all features in S is given as:

$$V_S = \frac{1}{|S|} \sum_{f_i \in S} I(C, f_i) \tag{3.10}$$

Hence, what is sought is:

$$\max V_S = \max \left(\frac{1}{|S|} \sum_{f_i \in S} I(C, f_i) \right) \tag{3.11}$$

Where $|S|$ is the number of features in S and $I(C, f_i)$ is the mutual information of the classes in C with respect to feature f_i, calculated as:

$$I(C, f_i) = \sum_{c \in C} p(c, f_i) \log \frac{p(c, f_i)}{p(c)p(f_i)} \tag{3.12}$$

The mutual information is also to used as a measure of similarity between features. Ideally the features should be mutually maximally dissimilar. That is, the mutual information between features in S should be minimised. Then, the redundancy condition for S to strive for is given as:

$$\min W_S = \min \Big(\frac{1}{|S|^2} \sum_{f_i, f_j \in S} I(f_i, f_j) \Big) \tag{3.13}$$

However, finding S using Eq. (3.11) and (3.13) directly, ideally requires an exhaustive search considering all $2^{|F|}$ feature combinations. Instead the features are ranked using the mutual information quotient (MIQ):

$$MIQ = \Big(\frac{I(C, f_i)}{\frac{1}{|S|} \sum\limits_{f_j \in S} I(f_i, f_j)} \Big) \tag{3.14}$$

MIQ expresses the ratio of the relevance of a feature f_i to the redundancy of this feature f_i to the other features f_j in S, where the numerator and denominator in Eq. (3.14), given by Eq. (3.15) and Eq. (3.16) are the relevance and redundancy conditions to optimise when adding an additional feature f_i to S.

$$\text{Relevance of feature } f_i\text{: } V_{f_i} = I(C, f_i) \tag{3.15}$$

$$\text{Redundancy of feature } f_i\text{: } W_{f_i} = \frac{1}{|S|} \sum_{f_j \in S} I(f_i, f_j) \tag{3.16}$$

The features in F are then ranked as follows [17]:

1. The feature with the highest relevance $\max_{f_i \in F} V_{f_i}$ is added to the empty feature set S.
2. Then features in S^c, where S^c is the complement of S within F, i.e. S^c consists of the features in F not in S, with non-zero relevance to C, Eq. (3.15), and zero redundancy to the features already added to S, Eq. (3.16), is found.
    - If S^c has no features with non-zero relevance and zero redundancy, go to 4.
    - Otherwise add to S the feature with highest relevance and zero redundancy, i.e. the feature with $\max_{f_i \in S^c, W_{f_i}=0} V_{f_i}$.
3. Repeat 2. till no more features with zero redundancy are in S^c.
4. Select the feature in S^c with largest MIQ value and non-zero redundancy, i.e. the feature with $\max_{f_i \in S^c} MIQ$ and add it to S.
5. Repeat 4. till no more features with non-zero relevance remains in S^c.
6. Add the remaining features in S^c to S is random order.

From the ranked set of features S, the features with largest relevance and lowest redundancy can then be selected.

In Table 3.2, the mRMR feature importance scores based on voltage data for the various features for window length $N = 4, ..., 10$ is shown, together with the average of feature scores for each value of N. The highest score for each feature is highlighted with red. For essential features, the feature importance score is defined by their relevance V_{f_i}, for features with some redundancy the scores are the MIQ values. In Table 3.2 an indication is seen that $N = 4$ provides the best feature importance scores. It is of interest to determine whether further decreasing of the window size would result in an even better feature importance score. However, the skewness and kurtosis feature need minimum 3 and 4 samples respectively for their computation. Therefore, an mRMR test is performed for $N = 2, ..., 10$ where the two aforementioned features are excluded. These features also have the lowest individual feature scores, so not a significant amount of information is lost by excluding these.

$\boldsymbol{N}$	4	5	6	7	8	9	10
$Mean$	0.1257	0.1228	0.1199	0.1175	0.0698	0.1033	0.1098
Var	0.0563	0.0221	0.0240	0.0204	0.0884	0.0578	0.0798
$Skew$	0.0232	0.0068	0.0051	0.0335	0.0084	0.0121	0.0182
$Kurt$	0.0060	0.0038	0.0007	0.0014	0.0011	0.0000	0.0013
Mob	0.0696	0.0354	0.0290	0.0260	0.0306	0.0415	0.0431
$Comp$	0.0863	0.0873	0.0854	0.0866	0.0394	0.0800	0.0843
IQR	0.0591	0.0436	0.0335	0.0259	0.0290	0.0408	0.0563
Average	0.0609	0.0460	0.0425	0.0445	0.0381	0.0479	0.0561

Table 3.2: mRMR feature importance scores for various window lengths N. Highest scores highlighted with red

The results of the second mRMR test is shown in Table 3.3. The scores indicate that with a window length $N = 2$, the features in general have the highest relevance, closely followed by $N = 3$. It is noteworthy that Var has a higher score for $N = 10$ in both Table 3.2 and 3.3, likewise for $Comp$ in Table 3.3. It should be noted that the skewness feature could have been included at $N = 3$. However, to properly determine the skewness and kurtosis, a larger sample size than what is of interest here is necessary. Therefore these two features are excluded for classification in work further on. It is important to note, that when the statistical features are considered together with the the sample-by-sample features, the feature importance scores may differ, as the latter features may have higher relevance and imply an increase in the redundancy of the statistical features to the then combined set of features.

N	2	3	4	5	6	7	8	9	10
$Mean$	0.1317	0.1122	0.1257	0.1228	0.1199	0.1175	0.0690	0.1033	0.1098
Var	0.0758	0.0429	0.0546	0.0216	0.0237	0.0267	0.0884	0.0571	0.0787
Mob	0.0858	0.0505	0.0675	0.0345	0.0287	0.0317	0.0303	0.0411	0.0425
$Comp$	0.0744	0.0544	0.0837	0.0851	0.0844	0.0850	0.0389	0.0791	0.0832
IQR	0.0721	0.0783	0.0573	0.0425	0.0331	0.0329	0.0287	0.0404	0.0556
Average	0.0880	0.0676	0.0778	0.0613	0.0580	0.0588	0.0511	0.0642	0.0740

Table 3.3: mRMR feature importance scores for various window lengths N. High scores highlighted with red

3.2.3 Naive Bayes Classifier

In addition to the feature importance scores from the mRMR algorithm, the effect of the window length N is also evaluated for classification based only on the statistical features for window lengths $N = 2, ..., 10$.

For classification the Naive Bayes classifier is employed. The classifier is "naive" as given a class c_k, the features $f_1, ..., f_P$ are assumed independent [18]. Generally this assumption does not hold, and certainly not in the case of the derived statistical features, however the classifier still in many cases performs well, often outperforming more sophisticated methods [18].

The Naive Bayes algorithm assigns a new observation to a class by estimating the posterior probability for each class, and then assign the observation to the most probable class, i.e. applying a *maximum a posteriori* decision rule. The Naive Bayes classifier algorithm is as follows [19]:

1. Estimate the probability density of each feature for each class

2. Model the posterior probability for a new observation to belong to a given class according to Bayes rule Eq. (3.17):

$$\hat{P}(Y = k|X_1, ..., X_P) = \frac{P(Y = k)\prod_{j=1}^{P} P(X_j|Y = k)}{\sum_{k=1}^{K} P(Y = k)\prod_{j=1}^{P} P(X_j|Y = k)} \tag{3.17}$$

Where:

- $k = 1, ..., K$ are the class indices for the classes available $c_1, ..., c_K$.
- Y is the random variable corresponding to the class of an observation, i.e. one class amongst $c_1, ..., c_K$.
- $X_1, ..., X_P$ is a random observation of the features $f_1, ..., f_P$.

- $P(Y = k)$ is probability for class k.
- $P(X_j|Y = k)$ is the probability for feature f_j having the observed value X_j given the class is c_k.

3. Classify the observation by estimation of the posterior probability for each class, calculated using Eq. (3.17), then assign the observation to the class with the maximum posterior probability.

For a Gaussian Naive Bayes, where the probability densities of the features are estimated as Gaussian distributions, the probability $P(X_j|Y = k)$ is calculated as Eq. (3.18)[20].

$$P(X_j|Y = c_k) = \frac{1}{\sqrt{2\pi\sigma_{jk}^2}} \exp\left(-\frac{\left(X_j - \mu_{jk}\right)^2}{2\sigma_{jk}^2}\right) \tag{3.18}$$

Where:

- μ_{jk} is the mean of feature f_j in class c_k.
- σ_{jk}^2 is the variance of feature f_j in class c_k

Such as Gaussian Naive Bayes classifier is employed for this classification problem. To use the classifier to assign classes to new observations, training data is necessary, to estimate the densities the feature/class combinations.

K-fold Cross-Validation

When evaluating the performance of a classification model, ideally one data set, which explains the different phenomena of the data distribution well, is used for training the model, and another set is used for estimating the error rate of the model. By applying the cross-validation method, the classification model can be trained and validated on the the same data set, by using different data partitions for training and validation, and then performing several iterations of training and validating, such that in the end the model has been validated on the entire data set.

One partition scheme for cross-validation is K-fold cross-validation [18]. In K-fold cross-validation, the data is partitioned into K partitions, then for the kth iteration, the model is validated on the kth partition, while having been trained on the other $K - 1$ partitions. Figure 3.7, illustrates the data partitioning and cross-validation iterations of a five-fold cross-validation scheme.

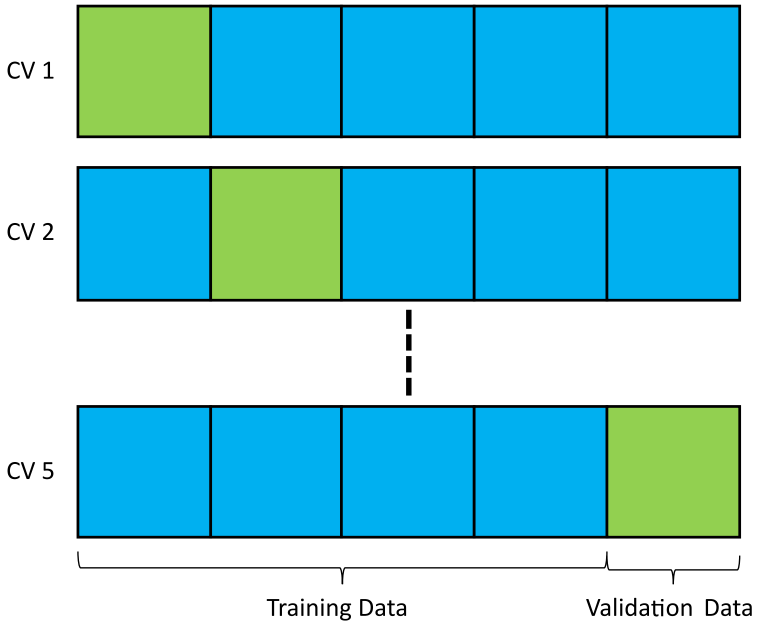


Figure 3.7: Illustration of five-fold cross-validation of a data set. Blue partitions are training data, green partitions are validation data. CV k signifies the kth iteration of the cross-validation.

For the evaluation of the Naive Bayes classifier described above a ten-fold cross-validation scheme is employed, as it is recommended as a good compromise between model bias and variance [18]. Furthermore, the cross-validation partitioning is stratified, such that the proportion of each class in each partition is roughly equal. After cross-validation, an estimate of the performance of the classifier on the entire data set is obtained.

Classification Performance Measures

To quantify the performance of the classification model, the performance measures, sensitivity, specificity, precision and accuracy, are computed [21]. In this case, the performance measures are evaluated for the binary classification problem with the "Short" and "Rupture" classes, where the "Rupture" class is considered the positive class and "Short" is considered the negative class.

Sensitivity or True Positive Rate (TPR), is the proportion of samples belonging to the positive class that was correctly identified as such. Values close to 1 is better.

Specificity or True Negative Rate (TNR), is the proportion of samples not belonging to the positive class, that was correctly identified as such. Values close to 1 is better.

Precision or Positive Predictive Value (PPV), is the proportion of samples correctly identified to belong to the positive class, out of all samples identified to belong to that class. Values close to 1 is better.

Accuracy (ACC), is the proportion of samples that correctly identified to belong to the positive class plus samples correctly identified to not belong to the positive class, out of all samples in the population. Values close to 1 is better.

The performance measures are calculated as follows:

$$TPR = \frac{TP}{TP + FN} \tag{3.19}$$

$$TNR = \frac{TN}{FP + TN} \tag{3.20}$$

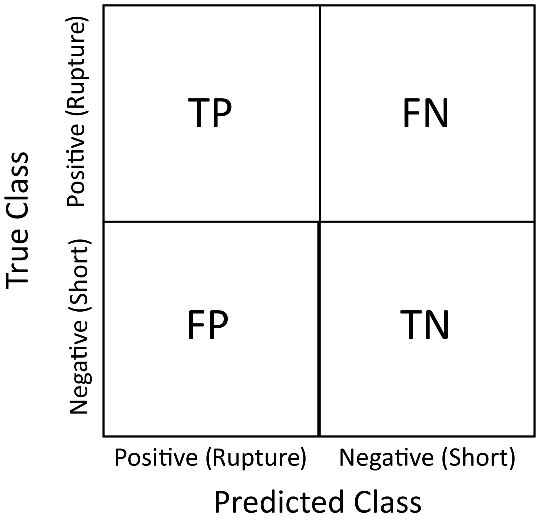
$$PPV = \frac{TP}{TP + FP} \tag{3.21}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.22}$$

Where:

- TP are True Positives, samples correctly identified to belong to the positive class.
- TN are True Negatives, samples correctly identified to not belong to the positive class.
- FP are False Positives, samples incorrectly identified to belong to the positive class.
- FN are False Negatives, samples incorrectly identified to not belong to the positive class.

An example confusion matrix for the binary classification system, with the "Rupture" class considered the positive class is shown in Figure 3.8.



		Predicted Class	
		Positive (Rupture)	Negative (Short)
True Class	Positive (Rupture)	TP	FN
	Negative (Short)	FP	TN

Figure 3.8: Example confusion matrix for binary classification problem with positive class, "Rupture", and negative class, "Short".

The performance measures for the binary classification problem, with decisions "Rupture", or "not-Rupture (Short)" are shown in Table 3.4, red highlighted values are better. It is apparent that a trade-off between measures have to be made. Two measures, PPV and ACC have maximum values for $N = 3$, one measure at, TNR, has maximum at $N = 2$, and only a little less for $N = 3$, while the measure TPR has maximum value for $N = 10$ and nearly one third less for $N = 3$. Since for all measures higher values are better, an average of the performance measures for each N is computed, higher averages being better. From the average measure, $N = 3$ performs better by a significant margin to the other values of N, and is therefore chosen as the best performing window length for the Naive Bayes classifier test, supported also by the PPV and ACC, measures, and to a major degree the TNR measure, as remarked above.

N	2	3	4	5	6	7	8	9	10
TPR	0.2130	0.6573	0.8231	0.8755	0.8994	0.9107	0.9229	0.9300	0.9358
TNR	0.9683	0.9395	0.7399	0.5471	0.4163	0.3427	0.3029	0.2826	0.2707
PPV	0.4225	0.5420	0.2562	0.1739	0.1436	0.1311	0.1260	0.1237	0.1226
ACC	0.8941	0.9118	0.7481	0.5794	0.46377	0.3985	0.3638	0.3462	0.3360
Average	0.6245	0.7627	0.6418	0.5440	0.4808	0.4458	0.4289	0.4206	0.4163

Table 3.4: Naive Bayes classification performance measures for various window lengths N. Highest scores are highlighted in red.

Recall,in the the mRMR importance test, Section 3.2.2, the best performing window length was $N = 2$, closely followed by $N = 3$. Since, in the Naive Bayes classifier

test, the performance is significantly better at $N = 3$, then even though the mRMR test indicates higher average score for $N = 2$, $N = 3$ is chosen as window length for the work further on.
From both the mRMR importance test, and the Naive Bayes classifier test, it is concluded that the window length $N = 3$ has the best performance. Examples of the extracted statistical features with $N = 3$ is shown in Fig. 3.9.

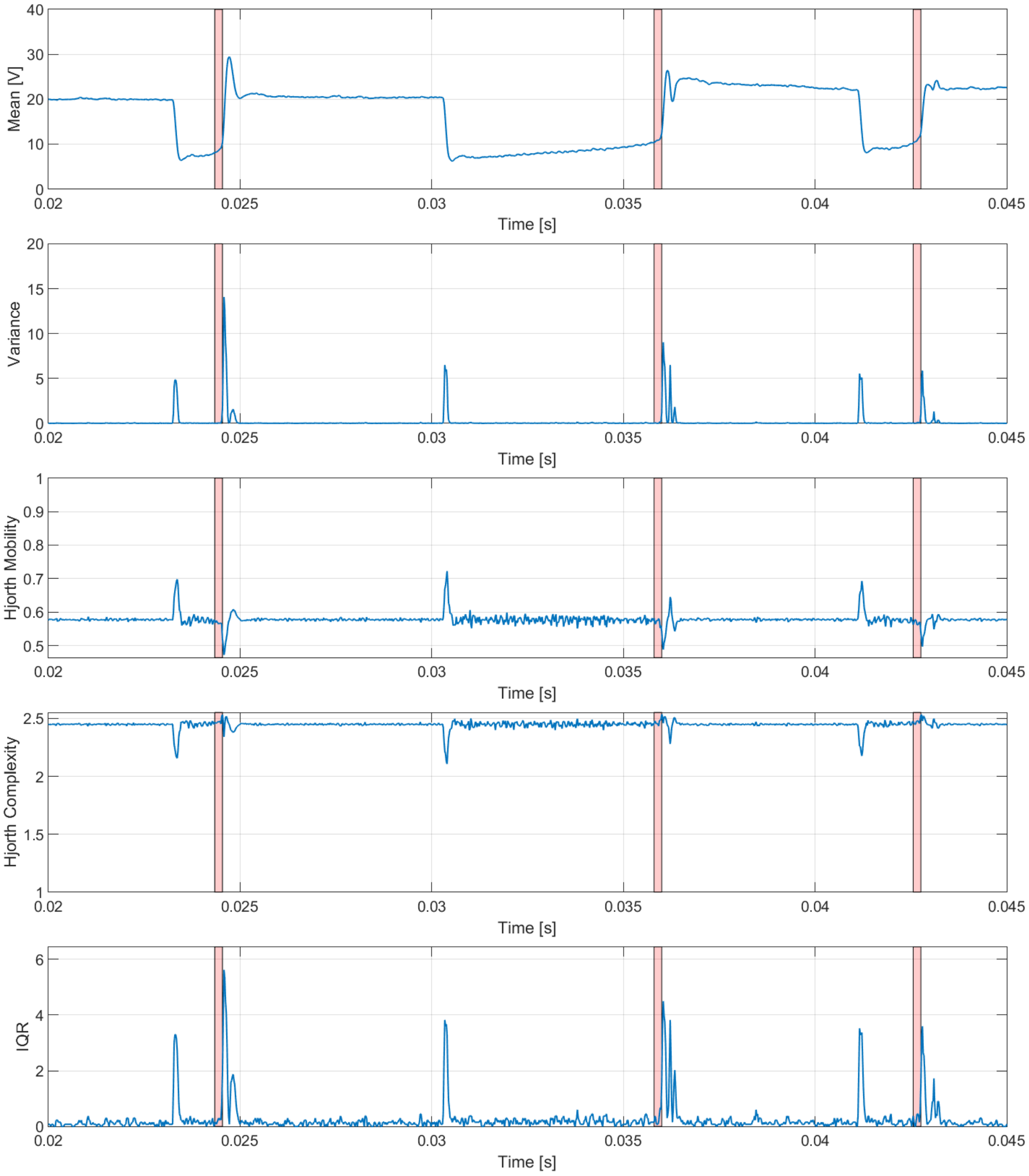


Figure 3.9: Plots of the extracted statistical features for a short example section. Rupture periods estimated in data preprocessing, Section 3.1, of length 10 samples are highlighted with red

3.3 Rupture Period Ensemble Correlation

A somewhat different approach to the estimation of rupture periods in the welding data signals is to apply a method based on rupture period ensemble correlation: By

creating a template rupture period signal, rupture periods can be identified in the welding process data by finding periods in the signal with high correlation to the template rupture period signal [22]. Intuitively, it makes sense to create the template rupture period signal, using the rupture period data from the actual welding data, based on the pre-segmentation of this into rupture periods and others obtained in the data pre-processing Section 3.1.
Three approaches are taken to create the template signal.

1. Ensemble mean of the rupture period.
2. Ensemble median of the rupture period.
3. Ensemble quartile mean: discarding data below the first quartile and above the the third quartile, to eliminate outlier samples in calculation of the mean values.

Recall, all rupture periods found in the data pre-processing are of the same fixed length of 10 samples. An ensemble here is the set of all data samples of same location in time within the rupture period to which they belong, respectively. E.g. one ensemble is the set of all samples being the first sample in a rupture period. An ensemble mean sample value is then the mean value of the data values of all samples in a specific ensemble. And the ensemble mean of the rupture period is the signal synthesised by the sequence of ensemble mean values, in the order of the data samples on which they are based. Similar for ensemble median and ensemble quartile mean of the rupture period.
The resulting template signals are shown in Figure 3.10.

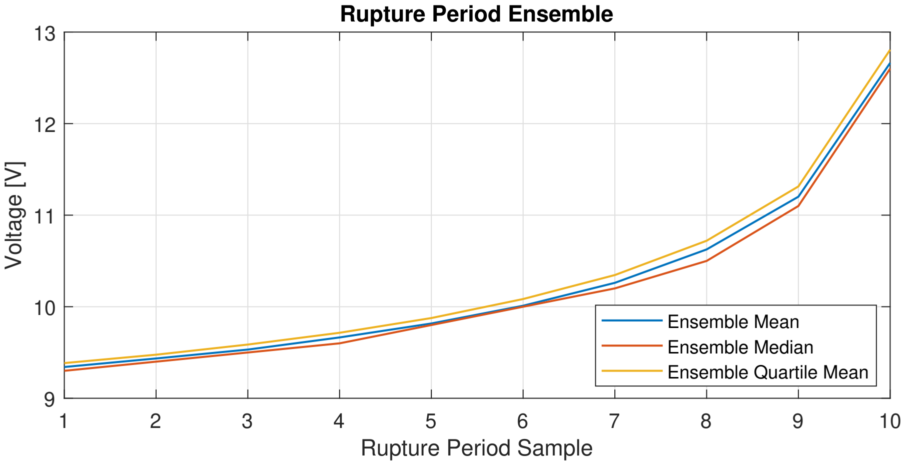


Figure 3.10: Rupture period ensemble mean, ensemble median, and ensemble quartile mean.

To determine which of the three approaches results in the best performing template, and whether the entire template signal should be employed, the mRMR algorithm,

see Section 3.2.2, is used to quantify the performance of the correlation coefficient between the welding process data and the template signal of each approach, for a range of sub-parts of the template signal. In the mRMR algorithm the correlation coefficient of the three approaches are evaluated against each other for calculation of redundancy. The resulting mRMR feature importance scores are shown in Table 3.5. The samples parameter indicate the interval of samples being used from the template signal. The number of samples directly influence the length of the template signal, and therefore also the corresponding length of the welding process data window being investigated, i.e. correlation with the template.
The results show the best performing template is the ensemble quartile mean of the rupture period, where the entire length of the template signal is used. This correlation coefficient is then added to the feature set for classification.

Samples	1-5	1-6	1-7	1-8	1-9	1-10	2-10	3-10	4-10	5-10	6-10
Mean	0.0608	0.670	0.0541	0.1197	0.0863	0.0722	0.0677	0.0618	0.0571	0.0523	0.0523
Median	0.0928	0.1009	0.0724	0.0792	0.1295	0.0908	0.0860	0.0806	0.0744	0.0674	0.0610
Quartile	0.0609	0.0663	0.1093	0.0534	0.0854	0.1363	0.1292	0.1210	0.1118	0.1017	0.0924

Table 3.5: mRMR feature importance scores for a range of sub-parts of the rupture period ensemble signal. The highest score is highlighted in red

An example of the correlation coefficient signal for the best performing template, ensemble quartile mean of full length, computed by windowing the original welding data, and computing the correlation with the template signal, is shown in Figure 3.11.

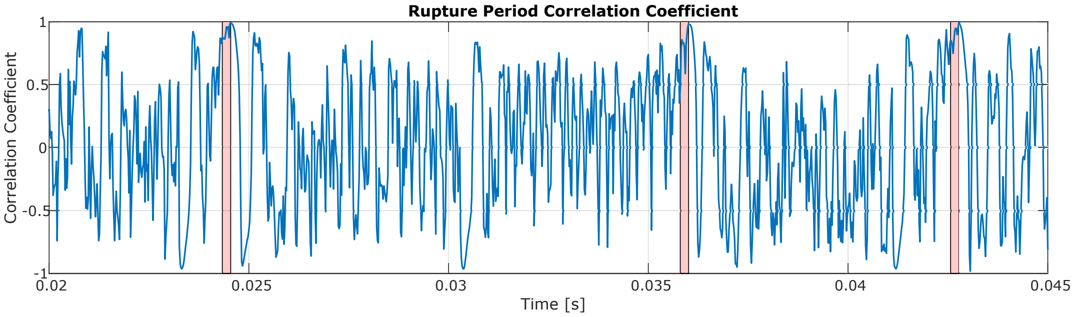


Figure 3.11: Plot of the rupture correlation coefficient for the ensemble quartile mean template of full length for an example section of a voltage data set. Rupture periods are highlighted in red.

3.4 Feature Relevance and Redundancy

From the feature extraction process, the 12 features for further consideration are as follows:

$f_1 =$ Current, I

$f_2 =$ Voltage, U

$f_3 = I^2 + U^2$

$f_4 =$ Power, P

$f_5 =$ Voltage Time Gradient, $\frac{dU}{dt}$

$f_6 =$ Resistance, R

$f_7 =$ Sample Mean of U

$f_8 =$ Sample Variance of U

$f_9 =$ Hjorth Mobility of U

$f_{10} =$ Hjorth Complexity of U

$f_{11} =$ Interquartile Range, IQR of U

$f_{12} =$ Rupture Correlation of U

The set of features for classification is then $F = \{f_1, f_2, ..., f_{12}\}$.
This section aims to analyse the various features and quantify the relevance of the features before used for classification.

3.4.1 Sample Correlation Coefficient Matrix

For all features, a sample correlation test is carried out to investigate the interdependence of the various features. The test is conducted by calculation of for all pairs of features the Pearson correlation coefficient; for two features x and y given as r_{xy} in Eq. (3.23) [23]:.

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{3.23}$$

Where:

- $x_i =$ the value of feature x for data sample i.
- $\overline{x} =$ the mean value of feature x for all data samples.

From the calculated coefficients the correlation coefficient matrix is constructed for the set of features. This a matrix with the correlation coefficients for each pairwise feature combination, as exemplified in Eq. (3.24) for a feature set of of two features:

$$R = \begin{bmatrix} r_{xx} & r_{xy} \\ r_{yx} & r_{yy} \end{bmatrix} \tag{3.24}$$

For the features computed from one data set, the correlation coefficient matrix is shown in Table 3.6. The correlation coefficient is in the range of ±1, where 1 is total positive linear correlation, 0 is no linear correlation, and -1 is total negative linear correlation. As the order of the features in the Pearson correlation coefficient Eq. (3.23 is irrelevant, that is $r_{f_1,f_2} = r_{f_2,f_1}$, the matrix symmetric, and only the upper triangular matrix is shown.

It is notable from Table 3.6, that multiple features are highly correlated, especially f_1 and f_2 being highly correlated with several of the non-statistical features. This is expected, as it was previously concluded that several features are to some degree, scaled features of the original data. It also expected that f_2 and f_7 are positively correlated, since f_7 is mainly f_2 after moving average filtering.
It cannot be determined solely based on the sample correlation test, which features should be selected for classification purposes, as the correlation shows only the redundancy of the one feature in comparison to another. It is also necessary to evaluate the relevance of each feature, to the targeted classes.

Features	f_1	f_2	f_3	f_4	f_5	f_6
$f_1 = I$	1.00	0.2910	0.9975	0.7548	0.2944	-0.1439
$f_2 = U$	$\star$	1.00	0.3052	0.8412	-0.1631	0.7411
$f_3 = I^2 + U^2$	$\star$	$\star$	-0.3950	0.2641	-0.4257	1.00
$f_4 = P$	$\star$	$\star$	$\star$	1.00	0.0932	0.2641
$f_5 = \frac{dU}{dt}$	$\star$	$\star$	$\star$	$\star$	1.00	-0.4257
$f_6 = R$	$\star$	$\star$	$\star$	$\star$	$\star$	1.00
$f_7 = Mean$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$
$f_8 = Var$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$
$f_9 = Mob$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$
$f_{10} = Comp$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$
$f_{11} = IQR$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$
$f_{12} = Rupt.Corr.$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$

Features	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}
$f_1 = I$	0.1495	-0.3123	-0.3659	0.3649	-0.3260	0.4398
$f_2 = U$	0.9545	0.5530	0.1747	-0.2101	0.5712	0.1678
$f_3 = I^2 + U^2$	0.1677	-0.2930	-0.3459	0.3448	-0.3051	0.4287
$f_4 = P$	0.7209	0.1763	-0.1082	0.0852	0.1895	0.3764
$f_5 = \frac{dU}{dt}$	-0.4274	-0.7116	-0.7829	0.8274	-0.6218	0.4825
$f_6 = R$	0.8152	0.7810	0.4681	-0.5024	0.7977	-0.1830
$f_7 = Mean$	1.00	0.7439	0.4389	-0.4793	0.7416	-0.0299
$f_8 = Var$	$\star$	1.00	0.7825	-0.8230	0.9687	-0.4088
$f_9 = Mob$	$\star$	$\star$	1.00	-0.9657	0.7660	-0.6115
$f_{10} = Comp$	$\star$	$\star$	$\star$	1.00	-0.8033	0.5889
$f_{11} = IQR$	$\star$	$\star$	$\star$	$\star$	1.00	-0.3995
$f_{12} = Rupt.Corr.$	$\star$	$\star$	$\star$	$\star$	$\star$	1.00

Table 3.6: Feature sample correlation coefficients matrix

To quantify the relevance of the feature set to the targeted classes, the mRMR algorithm, see Section 3.2.2, is once again employed. However this time with the full feature set, that is $S = F$. The ranked features in S are shown in Fig. 3.12 along with the resulting feature importance scores. A large score value indicates the corresponding feature is important, furthermore, a large drop in score value from feature f_i to feature f_j indicates confidence in the importance of feature f_i.

It is evident that f_{12}, *Rupt.Corr.*, is the most important feature, followed by the features f_4, P, and f_5, $\frac{dU}{dt}$. Though the remaining features do not seem of great importance from the mRMR results in Figure 3.12, it is in Section 5.1.5 investigated whether discarding or keeping them has an effect on classification.

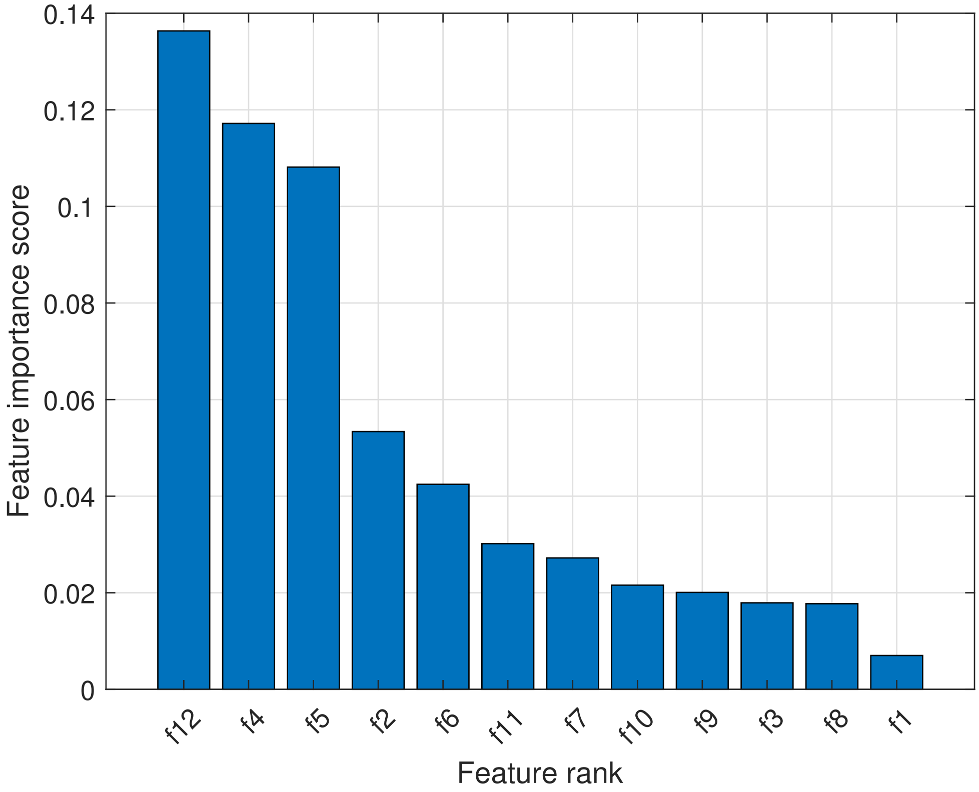


Figure 3.12: Feature ranking based on mRMR

3.5 Chapter Summation

In this chapter, a set of 12 features are extracted from the original welding data, including the current and voltage measurements. Included in this set of features are 7 sample-by-sample features, i.e. features computed sample-by-sample from the original data. Additionally 5 statistical features are extracted. These statistical features are computed from a window of samples from the original voltage measurements. For these statistical features, it is found that the optimal window length was $N = 3$ samples. A template rupture period voltage signal is created from ensemble mean values, and the correlation between the template signal and the welding process voltage data is computed and added as a feature. The correlation, relevance and redundancy of the complete feature set is examined, and it is concluded that all extracted features could be of some significance in classification of samples in the short-circuiting phase of the welding process into classes "Rupture" respectively "Short", i.e. "not-Rupture". The following chapter is exploring the possibility of adding features by state observation.

Chapter 4

System State Observation

This chapter looks into increasing the number of features available for classification by estimation of system parameters and states, while also incorporating the current measurements in the estimations. The system parameters and states are estimated with a kalman filter, based on models presented in Chapter 2.

4.1 State Space Representation

Migatronics already uses a kalman filter to estimate system resistance and inductance, based on on a simplified version of Eq. (2.1). The purpose of the kalman filter employed in the work here presented is to estimate "hidden" system states, like the cross-sectional area of the metal bridge neck, apparent in the short circuiting phase. In order to use a kalman filter to estimate these parameters, a complete state space model is set up.

Chapter 2 presents continuous time models for the different parts of the welding system. One of these models is the electrical model for the welding system in short circuit mode and was given by Eq. (2.5).

$$U_c(t) = RI(t) + L\dot{I}(t) + I(t)\frac{\rho l}{A(t)} \tag{4.1}$$

Where:

U_c = control voltage	$[V]$
I = current	$[A]$
$\dot{I}$ = time derivative of current	$[\frac{A}{s}]$
$R = R_m + R_w + R_e$ = combined resistance	$[\Omega]$
$L = L_m + L_w$ = combined inductance	$[H]$
A = cross-sectional area of the neck of the metal bridge	$[m^2]$
ρ = resistivity of the eletrode material	$[Wm]$

l = Distance from eletrode tip to workpice (length of the metal bridge $[m]$

The cross-sectional area of the neck, $A(t)$ is an unknown system state, so it has to be estimated by the kalman filter. For this $A(t)$ is added as a seperate state equation. In Appendix C it is discovered that the pinching of the neck does not happen at a constant rate. One way to model this is as a negative integral, multiplied with constant, as in Eq. (4.2). The bridge length l might not be a true constant, but as it's variations are quite small, it will be considered a constant.

$$A(t) = -c\int_0^t I(t)dt \Rightarrow \dot{A}(t) = \frac{d}{dt}\left(-c\int_0^t I(t)dt\right) = -cI(t) \tag{4.2}$$

To make sure that the system is propper, the pole/zero placement is checked. This is done by finding the transfer function for the system. The first system state equation, $\dot{U}_c(t)$ has two inputs: $I(t)$ and $A(t)$, while the second system state equation, $\dot{A}(t)$ only has $I(t)$ as input. However, as Eq. (4.1) is non-linear, it has to be linearized before it can be Laplace transformed.

$$U_c(t) = RI(t) + L\dot{I}(t) + I(t)\frac{\rho l}{A(t)} \approx \hat{U}_c(t) = R\hat{I}(t) + L\dot{\hat{I}}(t) + \frac{\rho l}{\bar{A}}\hat{I}(t) - \frac{\rho l\bar{I}}{\bar{A}^2}\hat{A}(t) \tag{4.3}$$

Eq. (4.3) shows the small signal approximation. Similar is done to Eq. (4.2), but as it is already linear, $A(t)$ and $I(t)$ is simply replaced with $\hat{A}(t)$ and $\hat{I}(t)$. Now that the equations are linearized, they can be Laplace transformed:

$$\begin{aligned}
&\mathcal{L}\left\{\hat{U}_c(t) = R\hat{I}(t) + L\dot{\hat{I}}(t) + \frac{\rho l}{\bar{A}}\hat{I}(t) - \frac{\rho l\bar{I}}{\bar{A}^2}\hat{A}(t)\right\} \Rightarrow \\
&\hat{U}_c(s) = R\hat{I}(s) + sL\hat{I}(s) + \frac{\rho l}{\bar{A}}\hat{I}(s) - \frac{\rho l\bar{I}}{\bar{A}^2}\hat{A}(s) \\
&\mathcal{L}\left\{\dot{\hat{A}}(t) = -c\hat{I}(t)\right\} \Rightarrow s\hat{A}(s) = -c\hat{I}(s)
\end{aligned} \tag{4.4}$$

Now that the state equations have been Laplace transformed, they can be turned into a transfer function, by first finding a transfer function for each input/output combination for the different states in Eq. 4.4, and then combining them into one transfer function, as shown on Figure 4.1.

$$
\begin{aligned}
H_1(s) =& \frac{\hat{U}_c(s)}{\hat{I}(s)} = sL + R + \frac{\rho l}{\bar{A}} \\
H_2(s) =& \frac{\hat{U}_c(s)}{\hat{A}(s)} = -\frac{\rho l \bar{I}}{\bar{A}^2} \\
H_3(s) =& \frac{\hat{A}(s)}{\hat{I}(s)} = -\frac{c}{s} \\
H(s) =& H_1(s) + (H_3(s)H_2(s)) = sL + R + \frac{\rho l}{\bar{A}} + \frac{\rho l c \bar{I}}{\bar{A}^2 s} \\
=& \frac{s^2 L \bar{A}^2 + s(R\bar{A}^2 + \rho l \bar{A}) + \bar{I}\rho l c}{s\bar{A}^2}
\end{aligned}
\tag{4.5}
$$

One combined transfer function, $H(s)$, is possible because both $\dot{U}_c(t)$ and $\dot{A(t)}$ have $I(t)$ as input. Figure 4.1 shows a system block diagram for the entire system from current input to voltage output.

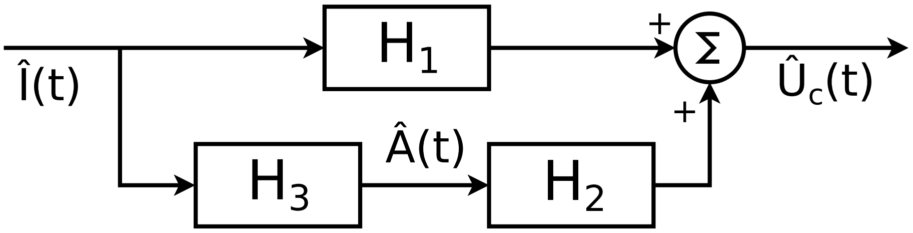


Figure 4.1: Block diagram of the system transfer function.

The transfer function, $H(s)$ form Eq. (4.5, has more zeros than poles, 2 zeros and 1 pole. This is a non proper transfer function, which is not desired. To remedy this, a small time constant is included in the model for the first state, U_c, to add a pole to the transfer function, see Eq. (4.6). The model for the second state, A stays the same as before.

$$
\tau\dot{U}_c(t) + U_c(t) = RI(t) + L\dot{I}(t) + I(t)\frac{\rho l}{A(t)} \tag{4.6}
$$

The equation is then again linearized:

$$
\begin{aligned}
\tau\dot{U}_c(t) + U_c(t) = RI(t) + L\dot{I}(t) + I(t)\frac{\rho l}{A(t)} \approx \\
\tau\dot{\hat{U}}_c(t) + \hat{U}_c(t) = R\hat{I}(t) + L\dot{\hat{I}}(t) + \frac{\rho l}{\bar{A}}\hat{I}(t) - \frac{\rho l \bar{I}}{\bar{A}^2}\hat{A}(t)
\end{aligned}
\tag{4.7}
$$

Then Laplace transformed:

$$
\begin{aligned}
&\mathcal{L}\left\{\tau\dot{\hat{U}}_c(t)+\hat{U}_c(t)=R\hat{I}(t)+L\dot{\hat{I}}(t)+\frac{\rho l}{\bar{A}}\hat{I}(t)-\frac{\rho l\bar{I}}{\bar{A}^2}\hat{A}(t)\right\}\Rightarrow\\
&\tau s\hat{U}_c(s)+\hat{U}_c(s)=R\hat{I}(s)+sL\hat{I}(s)+\frac{\rho l}{\bar{A}}\hat{I}(s)-\frac{\rho l\bar{I}}{\bar{A}^2}\hat{A}(s)
\end{aligned}
\tag{4.8}
$$

leading to changed transfer functions $H_1(s)$ and $H_2(s)$, see Eq. (4.9), which combined with the unchanged $H_3(s)$, results in a changed combined transfer function $H(s)$, as also shown on Eq. (4.9):

$$
\begin{aligned}
H_1(s)=&\frac{\hat{U}_c(s)}{\hat{I}(s)}=\frac{sL+R+\frac{\rho l}{\bar{A}}}{\tau s+1}\\
H_2(s)=&\frac{\hat{U}_c(s)}{\hat{A}(s)}=\frac{-\frac{\rho l\bar{I}}{\bar{A}^2}}{\tau s+1}\\
H_3(s)=&\frac{\hat{A}(s)}{\hat{I}(s)}=-\frac{c}{s}\\
H(s)=&H_1(s)+(H_3(s)H_2(s))=\frac{\bar{I}\rho lc}{\bar{A}^2(\tau s+1)s}+\frac{sL+R+\frac{\rho l}{\bar{A}}}{\tau s+1}\\
=&\frac{s^2L\bar{A}^2+s(R\bar{A}^2+\rho l\bar{A})+\bar{I}\rho lc}{s^2\bar{A}^2\tau+s\bar{A}^2}
\end{aligned}
\tag{4.9}
$$

The transfer function $H(s)$ in Eq. (4.9) shows that the addition of the time constant have resulted in an extra pole, bringing the number of poles up to 2. This means there is now an equal amount of poles and zeros, making $H(s)$ a proper transfer function.

Now that equations for all the system states has been derived, they can be written on state space from, first as a non-linear state space equation on the form $\dot{x}=f(x,u)$ and then as a linear state space equation on the form $\dot{x}=Ax+Bu$. Eq. (4.10) shows the non-linear representation, while Eq. (4.11) shows the linear representation derived from the transfer function given in Eq. (4.9).

$$
\begin{aligned}
\dot{U_c}(t)&=-\frac{1}{\tau}U_c(t)+\left(R+\frac{\rho l}{A(t)}\right)I(t)\frac{1}{\tau}+L\dot{I}(t)\frac{1}{\tau}\\
\dot{A}(t)&=-cI(t)
\end{aligned}
\tag{4.10}
$$

$$
\begin{aligned}
\begin{bmatrix}\dot{x_1}(t)\\ \dot{x_2}(t)\end{bmatrix}&=\begin{bmatrix}0 & 1\\ 0 & -\frac{1}{\tau}\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}+\begin{bmatrix}0\\ 1\end{bmatrix}I(t)\\
U_c(t)&=\begin{bmatrix}\frac{\bar{I}\rho lc}{\tau\bar{A}^2} & \frac{(R\tau-L)\bar{A}+l\rho\tau}{\tau^2\bar{A}}\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}+\frac{L}{\tau}I(t)
\end{aligned}
\tag{4.11}
$$

As the system modelled by these equations, is to be implemented on a computer and applied to discrete data sets, it has to be transformed from continuous time to discrete time. This is done using the forward Euler method [24].

$$\begin{aligned} U_c(k+1) &= U_c(k) + T_s\left(-\frac{1}{\tau}U_c(k) + \left(R + \frac{\rho l}{A(k)}\right)I(k)\frac{1}{\tau} + L\frac{I(k)-I(k-1)}{T_s}\frac{1}{\tau}\right) \\ A(k+1) &= A(k) - T_s(cI(k)) \end{aligned} \tag{4.12}$$

$$\begin{aligned} \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} &= \begin{bmatrix} 0 & \frac{1}{T_s} \\ 0 & 1-\frac{\frac{1}{\tau}}{T_s} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{T_s} \end{bmatrix} I(k) \\ U_c(k) &= \begin{bmatrix} \frac{\bar{I}\rho lc}{\tau\bar{A}^2} & \frac{(R\tau-L)\bar{A}+l\rho\tau}{\tau^2\bar{A}} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \frac{L}{\tau}I(k) \end{aligned} \tag{4.13}$$

Where:

T_s = Sampling rate [s]

With the discrete state space model presented in Eq. (4.12) and (4.13), an extended kalman filter can now be designed, to estimate the system states.

4.1.1 Test of Transfer Function

In order to validate the system model, a linear simulation of the system is performed. This is done in MATLAB using the function *lsim* on the system transfer function in Eq. (4.9). As the model only is valid when the system is short circuited, the simulation is ran only upon measurements from when the system is short circuited. Also the measurements are divided into segments of measurements from single short circuit periods, as the bridge model has to be reset after each rupture. When using the *lsim* function for a transfer function, clearest results are achieved if the input magnitude starts at 0, so input data magnitude is altered to start at 0, while still following the original shape. Figure 4.2 shows the result of the simulation using the parameters given in Table 4.1 in Section 4.3.

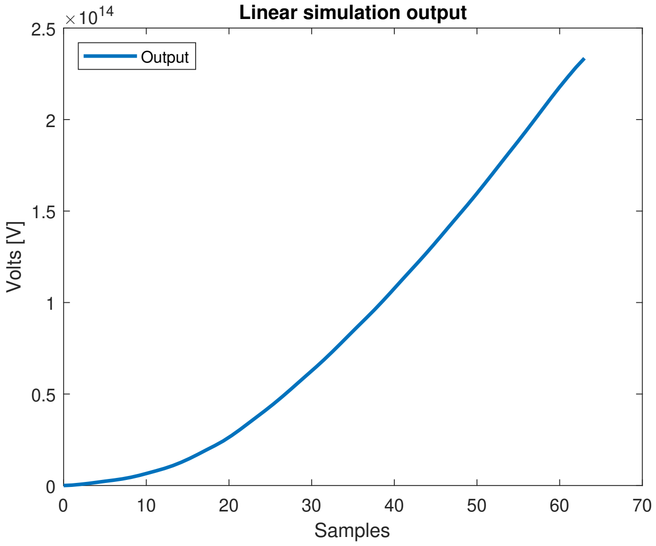


Figure 4.2: Voltage output of the linear simulation of the linearized system transfer function.

The output of the simulation, very vaguely resembles the shape of the voltage output from the real system, shown in Figure 4.3, the magnitude for the simulation output is however several time larger than that of the real system. The parameter values used values can be seen in Table 4.1. The big difference in magnitude of the outputs might result in inaccurate state approximation, but to test this, the extended kalman filter has to be designed.

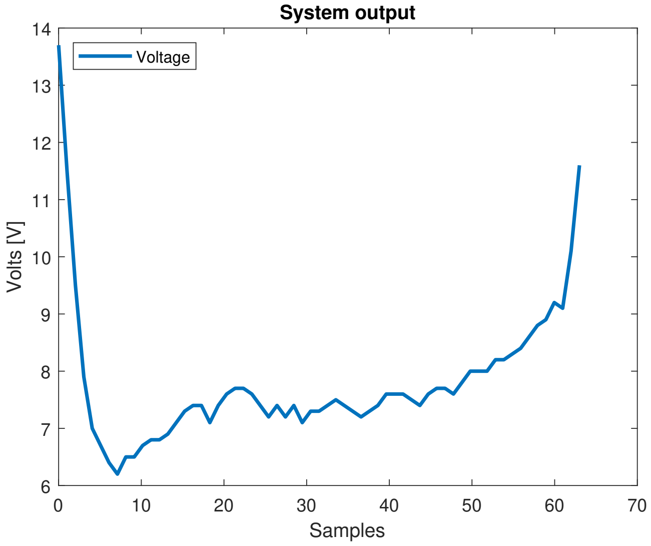


Figure 4.3: System output for one short circuit.

4.2 Extended Kalman Filter

The extended kalman filter is an expansion of the normal kalman filter. The extended kalman filter is used for estimation of non-linear systems in the same way as the normal kalman filter is used for linear systems, and can be described by the same block diagram, shown in Figure 4.4.

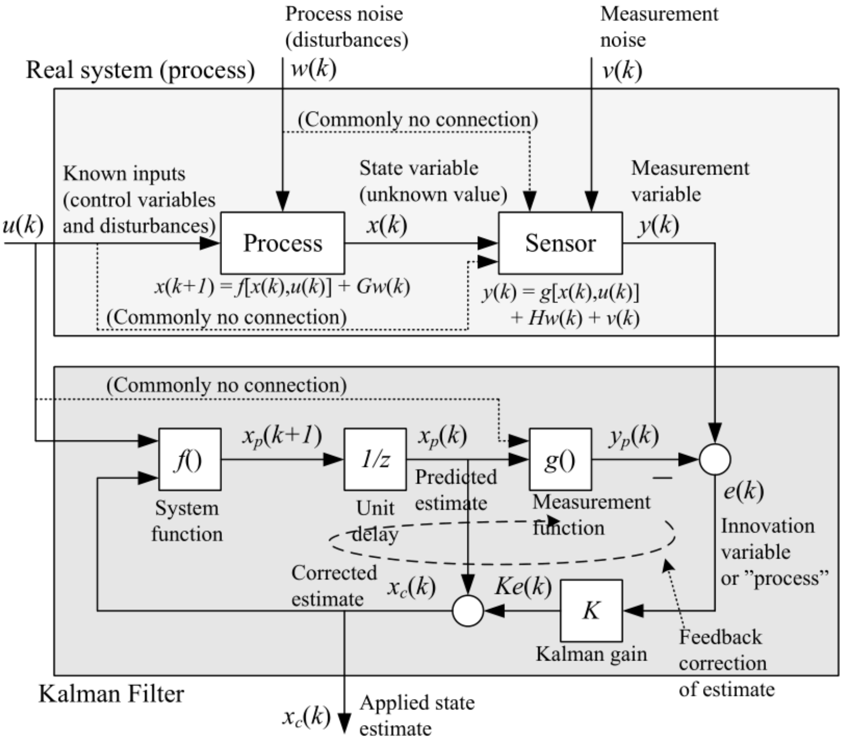


Figure 4.4: Block diagram for an extended kalman filter. [24].

4.2.1 Kalman Filter Design

The extended kalman filter works by first calculating an uncorrected estimated output based on the previous corrected estimate. This new estimate is then compared to the real system output, resulting in an error value, that multiplied with a kalman gain can be used to correct the estimate. The now corrected estimate is then used to calculate the next uncorrected estimate [24]. All of this is done using the non-linear system model, see Eq. (4.12). The kalman gain however is calculated using linear approximation of the system, see Eq. (4.13). However the model from Section 4.1 can not be used, as it does not have U_c and A as system states. A new partial model is then made by calculating the Jacobian matrix [24] for the both the system and output functions, see Eq. (4.12). This results in two matrices $\mathbf{A}$ and $\mathbf{C}$, shown on

Eq. (4.14). The **B** and **D** matrices are not calculated, as they are not needed for the filter calculations.

$$\mathbf{A} = \begin{bmatrix} 1 - \frac{T_s}{\tau} & -\frac{\rho l \bar{I} T_s}{\bar{A}^2 \tau} \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{4.14}$$

For calculation of the kalman gain K, the initial values for the auto-covariance matrices, **Q** and **R**, (the process and measurement noise) are given as:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{R} = \begin{bmatrix} 1 \end{bmatrix} \tag{4.15}$$

With the new linearized partial model and auto-covariance matrices, the kalman filter is now implemented in MATLAB

4.3 Test of Kalman Filter

The extended kalman filter implemented in MATLAB, is now tuned and tested. Tuning is done by first assuming initial values for the systems parameters, then slowly changing the parameters after each filter iteration, until an acceptable result is achieved. Some of the he initial values for the system parameters are based upon prior research done on a similar system [11], parameters like resistance, inductance and resistivity of the welding wire. Other parameters are determined by either the measurement settings, e.g. the sample rate, or by how the mathematical model has been derived, e.g. the operating points. Table 4.1 shows the final values of all parameters after tuning of the kalman filter. The initial conditions for U_c and A are chosen to be 0 and $3.142 \cdot 10^{-6}$.

Constant	**Value**	**Unit**	**Description**
R	$10 \cdot 10^{-3}$	Ω	Resistance
L	$230 \cdot 10^{-6}$	H	Inductance
l	$2 \cdot 10^{-3}$	m	Distance from wire tip to workpiece
ρ	$2 \cdot 10^{-7}$	$\frac{\Omega}{m}$	Resistivity of wire
τ	0.01	s	Time constant
T_s	$\frac{1}{50000}$	s	Sample rate
c	3	-	Bridge constant
$\bar{A}$	$3.142 \cdot 10^{-6}$	m^3	Operating point, cross-sectional area
$\bar{I}$	120	A	Operating point, current

Table 4.1: Values for system parameters used in testing of the kalman filter.

As mentioned in Section 4.1.1, the model for the system is valid only when the system is short circuited. The kalman filter is therefor tested on measurements only from system short circuit periods, divided into segments of single short circuits, and tests are carried out for a single short circuit period at a time, as the model for the cross sectional area has to reset for the start of each new short circuit period as noted in Section 4.1.1. Figures 4.5, 4.6 and 4.7 shows the results of the extended kalman filter. It is seen on Figure 4.6, that the corrected estimate of U_c fits very well to the real value, while the estimate of A, see Figure 4.7, is quite poor, as the estimate quickly becomes negative, which is physically impossible. No explanation has been found for this behavior.

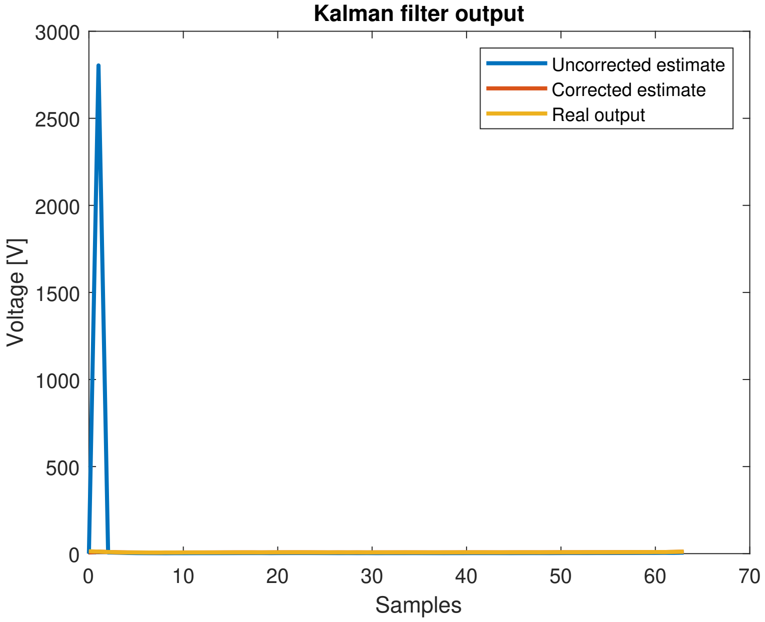


Figure 4.5: Comparison between the uncorrected, corrected and real voltage output of the kalman filter.

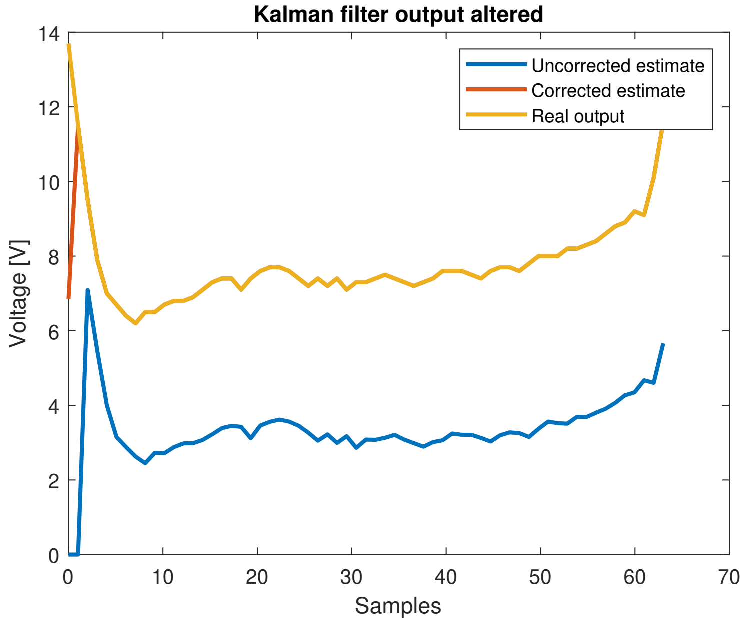


Figure 4.6: Same comparison as Figure 4.5, but with the 2 first samples of the uncorrected output removed of better visual clarity.

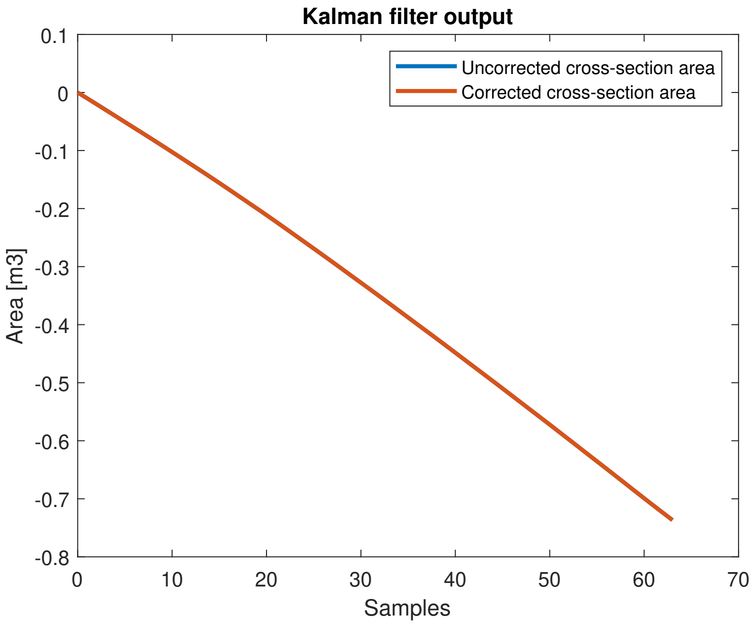


Figure 4.7: Comparison of the uncorrected and corrected cross-section area of the metal bridge.

Figure 4.5 shows the magnitude of the uncorrected output, rises fast at the beginning, but quit soon falls to follow the corrected output and real values, most of the time below these, as seen on Figure 4.6. The filter is tested with several different combinations of parameters, inputs and covariance matrices, all resulting in an estimate of the cross section that becomes 0 long before the metal bridge collapses. This may be the result of the model by its relative simplicity being incapable of properly describing the physics of the metal bridge.

4.3.1 Residual

The kalman filter may provide another possible feature that worth considering to employ for classification, is the residual. The residual is used to correct the predicted output of the kalman filter as it is the difference between the real system output and the non corrected output of the kalman filter, denoted $e(k)$ on Figure 4.4. Figure 4.8 shows the residual generated by the same kalman filter with 4 different inputs. The first 2 samples of the residual is discarded and not shown on Figure 4.8, as the residual generated by the firsts samples is magnitudes larger than the following, making it harder to find tendencies. It also takes the kalman filter 2 samples to have the corrected output match the real output.

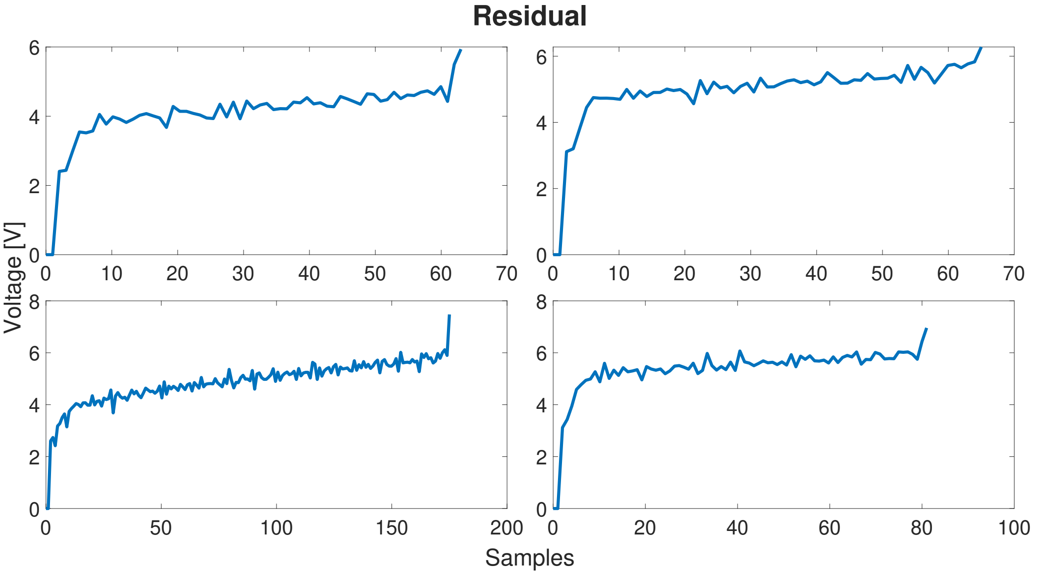


Figure 4.8: Residual generated by the kalman filter for 4 different short circuits.

When comparing the different residuals, a clear tendency emerges. The residual exhibits minor rapid fluctuations around a trend that slowly rises until the last few samples, where the residual starts to increase steeply. This tendency is consistent for both short and long short circuits. In order to test the relevancy of the residual as a feature for classification, an error signal is calculated for all data sets. This signal consists of the residuals generated by the kalman filter for each short circuits period, placed in time so they each lines up with the voltage and current signals for the according short circuit period in the measurement data set. The mRMR method from Chapter 3 is then used to determine the relevancy of the residual, compared to the other features selected for classification, see Section 3.4. Figure 4.9 shows the resulting mRMR feature importance score with the residual denoted as f_{13}.

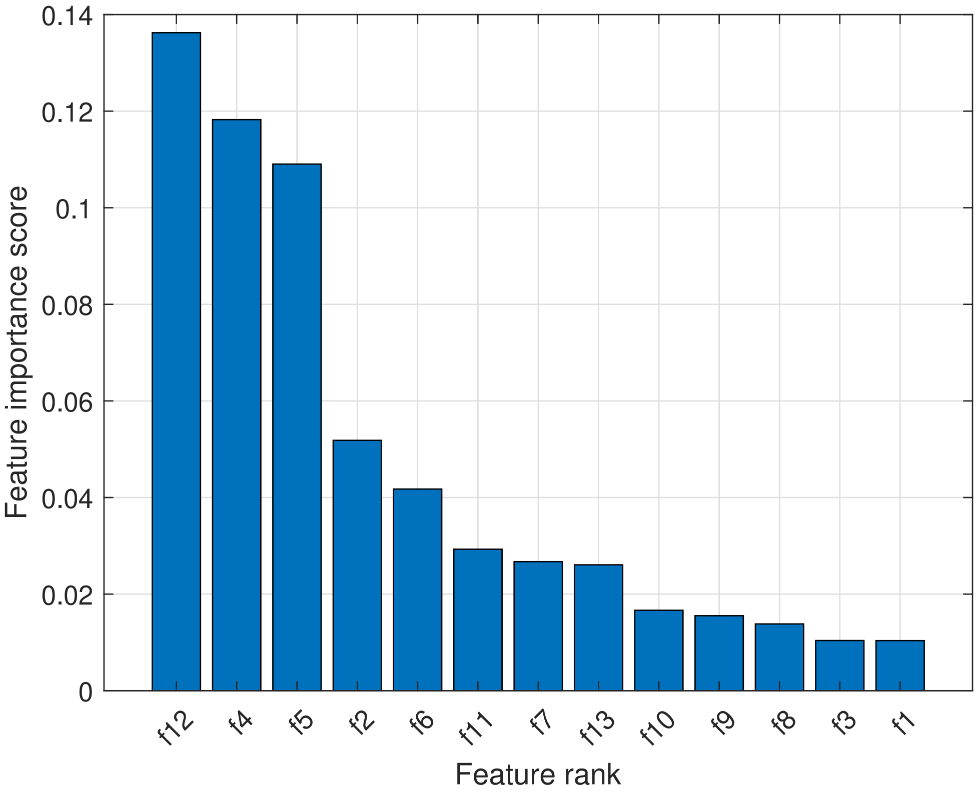


Figure 4.9: Updated feature ranking based on mRMR. The residual is feature 13

From Figure 4.9, the importance score of the residual, f_{13}, is almost as high as feature f_7 and f_{11} (Sample Mean and IQR). This means the residual is a more redundant feature, but is however still worth to include in the feature set tested for classification, to see how this feature impacts the classification algorithm.

4.4 Chapter Summation

In this chapter, a kalman filter was designed with the goal of estimating system states for use as additional features for clasification. Targeted state was the cross-sectional area of the molten metal bridge that exists when the system is in the short circuit phase. The kalman filter was however unable to produce a usable estimation of the cross-sectional area, but the residual generated by the filter proved to be a usable feature of some relevancy. In the following chapter, this feature, along with the features selected in Chapter 3, is tested for different classification algorithms to determine which algorithm is best suited.

Chapter 5

Classification

This chapter contains an introduction to multiple classification model options, followed by a comparison between these to determine the best performing model. Optimum hyperparameters are found for the determined model, the optimal classification is tested, and the performance of the model is evaluated.

5.1 Comparison of Classification Models

In this section, comparison of various classification methods is conducted. The methods considered are selected based on frequent occurrence in literature [25][12]. The methods are:

- Naive Bayes Classifier
- K-Nearest Neighbour
- Discriminant Analysis
  - Linear Discriminant Analysis
  - Quadratic Discriminant Analysis
- Support Vector Machines

An introduction to the selected classification methods is given in the following, in addition hyperparameters for each of the methods are described. Hyperparameters are parameters set to fixed values before model training, meaning they are not determined while the model is fit to the training data. The hyperparameters therefore influence the performance of the trained model, and they may optimised.

5.1.1 Naive Bayes Classifier

The Naive Bayes Classifier was introduced previously, in Section 3.2.3.

Hyperparameters

When fitting a Naive Nayes classifier, there are several parameters to optimise in order to obtain the best performance. These parameters are as follows:

- Distribution: Class probability density distribution.

5.1.2 K-Nearest Neighbour (KNN)

The K-Nearest Neighbour (KNN) classification method [18] is one of the simplest methods. Given an input sample, the classification of the sample is determined by the classes of the k nearest samples, in the training data, defined by a neighbourhood function,. The class of the input sample is then the majority class of the K neighbouring samples, ie. the class to which the majority of the K neighbouring samples belong. The neighbourhood function determines for a given input sample the K-nearest neighbouring samples based on a specific measure of distance, e.g. city block, euclidean etc.
An illustration of a 5-Nearest Neighbour classification of an input sample is shown in Figure 5.1.
For KNN classification, the classification model is the training data set itself, which may be a limiting factor depending on the implementation platform for the model. It is also an aspect worth consideration that the model complexity increases with training data sample size, and with the value of K.

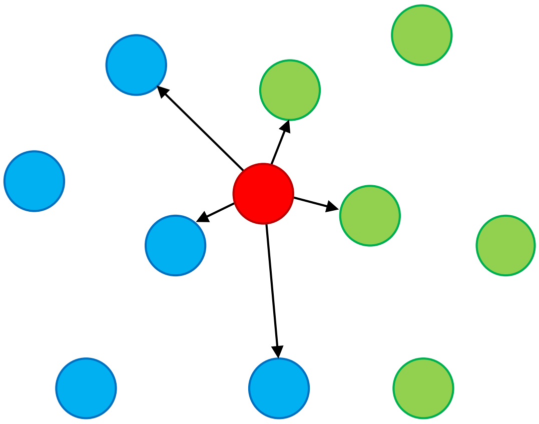


Figure 5.1: Illustration of 5-Nearest Neighbour example: The red sample is the input sample to be classified. Blue samples belong to class 1, green samples to class 2. Since the majority of the 5 nearest samples, indicated with arrows, belongs to class 1, the red input sample is classified as belonging to class 1.

Hyperparameters

When fitting a KNN classifier, a couple of parameters to optimised in order to obtain the best performance. These parameters are as follows:

- Distance function. Choices include City block, Euclidean, Mahalanobis and more.
- Number of neighbours K

5.1.3 Discriminant Analysis

Similar to the Naive Bayes classifier, discriminant analysis relies on estimating the probability densities of the classes, i.e. the probability of having a data sample of a certain class given the value of the data sample, in this case Gaussian densities, in order to determine so called decision boundaries defined by the decision function [18]. Two different discriminant analysis techniques, Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) exist. The two techniques differ in the way the decision function forms the class boundaru by linear or quadratic functions of the sample value respectively. Figure 5.2 illustrates the decision boundary forms in discriminant analysis

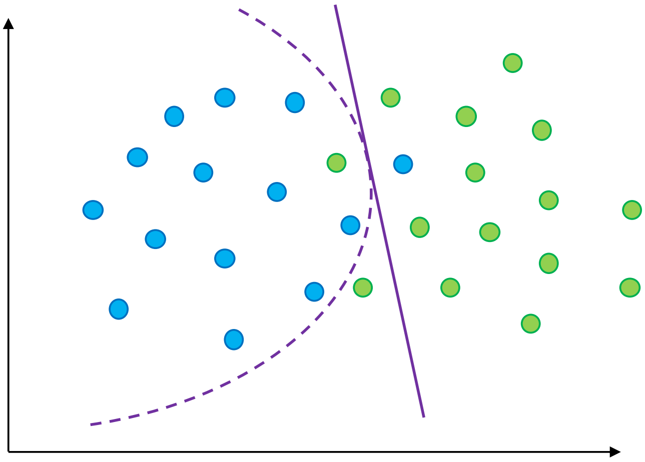


Figure 5.2: Illustration of the linear (solid) and quadratic (stipled) decision boundaries of LDA and QDA respectively. Colours of the data samples indicate their true class membership; blue, green are different classes.

Linear Discriminant Analysis (LDA)

Analogous to the Naive Bayes classifier, the discriminant analysis also relies on Bayes Rule for estimating the class posterior probability, i.e.

$$\hat{P}(Y = k|X = x) = \frac{\hat{P}(X = x|Y = k)\hat{P}(Y = k)}{\hat{P}(X = x)} \tag{5.1}$$

Where:

- Y is the random variable corresponding to the class of an observation, amongst the set of classes $c_1, ..., c_K$.
- $X = x$ is the random variable corresponding to an observation of the features $f_1, ..., f_P$.
- $\hat{P}(Y = k)$ is the prior probability estimate, also denoted $\hat{pi}_k$.
- $\hat{P}(X = x|Y = k)$ is the likelihood estimate, also denoted $\hat{f}_k(x)$.

The prior probability $\hat{P}(Y = k) = \hat{\pi}_k$ is estimated from the input data samples. The likelihood estimate $\hat{P}(X = x|Y = k) = \hat{f}_k(x)$ is modelled as a multivariate normal distribution.
Assume the prior probability π_k and the likelihood $f_k(x)$ are both known exactly.

With the multivariate normal distribution model, the likelihood $f_k(x)$ may be written:

$$f_k(x) = \frac{1}{(2\pi)^d|\Sigma|^{\frac{1}{2}}} \exp\Big(-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\Big) \tag{5.2}$$

Where:

- μ_k is the mean of input data samples belonging to class c_k.
- Σ is the class covariance matrix, assumed common for all classes.
- d is the dimension of the feature space.

Bayes Rule Eq. (5.1) are now reformulated in terms of the above, while all terms not depending on the class k are collected into constant terms C, C', C''. This gives:

$$\begin{aligned} P(Y = k|X = x) &= \frac{f_k(x)\pi_k}{P(X = x)} = C \times f_k(x)\pi_k \\ &= \frac{C\pi_k}{(2\pi)^d|\Sigma|^{\frac{1}{2}}} \exp\Big(-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\Big) \\ &= C'\pi_k \exp\Big(-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\Big) \end{aligned}$$

Taking the logarithm on both sides yields

$$\log P(Y = k|X = x) = \log C' + \log \pi_k - \frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) \tag{5.3}$$

From Eq. (5.3) it follows, the likelihood $P(Y = k|X = x)$ is maximised over k for the expression below being maximised, ie.e the right hand side of Eq. (5.3) disregarding the constant term C'.

$$\log \pi_k - \frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) = C'' + \log \pi_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + x^T\Sigma^{-1}\mu_k$$

From this the decision function $\delta_k(x)$ is defined as:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \Sigma^{-1} \mu_k + x^T \Sigma^{-1} \mu_k \tag{5.4}$$

An observation x is classified to belong to the class k for which the decision function has maximum value.

The linear decision boundary in LDA are the points in feature space, i.e. the sample values x, where 2 classes perform equally well with respect to the decision function. E.g. the boundary between the classes k and l is given by:

$$\delta_k(x) = \delta_l(x)$$
$$\log \pi_k - \frac{1}{2}\mu_k^T \Sigma^{-1} \mu_k + x^T \Sigma^{-1} \mu_k = \log \pi_l - \frac{1}{2}\mu_l^T \Sigma^{-1} \mu_l + x^T \Sigma^{-1} \mu_l$$

which is a linear function of x.

Quadratic Discriminant Analysis (QDA)

QDA differs from LDA in that the class covariance matrix Σ_k may be different for different classes [18].

This means for QDA both mean μ_k and covariance matrix Σ_k have to be estimated for each class c_k.

Following the steps as above for LDA in reformulation of Bayes Rule, the decision function for QDA then becomes:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \Sigma_k^{-1} \mu_k + x^T \Sigma_k^{-1} \mu_k - \frac{1}{2} x^T \Sigma_k^{-1} x - \frac{1}{2}\log |\Sigma_k| \tag{5.5}$$

The decision function is now quadratic in x, and so is the decision boundary in between any two classes as well.

Hyperparameters

When fitting a discriminant analysis classifier, several parameters are to be optimised in order to obtain the best performance. These parameters which are applicable for both LDA and QDA, see Discriminant Type, are as follows:

- Delta. Linear coefficient threshold. Features with modelled linear coefficient less than the threshold can be excluded from the classifier model.
- Discriminant Type. Linear or Quadratic, with class covariance matrices estimated to be of type regular or diagonal, using the pseudo-inverse.
- Gamma. Amount of regularisation to apply when estimating the covariance matrix for each class.

5.1.4 Support Vector Machines (SVM)

A Support Vector Machine (SVM) classifier [18][26] is a linear classifier, mostly binary classifier. It is a popular classifier due to its good generalisation properties, i.e. the classification model is kept as general as possible by avoiding overfitting the model. The main idea of SVM is to fit a linear hyperplane to separate the two training data classes, such that the distance from the hyperplane to each class, ie.e the nearest data point of each class, respectively is the same and tlargest possible. The sum of these two distances is denoted the margin between the classes; hence the hyperplane is fitted such that the largest possible margin between the data classes is obtained.
Initially two parallel hyperplanes H_1 and H_2 are fitted such that neither makes a classification error, both hyperplanes H_1 and H_2 separate correctly all points of one class from all data points of the other class, while the distance in between these initial hyperplanes, the class margin, is maximised. The final hyperplane is the hyperplane located halfway between these initial hyperplanes, see Figure 5.3. The name Support Vector Machine comes from the data points which lie on these initial hyperplanes, and therefore are "supporting" the final separating hyperplane. These "supporting" points are seen as vectors, i.e. they are the support vectors of the classifier.

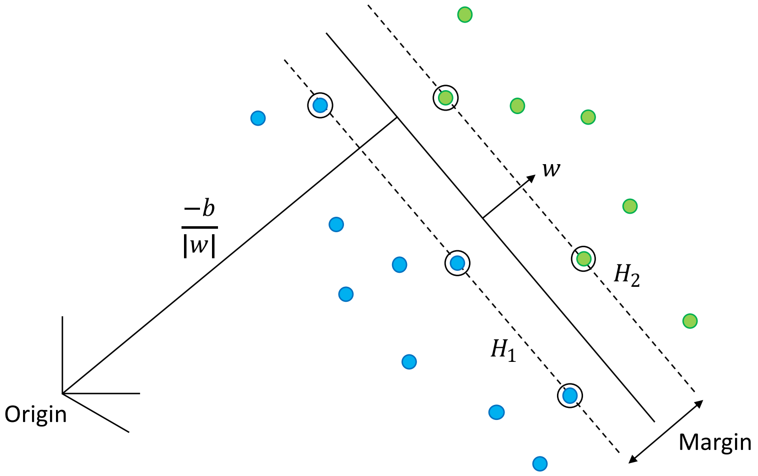


Figure 5.3: The maximal margin linear separation by SVM. The two data classes, blue and green, are separated by the margin between the two initial hyperplanes H_1 and H_2 (indicated by stipled lines); and the final hyperplane halfway in between these (shown as a solid line). Support vectors are circled.

The training data for the SVM are the data points x_i, with the corresponding label $y_i = \pm 1$, such that the labels for the two classes are -1 and $+1$ respectively. Element in the training data set may be written:.

$$\{x_i, y_i\}, i = 1, .., N, \; y_i \in \{-1, 1\}, \; x_i \in \mathcal{R}^d \tag{5.6}$$

Where N is the number of training data points. The points x lying on the final hyperplane satisfy

$$w^T x + b = 0 \tag{5.7}$$

where w is a normal to the hyperplane. Assuming, in the feature space considered the training data are linearly separable, points x lying on one of the initial hyperplanes H_1 and H_2, the hyperplanes located at the same side of the final hyperplane as the data points of the class with label -1 respectively same side as data points of the class with label -1. satisfy the first respectively the second equation below:

$$w^T x + b = +1 \text{ for } y_i = +1 \quad \text{(for initial hyperplanes boundary of class with label } +1) \tag{5.8}$$

$$w^T x + b = -1 \quad \text{(for initial hyperplanes boundary of class with label } -1) \tag{5.9}$$

The distance between these hyperplanes, the class margin is $\frac{2}{||w||}$, so to find the pair of hyperplanes resulting in the maximum margin, the aim is to minimise

$$\frac{1}{2}||w||^2 \tag{5.10}$$

subject to the constraints

$$w^T x + b \geq +1 \text{ for } y_i = +1 \tag{5.11}$$

$$w^T x + b \leq -1 \text{ for } y_i = -1 \tag{5.12}$$

which can be combined into one set of inequalities

$$y_i(w^T x + b) - 1 \geq 0 \quad \forall i \tag{5.13}$$

The solution to the optimisation problem can be found through a Lagrangian formulation of the problem [26]. The training of the SVM is then to maximize L_D in Eq. (5.14) with respect to the lagrange multipliers α_i,

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i^T x_j \tag{5.14}$$

subject to

$$w = \sum_i \alpha_i y_i x_i \tag{5.15}$$

$$\sum_i \alpha_i y_i = 0 \tag{5.16}$$

There is a lagrange multiplier α_i for every training point. In the solution to the SVM, data points where $\alpha_i > 0$ are the support vectors, lying on either of the margin boundaries, hyperplanes H_1 or H_2. All other points have $\alpha_i = 0$. The constant b in Eq. (5.7), is found by Eq. (5.13), using support vectors only.

The decision function for the SVM classifier is then

$$f(x) = w^T x + b = \sum_{i=1}^{N} \alpha_i y_i x_i^T x + b \tag{5.17}$$

from which prediction of class membership for new input sample x is made according to the sign of the decision function $f(x)$: x is predicted to have class label of same sign as $f(x)$.

SVM for linearly separable data as here is called hard margin SVM. A soft margin SVM definition exists for the case where the data is not linearly separable.

Non-Separable Case

When data is not linearly separable the optimization problem, for linearly separable data stated in Eq. (5.10)-(5.12) can be modified by adding a penalty for violating the constraints. The new problem is then to minimize

$$\frac{1}{2}||w||^2 + C \sum_{i=1}^{n} \zeta_i \tag{5.18}$$

subject to relaxed constraints

$$y_i(w^T x + b) - 1 + \zeta_i \geq 0 \quad \forall i \tag{5.19}$$

Here $\zeta_i \geq$ are "slack" variables, and C is a constant defining the allowed violation of the margin, where as $C \to \infty$ less violations are allowed, and the system eventually behaves like a hard margin case. The solution to the SVM optimisation problem for the non-separable case here is given in similar manner as for the linearly separable case, with the difference from the hard margin case is that $0 \leq \alpha_i \leq C$.

Kernel Functions

Initially the SVM classifier can perform only a linear separation of the feature space. However by mapping the feature vectors into a higher dimension space and apply the linear separation, a non-linear separation may be achieved in the original feature space. The mapping of feature vectors is performed by use of kernel functions. Kernel functions are are mapping functions of the feature space on to the high dimension space for the SVM to operate in. The SVM classifier considers data points, i.e. data vectors, only in the form of the inner products of those. The advantage of this is, the kernel functions allows for evaluation of the inner products without actually constructing the data vectors resulting from the mapping in the higher dimension space On result is that mappings can be performed to spaces of very high dimensions. Various kernel functions exist, including [26]:

- Linear Kernel
$$K(x_i, x_j) = (x_i^T x_j) \tag{5.20}$$
- Polynomial Kernel
$$K(x_i, x_j) = (1 + (x_i^T x_j))^p \tag{5.21}$$
- Radial Basis Kernel
$$K(x_i, x_j) = \exp\Big(- |x_i - x_j||^2 \Big) \tag{5.22}$$

In the above $K(x_i, x_j)$ defines the kernel function computing the inner product the mapped data vector.
Another way of defining a mapping is by explicitly stating the data vector resulting from the mapping of the original feature vector, e.g. as in the polynomial mapping exapme $\phi(x)$ given below for an original feature vector $x = [x_1 \quad x_2]$:

$$x \to \phi(x) = [x_1^2 \quad x_2^2 \quad \sqrt{2}x_1x_2 \quad \sqrt{2}x_1 \quad \sqrt{2}x_2 \quad 1]$$

As mentioned, use of the kernels in SVM means the resulting mapped feature vector as e.g. $\phi(x)$ above is not computed for each original feature vector, only the inner product of the mapped feature vectors $\phi(x)^T\phi(x)$ given in the kernel definition, in terms of the original feature vector x.

SVM is often used as the classifier in general has a good performance, and is quite easy to set-up due to the small amount of parameters and kernels to choose from. One drawback of SVM is that the training can be quite slow in the case of a large sample size. Likewise, testing can be slow in the case of a large number of support vectors.

Hyperparameters

When fitting a SVM classifier, several parameters are to be optimised in order to obtain the best performance. These parameters are as follows:

- Box constraint. Margin violating constraint C, the cost of misclassification of training data in soft margin SVM.
- Kernel scale. Feature data values are divided by kernel scale.
- Kernel function. Linear, polynomial or radial basis kernel.
- Polynomial order. Order of polynomial kernel from 2 to 4, only applies if the polynomial kernel is employed.

5.1.5 Comparison Results

A comparison between the the above, in Section 5.1.1-5.1.4, presented classification methods:

- Naive Bayes (NB)
- K-Nearest Neighbour (KNN)
- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- Support Vector Machines (SVM)

is conducted in the same manner as applied for the window length comparison in Section 3.2.3: Each classification method is evaluated using a 10-fold cross-validation scheme in order to obtain the performance across the entire training and validation data set.

From the welding process data set the 12 features selected for classification use, described in Chapter 3, listed in Section 3.2.2, and the residual feature described in Chapter 4, in particular in Section 4.3.1, are extracted and used for training of the methods, and for prediction of the data sample class.
Prior to classification, it is advised in [27] to linearly scale each feature in the training data to a range of $[-1 \quad 1]$ or $[0 \quad 1]$, to avoid features with larger numeric ranges from dominating other features. Hence, each feature f_i, $i = 1, ..., 13$, is scaled as follows:

$$f_i' = \frac{f_i - min(f_i)}{max(f_i) - min(f_i)} \tag{5.23}$$

Where f_i' is the scaled feature vector for feature f_i. Eq. (5.23) applied element wise, ensures all scaled feature f_i' have values in the range $[0 \quad 1]$ only. The same scaling, understood as scaling by the same off-set and divisor in Eq. (5.23) as used for scaling feature values in the training data, has to be applied to the original feature values in the data used for validating or testing the classification model, i.e. if a feature in the training data was scaled from a range of $[166.1 \quad 317.3]$ to $[0 \quad 1]$ using scale off-set 166.1 and divisor $317.3 - 166.1 = 151.2$, then the corresponding feature in the validation or testing set with a range of e.g. $[165.2 \quad 317.3]$ is scaled to $[-0.006 \quad 1]$.
Performing the mRMR test, described in Section 3.2.2, to acquire feature importance scores and rankings of features according to those after scaling of feature data, see Figure 5.4, shows that the relevance of some features have changed slightly, however the overall hierarchy of the features is still equivalent to what was found for the original features values, presented at the end of Chapter 4, Figure 4.9.

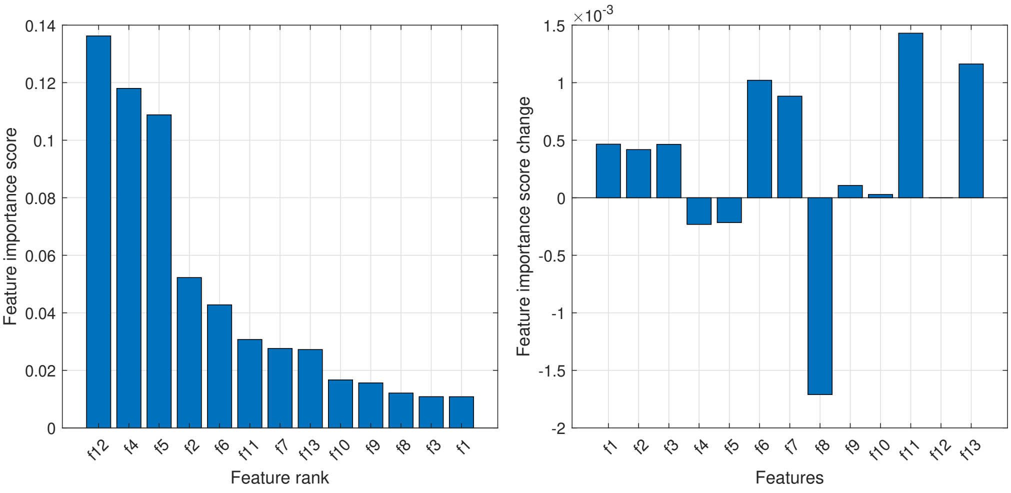


Figure 5.4: Left: Updated feature ranking and importance scores based on mRMR after scaling. Right: The changes in importance score for each feature from unscaled to scaled features.

The list of features considered for classification is as follows:

f_1 = Current, I

f_2 = Voltage, U

$f_3 = I^2 + U^2$

f_4 = Power, P

f_5 = Voltage Time Gradient, $\frac{dU}{dt}$

f_6 = Resistance, R

f_7 = Sample Mean of U

f_8 = Sample Variance of U

f_9 = Hjorth Mobility of U

f_{10} = Hjorth Complexity of U

f_{11} = Interquartile Range, IQR of U

f_{12} = Rupture Correlation of U

f_{13} = Residual from kalman filter

It is concluded in the the end of Section 3.2.2, that features f_4, f_5 and f_{12}, from the evaluations performed, are significantly more relevant for classification use than the remaining set of features. For purpose of clarification, the classification comparison is performed with three different feature sets as follows:

- $F = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}\}$
- $F_1 = \{f_4, f_5, f_{12}\}$
- $F_2 = \{f_1, f_2, f_3, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{13}\}$

The intend is to showcase the performance using the full feature set, a feature set including only the most relevant features, and as contrast the performance using a large number of features, here the remaining features, all of minor relevance.

The hyperparameters for the various classification methods in this comparison are chosen to be the default hyperparameter options in the MATLAB software [28]. These are as follows:

- Naive Bayes
  - Distribution: Normal
- K-Nearest Neighbour
  - Distance function: Euclidean
  - Number of neighbours: 1
- Linear Discriminant Analysis and Quadratic Discriminant Analysis
  - Delta: 0
  - Gamma: 0
- Support Vector Machines
  - Box constraint: 1
  - Kernel scale: 1
  - Kernel function: radial basis kernel
  - Polynomial order: Not applied

The results of the classification method comparison for the different feature sets, in terms of the confusion matrix based performance measures introduced in Section 3.2.3, are shown in Tables 5.1-5.3 respectively.
Confusion matrices for each classification method obtained for the comparison is shown in Appendix E, Figures E.1-E.3. Note, the true class populations are 14720 samples for "Rupture" and 139461 samples for "Short".

Method	**TPR**	**TNR**	**PPV**	**ACC**	**Average**
NB	0.8344	0.8987	0.4653	0.8925	0.7727
KNN	0.8469	0.9866	0.8697	0.9732	0.9191
LDA	0.5101	0.9938	0.8963	0.9475	0.8369
QDA	0.9416	0.7227	0.2641	0.7436	0.6680
SVM	0.5718	0.9962	0.9407	0.9556	0.8661

Table 5.1: Performance measures for the classification methods using feature set F. Highest result for measure highlighted in red

Method	TPR	TNR	PPV	ACC	Average
NB	0.5414	0.9851	0.7932	0.9427	0.8156
KNN	0.8309	0.9826	0.8349	0.9681	0.9041
LDA	0.3394	0.9899	0.7803	0.9277	0.7593
QDA	0.4092	0.9953	0.8997	0.9386	0.8091
SVM	0.5159	0.9863	0.7995	0.9414	0.8108

Table 5.2: Performance measures for the classification methods using feature set F_1. Highest result for measure highlighted in red

Method	TPR	TNR	PPV	ACC	Average
NB	0.7917	0.8952	0.4439	0.8853	0.7540
KNN	0.7771	0.9835	0.8328	0.9638	0.8893
LDA	0.4910	0.9916	0.8612	0.9438	0.8219
QDA	0.9379	0.6608	0.2262	0.6873	0.6280
SVM	0.5370	0.9871	0.8145	0.9441	0.8206

Table 5.3: Performance measures for the classification methods using feature set F_2. Highest result for measure highlighted in red

The comparison results show the overall best performance is obtained by KNN with feature set F. KNN is on average also the best method for the other feature sets, which is interesting as KNN is by far the simplest of the classification methods. Generally good results are found for all methods using feature set F. However for methods NB and QDA, it seems less is more, as these methods on average achieve better results with F_1. On the contrary to this, LDA performs better on average with F_2 than with F_1. SVM, which is the most complex of the methods has the overall second best performance on average for feature set F, however the TPR is low compared to other methods.

As KNN, from the comparison here conducted, now has been identified as the best performing classification method for the classification problem at hand, the optimisable hyperparameters for this method are investigated to determine whether the performance of this classifier can be futher improved.

5.2 Optimisation of Classification Model Hyperparameters

To improve the performance of the KNN classification method, which in Section 5.1.5 above was determined to have the best performance, optimal hyperparameters are identified. Recall from Section 5.1.2, for the KNN the optimisable hyperparameters

comprise the distance function and the number of neighbours K.
Distance function options are as follows [29]:

- City block distance
- Chebyshev distance
- Correlation distance
- Cosine distance
- Euclidean distance
- Jaccard distance
- Mahalanobis distance
- Minkowski distance
- Standardised Euclidean distance
- Spearman distance

The number of neighbours K, is investigated with values for K from 1 to 51 with a 2 neighbour increment. For each distance function and value of K, a 10-fold cross-validation is performed, and the confusion matrix based performance measures are evaluated.
Figures 5.5 and 5.6 show performance measures for the different distance functions and numbers of neighbours.

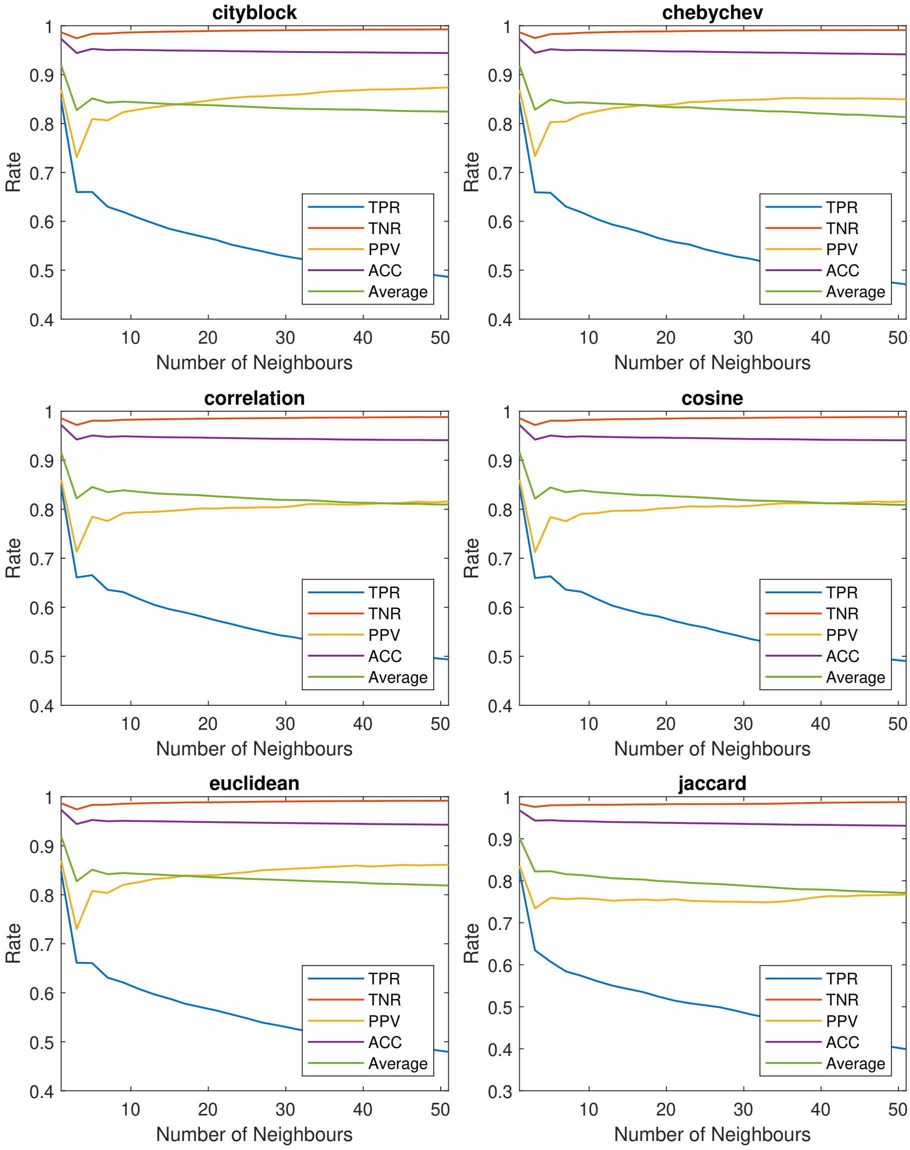


Figure 5.5: Performance measures for different distance functions as a function of number of neighbours. Note: Rate axis scale differ for distance function Jaccard graph.

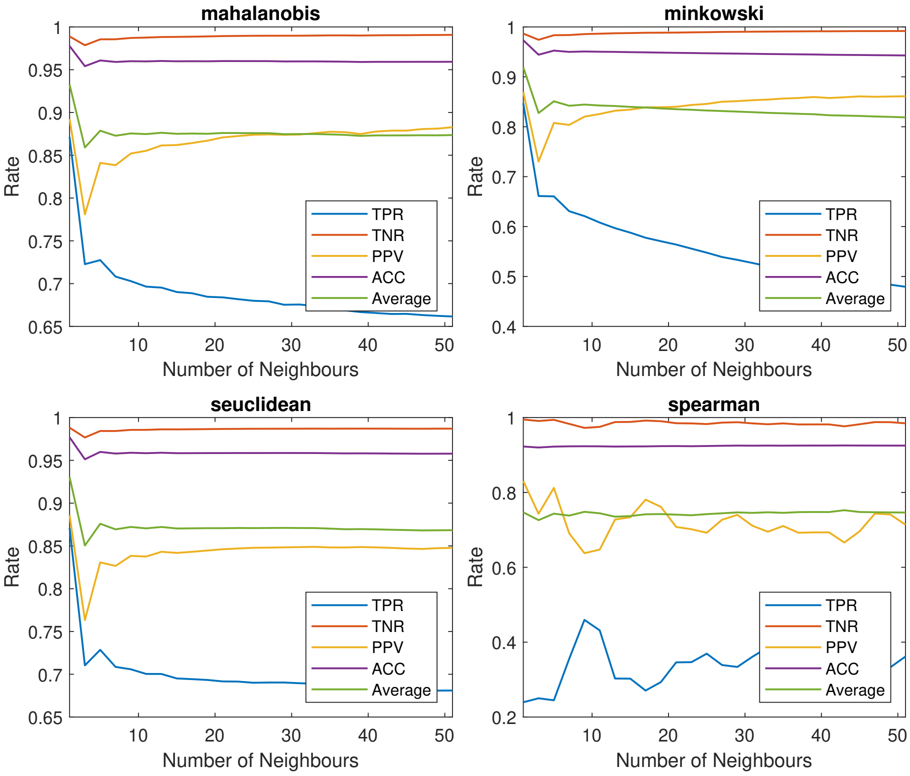


Figure 5.6: Performance measures for different distance functions as a function of number of neighbours. Note: Rate axis scale differ for the different distance function graphs.

The maximum performance measures observed among all the distance function performance measures obtained, are shown in Table 5.4. It is evident, the Mahalanobis distance function generally has the best performance, and the optimal number of neighbours is $K = 1$.

From this analysis the optimal classification method is then the 1-Nearest Neighbour algorithm with Mahalanobis distance function.

Max. Measure	**TPR**	**TNR**	**PPV**	**ACC**	**Average**
Rate	0.8714	0.9948	0.8924	0.9777	0.9326
Dist. Func.	Mahalanobis	Spearman	Mahalanobis	Mahalanobis	Mahalanobis
K	1	1	1	1	1

Table 5.4: Maximum performance measures with the corresponding distance function and number of neighbours K. For comparison: TNR = 0.9907 for Mahalanobis distance function with $K = 1$.

Following the above selection of an optimal classification model based on performance measures, by cross-validation, the selected model is tested on unused data to achieve an unbiased evaluation of the performance.

5.3 Test of Classification Model

The purpose of test here described of the optimal classification model, a 1-Nearest Neighbour classifier using Mahalanobis distance, is to provide a final unbiased evaluation of the model performance. The optimal model is trained on the training data previously used, however evaluation is performed upon classification of data not employed before in the selection and optimisation process. The data should however orginate from a welding process of the same conditions and welding system settings as the training data. It is expected, the performance obtained in testing is lower than in the performance obtained for the classification comparison based on cross-validation in Section 5.1.5.

Table 5.5 shows the confusion matrix based performance measures, see Section 3.2.3, obtained from the testing of the optimal classification model upon unused data as described above. It is notable, the performance is significantly lower than in the classification model selection cross-validation Tables. 5.1-5.4, especially compared to the best results of those presented in Section 5.1.5. More specifically, a smaller proportion of sample of class "Rupture" are being correctly classified, see TPR and PPV, while a larger proportion of samples of class "Short" are incorrectly classified, see TNR and PPV. Both changes contribute to lower accuracy ACC and overall average score. The observed discrepancy between the performance in the classification model selection cross-validation and the testing of unused data is significant.

Measure	**TPR**	**TNR**	**PPV**	**ACC**	**Average**
Rate	0.7290	0.9445	0.5787	0.9241	0.7941

Table 5.5: Performance measures from test of the Nearest Neighbour model

The aforementioned errors in classification is visualised are the confusion matrix, Figure 5.7. Note, the true class populations are 4927 samples for "Rupture" and 47155 samples for "Short".

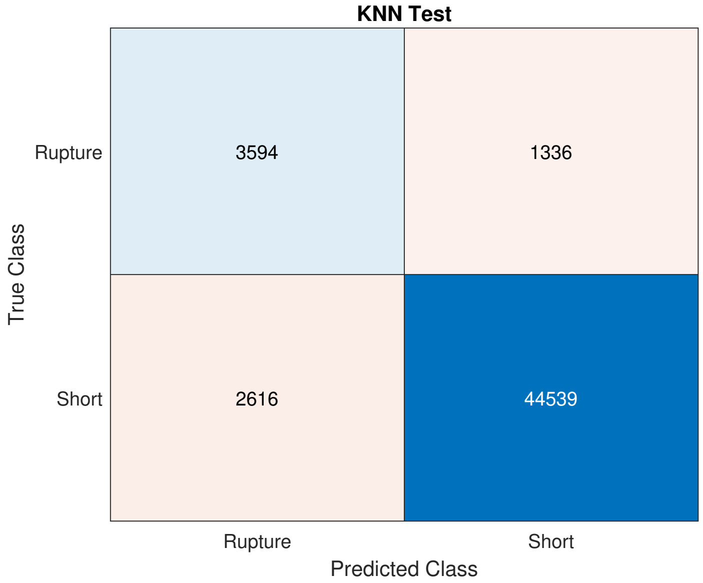


Figure 5.7: Confusion matrix from test of the Nearest Neighbour model.

Recall from the data pre-processing, all true rupture periods has a length of 10 data samples. By inspection of the predicted class for all samples in the true rupture periods, i.e. all samples of true class "Rupture", TPR for each sample number in the true rupture periods, i.e. TPR for all samples of timewise same location in a true period, can be computed for each sample number, sample 1 to sample 10, in the true rupture periods. The resulting TPR for each sample number in the rupture period are shown in Figure 5.8. The results show TPR for the first sample is 0.4634, meaning at this point fewer than 50% of "Rupture" class samples are correctly classified. By sample 6, five samples before rupture, TPR has increased to 0.7642. At sample 10, TPR is 0.998, meaning 99.8% of "Rupture" class samples are correctly classified at this point, however as per the problem formulation, this is too late in the rupture period to regulate the current to avoid a violent rupture. It could however be argued, it is better to predict the rupture late, than not predicting it at all, allowing the welding current to be lowered by some degree before rupture, though not to the desired current level.

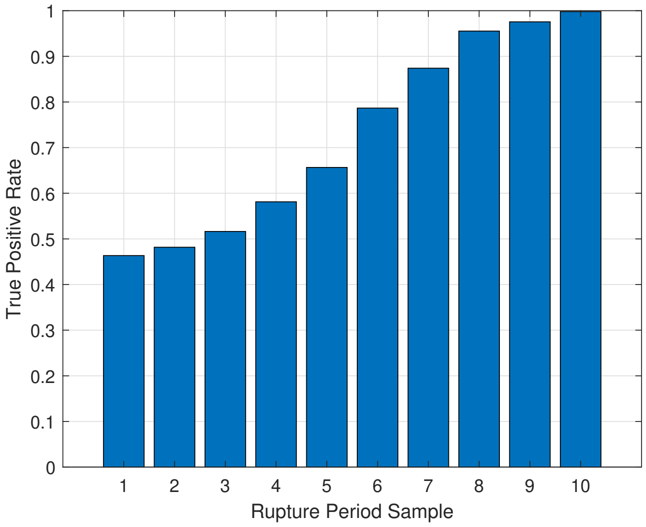


Figure 5.8: TPR for each rupture period sample

Obtaining false negative class predictions early in the rupture period is not a problem, as long as the rupture is otherwise predicted in due time to regulate the current properly. It is more of a concern to obtain a large number of false positive class predictions, samples that are predicted to be in the rupture period, but are not. As shown in Figure 5.8, the confidence in prediction of class membership for samples increase the further into the rupture period the welding process has reached. A system, based on the classification model here tested might be put in place to determine over a couple of samples, and not just one sample, whether the electrode metal bridge is about to rupture.

5.4 Test of Classification Model on Various Programs and Settings

Following the test above of the classification model on a data set acquired from a welding process applying the same welding program and system settings as applied for the welding process from which the training data were acquired, it is interesting to test the same model on data from welding processes applying different programs and settings in order to determine the generalisation properties of the model and the extracted features. The data sets have been produced and provided for this project by Migatronic A/S, and reflects realistic welding settings. Below are listed welding programs and settings for the data sets:

- Program 202

  - Current: 185A Voltage: 18.9V

- Program 112
  - Current: 65A Voltage: 15.9V
  - Current: 133A Voltage: 17.9V
- Program 113
  - Current: 106A Voltage: 15.7V
  - Current: 143A Voltage: 16.7V
  - Current: 181A Voltage: 19.4V

The data sets used hitherto used for training and testing, welding programs and settings were the first in the list, program 202, 185A current setting and 18.9V voltage setting.

Performance of classification model is evaluated for each of the data sets by means of the confusion matrix based performance measures, see Section 3.2.3. The results for each of the data sets is shown in Table 5.6.
From these performance measures, it is evident, performance of the classification model on the data sets acquired for programs 112 and 113 is not on par with the performance on the data set acquired for program 202. The performance is particularly worse on data sets from program 112, where TPR is low, meaning few samples of class "Rupture" are correctly classified and PPV is low, meaning a large proportion of false positives are found. Having a bad performance in both of these metrics is the worst case scenario. For data sets from program 113, primarily TPR is low, while PPV more acceptable.
The low TPR for programs 112 and 113 is further illustrated in Figure 5.9 showing TPR per sample number for sample 1 to sample 10 in the true rupture periods, where TPR for these programs is from extremely low to low, only acceptable for the last two samples, sample 9 and 10.
The conclusion must be, as the performance measures obtained in these cross-program, cross-setting tests indicate strongly, feature values extracted from a data set with a particular program and setting, are not equivalent to features extracted from another data set with a different program and setting.

Prg.	Curr. Set [A]	Volt. Set. [V]	TPR	TNR	PPV	ACC	Average
202	185	18.9	0.7290	0.9445	0.5787	0.9241	0.7941
112	65	15.9	0.1290	0.9249	0.0999	0.8767	0.5076
112	133	17.9	0.2239	0.9867	0.4994	0.9441	0.6635
113	106	15.7	0.1350	0.9995	0.9287	0.9623	0.7564
113	143	16.7	0.1748	0.9994	0.9271	0.9632	0.7661
113	181	19.4	0.2192	0.9997	0.9726	0.9665	0.7895

Table 5.6: Performance measures from test of the Nearest Neighbour model on various programs and settings

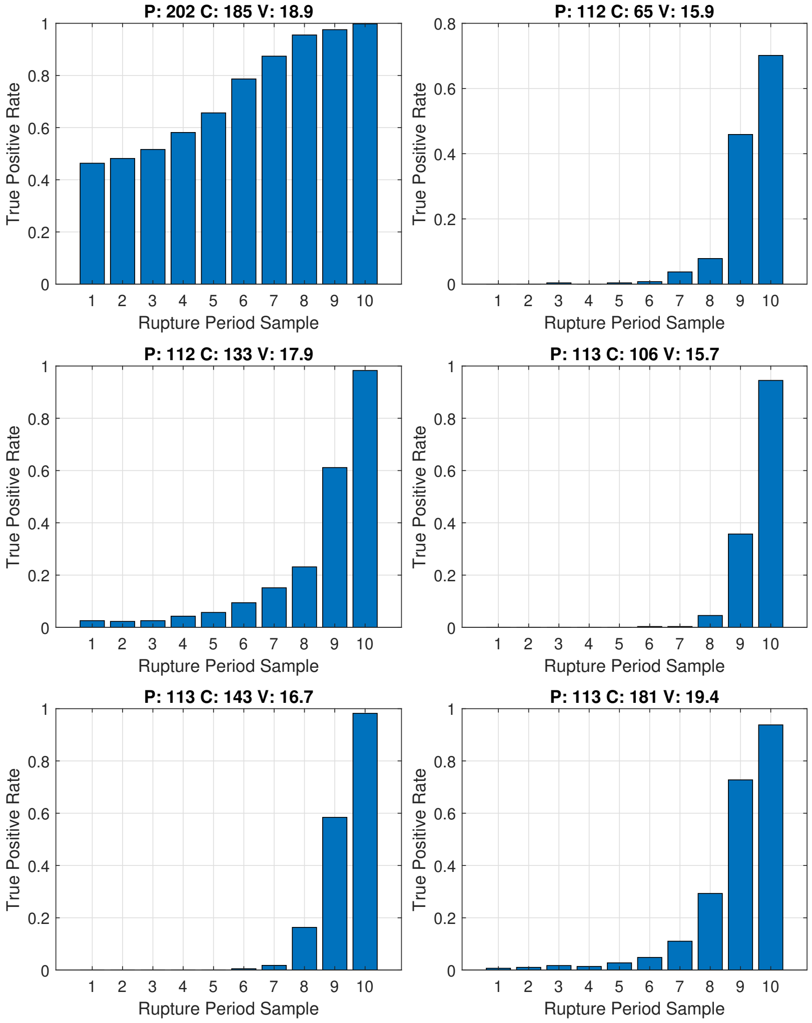


Figure 5.9: TPR for each true rupture period sample for various programs and settings. P is the program, C is the current setting, and V is the voltage setting.

Onwards from the prior conclusion, it is interesting to investigate whether the features extracted are useful at all in classification of rupture samples from rupture periods in data sets acquired from other welding programs than program 202. To determine this, a new Nearest Neighbour classification model is trained with data from all three programs 202, 112 and 113, with the different settings as listed in the start of the section.

In addition, the feature sets is extended with a program feature. In [27] it is advised to transform categorical predictors to numerical predictors in the same range as the other feature data is being scaled to, i.e. $[0 \quad 1]$. The transformation of values for the added program feature is as follows:

- Program 202 → 0
- Program 112 → 0.5
- Program 113 → 1

The performance measures obtained from testing the new Nearest Neighbour model on data sets from the different programs and settings is shown in Table 5.7. These measures show an increased performance for all programs and settings, but in particular for program 113, where the performance for 2 out of the 3 settings is better than for program 202. It is also evident, the performance for program 112 is not om par with the performance for program 202 and 113, which might indicate the features extracted are not as relevant for classification of data samples from this program, compared to data samples from programs 202 and 113.

In Figure 5.10 TPR per sample number for sample 1 to sample 10 in the true rupture periods is shown. From this it is evident, where for program 112, confidence in correct classification is not achieved until around rupture sample 7 or 8. In comparison the same confidence is achieved at rupture sample 5 for program 202 and at rupture sample 3 or 4 for program 113. The conclusion is, that in order to have an effective classification model for classification of samples from rupture periods from various welding programs and settings, the model has to be trained on data from the same variety of welding programs and settings.

Prg.	**Curr. Set [A]**	**Volt. Set. [V]**	**TPR**	**TNR**	**PPV**	**ACC**	**Average**
202	185	18.9	0.7314	0.9472	0.5912	0.9268	0.7993
112	65	15.9	0.3442	0.9984	0.9344	0.9588	0.8090
112	133	17.9	0.4917	0.9814	0.6105	0.9540	0.7594
113	106	15.7	0.7443	0.9945	0.8582	0.9837	0.8952
113	143	16.7	0.8132	0.9918	0.8207	0.9840	0.9025
113	181	19.4	0.8134	0.9285	0.3360	0.9236	0.7504

Table 5.7: Performance measures from test of the Nearest Neighbour model, with programs labels, on various programs and settings

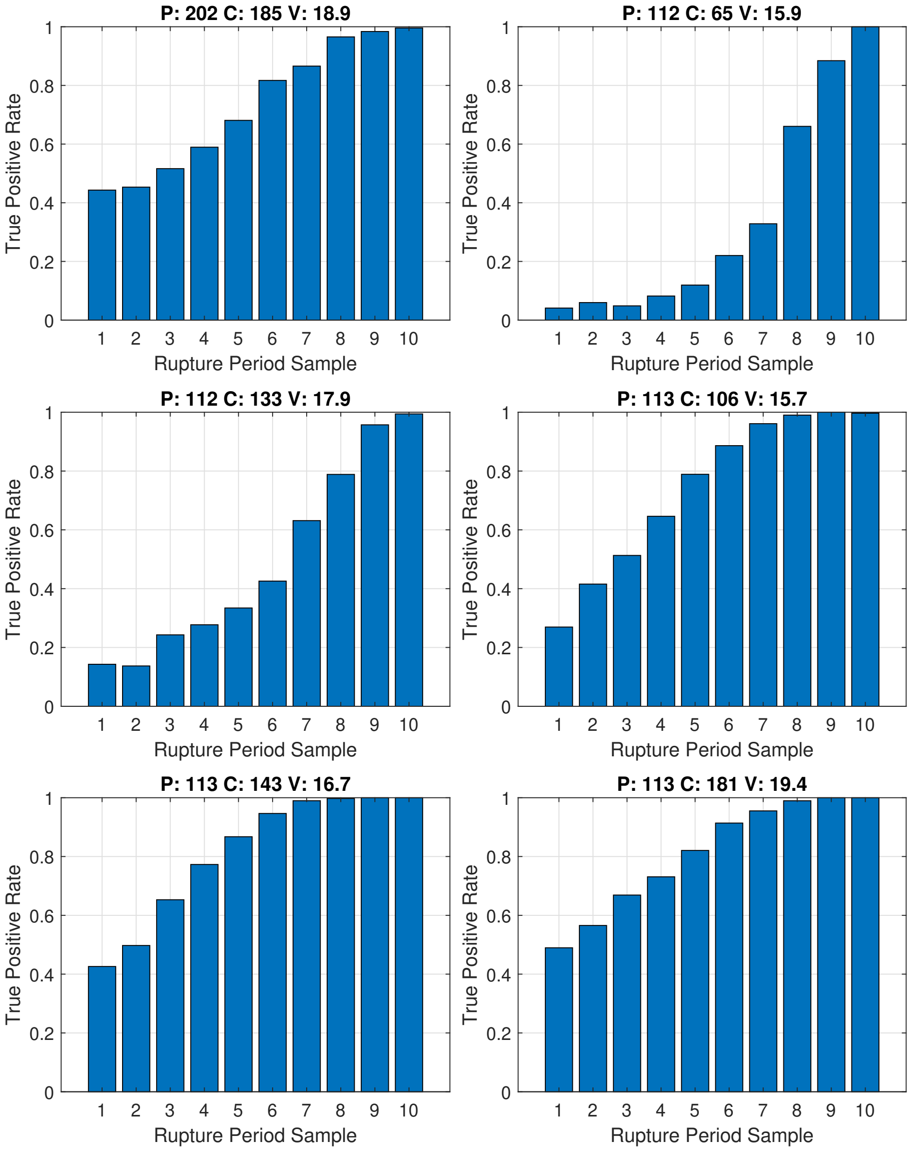


Figure 5.10: TPR for each true rupture period sample for various programs and settings. P is the program, C is the current setting, and V is the voltage setting.

5.5 Chapter Summation

In this chapter multiple options for classification model were introduced and the performance of these were compared based on cross-validation of the training data set. The K-Nearest Neighbour model was evaluated to be the best performing model, and optimum hyperparameters evaluated to be determined to be the Mahalanobis distance function with 1 neighbour. The performance of this optimal classification model was evaluated on testing data separate from the training data, where it was found, the performance especially lacked in terms of the number of false positives. The same classification model was tested on new acquired from various welding processes applying different welding programs and settings, with results showing the model trained on data from a particular welding program and setting, did not perform well in testing on data from other programs and settings. A new classification model of same algorithm and hyperparameters, K-Nearest Neighbour with Mahalanobis distance function and 1 neighbour, was trained on the full variety of programs and settings and a better interprogram performance was achieved.

Chapter 6

Conclusion

The aim of the work presented in this thesis is to investigate, and to develop a solution to the problem of predicting the start of free burn in the short-circuit phase of a conventional Gas Metal Arc Welding (GMAW) welding process. The problem is formulated as:

"How should a classification algorithm be, which, based on, features extracted from the system measurements, can reliably predict when the GMAW Short-Circuit welding process is about to free burn, in the rupture period, in the ending of the short-circuit phase."

In in this problem statement lies multiple sub-problems, which need to be addressed to answer the question put forth.

The problem of predicting the start of free burn in the short-circuiting phase of the welding process was formulated as a binary classification problem. In this formulation, data samples from the short-circuit phase of the welding process would classified as "Rupture" if they belonged to the last 10 data samples of the short-circuit phase, the rupture period, or as "Short" for the remaining data samples from the prior period of the short-circuit phase. The problem of estimating whether the welding process is in arc phase or short-circuit phase was not considered in the scope of this project work.
To establish a base for the classification algorithm, additional feature data were extracted from the provided current and voltage welding system measurements, samples throughout the welding process. The measurement data examined at this point were all acquired from welding processes set to the same welding program and settings. Since the data samples from to the rupture periods in the given measurement data is not sufficiently distinguishable from the data samples from the prior part of the short-circuit phases, this feature extraction is a necessary step. First, a set of seven

sample-by-sample features, meaning these features are based on one measurement sample at a time, was extracted. Secondly, five statistical features were extracted, using a sequence of data samples in a sliding window over the time series of measurement data samples as a basis. Lastly, a feature was extracted comprised of the correlation coefficient to the measurement signal computed for a template rupture period signal slid over the time series of measurement data. The features above were all extracted from the voltage measurement data. The relevance of these features to the targeted classes in the classification problem, classes "Rupture and "Short respectively, and their redundancy to each other was quantified and ranked based on feature importance scores computed by means of the Minimum Redundancy Maximum Relevance (mRMR) feature selection method. Here it was concluded the importance of the extracted features varied significantly, but no features were excluded from use for the classification problem.
In addition to the prior features, it was investigated whether the cross-sectional area of the metal bridge, formed when the welding electrode short-circuit with the welding workpiece, could be estimated and added as a feature, basing the estimation on a simple model of the evolution of the cross-sectional area and a model of the control voltage in the short-circuit phase of the welding process. This was done with an extended kalman filter as the model for the control voltage was non-linear. The resulting estimate of the cross-sectional area was not realistic, as it estimated the bridge to collapse far earlier than then the welding process entered the rupture period. Instead it was found that the residual, based on the error between the uncorrected estimate of the control voltage and the measurement of the control voltage was a suitable feature to add for the classification.

The overarching problem, development of a classification algorithm reliably applicable to the binary classification problem stated earlier, was successfully solved, based on feature extracted. Classification models such as the Naive Bayes classifier, the K-Nearest Neighbour classifier, Linear and Quadratic Discriminant Analysis classifier, and lastly the Support Vector Machines classifier, were trained with feature data. The performance of the classification models, set to standard hyperparameters as of MATLAB, was evaluated in cross-validation in terms of binary classification measures, True Positive Rate, True Negative Rate, Positive Prediction Value, and Accuracy, and it was found, the K-Nearest Neighbour classification model performed the best.
In optimisation of the hyperparameters for the K-Nearest Neighbour model, the best performing distance function and number of neighbours was found to be the Mahalanobis distance using 1 neighbour. In an unbiased test of the optimised K-Nearest Neighbour model, on hitherto unused data but still from the same welding program and setting, it was concluded that the performance be significantly worse, especially the Positive Prediction value was low, indicating a large number of false positives, i.e.

data sample4s not belonging to the "Rupture" class being classified as such. While the True Positive Rate was also lower than desired it was found the low rate mainly pertained to samples from early in the rupture period, and that the classification performance increased significantly halfway into the rupture period.
By testing the model on measurement data from different welding programs and settings, it became clear that the feature values extracted from one measurement data set, from a particular program and setting, where not equivalent to the feature values extracted from another measurement data set from a different program and setting. In order to obtain a classifier able to reliably predict the correct class for a given measurement data sample, regardless of program and setting, the classifier needs to be trained on data with the given program and setting. To incorporate data belonging to different programs into one classification model, an additional feature was created based on the program label, with the aim this would, in the classification model, separate data from the various programs according to program. However even then, performance was for data from some welding programs, not on the level of performance seen in the first test, where the classification model was trained on data from the same welding program as tested on, implying the features extracted may not be equally relevant for classification for all the tested welding programs.

In conclusion through the work presented in this thesis it is found that a classification algorithm, the K-Nearest Neighbour classifier, based on relevant feature data can be developed to predict when free burn is about to happen in a conventional GMAW Short-Circuit welding process. However for certain welding programs and settings, where the feature data here presented is less relevant to the classification problem, the prediction is less reliable than desired.

Chapter 7

Discussion

This chapter discusses the thesis results, along with thoughts on future improvements to the system. The thoughts presented in this chapter, are based on the findings in this thesis and any recommendations for improvement are purely speculative.

7.1 Thesis Results

As is evident from the conclusion in Chapter 6 and from Section 5.3, it was possible to some extend to predict when the welding process was about to free burn. This, however, was possible only if the K-Nearest Neighbor classification algorithm was trained with data, originating from a welding scenario similar to the one in question for the classification and prediction. When the training data origin differed in welding program and setting, the number of correct predictions fell drastically. To overcome this, welding measurement data from various welding programs and settings was included as training data for the K-Nearest Neighbour model, and in an attempt to separate data from the various programs from each other in the model, the program label was added as a feature. This saw an increase in the interprogram performance, however, it is possible that also adding the welding setting as a feature would improve performance even further, especially on the welding program and setting combinations where the performance was far less than desired.

Another aspect to note is the change in the amount of true positives as the welding process moves towards a rupture. In the cases where the classification algorithm had been trained with data from similar origin as the welding measurements at hand, the rate of true positives increased linearly. In the other cases, where the classification algorithm had been trained with data from dissimilar origin, the rate of true positives increased exponentially, from a rather low start value, as seen on Figure 5.9 and 5.10. This might indicate the extracted features are relevant for the different welding programs, but the absolute feature values are not transferable as classification parameters from one welding scenario to another. A possible solution could be to normalise all data, both training and testing data, as this might reduce the variations in data value magnitude between data from different welding programs. For practical applications, the normalisation should then be based on the feature values of the training data.

A different solution would be to identify and extract features with values that are universal for all welding programs and independent of system settings. The cross sectional area of the metal bridge is an example of a feature that retains its properties for different programs, since the neck of the metal bridge is always small just before a rupture occurs. In the suggested approach, obtain for an estimate suitable as a feature for use in classification, a new state model would have to be developed, as the one developed in this thesis did not produce realistic results. The kalman filter would have to be tuned for each welding electrode type, as the electrodes varies in electrode diameter and material composition. And, the progressing over time in the decrease of the cross-sectional area may differ depending on electrode type to a degree to prevent the use of the same model for all welding scenarios. That is, this might turn out a non-universal feature.

In Chapter 5, Section 5.1.5, the K-Nearest Neighbour model was determined to be as the best performing classification model evaluated on binary classification performance measures, based on cross-validation of training data, in comparison with a range of other classification models. After selecting from this the K-Nearest Neighbour model, as the model to use further in this project work the model was optimised by determining the optimal hyperparameters. It is plausible the hyperparameter optimisation should have taken place before the classification comparison, and for each of the compared classification models, as it is not given the comparison conducted based on the default hyperparameters in the MATLAB software would show an accurate comparison. Optimising the models before comparison would ensure the selected model would be the best performing model among the optimised versions of the models. Perhaps in this order of operations, the Support Vector Machines or the Linear Discriminant Analysis, which generally came second and third, respectively, in the score ranking in the comparison conducted, would then be performing better.

In the classification model comparison, Section 5.1.5, Support Vector Machines was the second best performing model, impeded only by a low True Positive Rate, while having a higher Positive Prediction Value than the K-nearest Neighbour. It is not out of question, the low TPR pertained only to the early samples in the rupture periods, while the TPR for samples in the latter part of the rupture periods could be acceptable. Given this is true, the Support Vector Machines classification model might with its lower amount of false positives, possibly be a model better fit for the classification problem.

7.2 Welding System Improvements

As mentioned in Section 7.1, the closer the welding process came to a rupture, the more reliable the classification became.If the welding control system performance can be improved, so that the prediction horizon, the time needed from prediction till occurrence of a rupture, could be lowered, from 5 samples times to e.g. 3 sample times i.e. from 0.1ms to 0.06ms, this would according to the results in Section 5.3, in some cases result in more than a doubling of the rate of true positives, without any changes to the classification algorithm. Such a control system upgrade, would also make a possible real time implementation of the classification algorithm more plausible, as the control system in its current state requires considerable computation time for the classification model suggested. One loop of the kalman filter alone takes up to 0.191 ms to complete, while measurement samples are taken every 0.02ms by the Digital Signal Processing unit of the control system, so a real time implementation would require either an optimisation of the code base or a hardware upgrade. This is stressed by consideration of the time to perform the actual classification, which with a K-Nearest Neighbour model takes a substantial amount of time. Other classification models which employ a decision function, such as Linear Discriminant Analysis or Support Vector Machines, may perform the classification task faster.

7.3 Future Work

The work presented in this thesis makes use mainly of machine learning algorithms to predict ruptures. A different strategy could be to approach the problem as a fault-detection problem. The idea is to improve the state model for the welding system, specifically for the periods when it is in short circuit mode, and then use this model to generate a residual, similar to what was done in Chapter 4. The residual would be the difference between the real system output and the model output. The idea is that as the system moves toward a rupture, the residual may exhibit a significant change, as the model would be further from its operating point. This is however only speculations, based on the results of Chapter 4, further development and tests would have to be done in order to determine the viability of this approach. One benefit of this approach is that the rupture prediction system then only needs tuning of a state observer, not training of a classifier, before it can produce results, depending on complexity, needed of the component to detect the residual change.

For the approach taken in the work here presented, to seek improvement of the reliability of the prediction, different classification algorithms or combinations of algorithms would have to be tested. If the data from the welding process is normalized, this might result in a different classification algorithm performing best. An increase or change in the set of features for the classification might produce a similar result.

Determination of whether a classification algorithm is usable lies for a large part not only in a high amount of true positives but also in a low amount of false positives. The low amount of false positives are important, because if the current is turned down prematurely, it results in a bad weld, which requires more work to correct than cleaning up splatter. It could be interesting to look into a method for checking whether the current was turned down prematurely, so that immidiately the current could be turned back up in order to avoid a bad weld.

Bibliography

[1] Migatronic A/S. Last accessed on 06-02-2020. URL: `https://www.migatronic.com/`.

[2] Howard B. Cary. *Modern welding technology.* 2nd ed. Prentice Hall, 1989. ISBN: 0-13-599283-4.

[3] Jesper Sandberg Thomsen. *Advanced control methods for optimization of arc welding.* Department of control engineering, Aalborg university, 2004.

[4] TWI Global. Last accessed on 06-02-2020. URL: `https://www.twi-global.com/technical-knowledge/faqs/faq-what-is-mig-mag-welding`.

[5] Annette O'Brien. *Welding Handbook, Welding Processes, Part 1.* 9th ed. American Welding Society, 2004. ISBN: 978-0-87171-729-0.

[6] Lincoln Electric. Last accessed on 06-02-2020. URL: `https://www.lincolnelectric.com/assets/global/Products/Consumable_MIGGMAWWires-SuperArc-SuperArcL-50/c4200.pdf`.

[7] Young-Cheol Jeong, Yoon-Gyo Jung, and Young-Tae Cho. "Simple monitoring of welding spatter for quantification and observation using a mobile phone". In: *Advances in Mechanical Engineering* 9.9 (2017), p. 1687814017725247. DOI: `10.1177/1687814017725247`.

[8] Bruce D. DeRuntz. "Assessing the Benefits of Surface Tension Transfer® Welding to Industry". In: *Industrial Technology* 19.4 (2003).

[9] John Norrish and Dominic Cuiuri. "The controlled short circuit GMAW process: a tutorial". In: *Journal of Manufacturing Processes* 16.1 (2014), pp. 86–92.

[10] P Kah, R Suoranta, and J Martikainen. "Advanced gas metal arc welding processes". In: *The International Journal of Advanced Manufacturing Technology* 67.1-4 (2013), pp. 655–674.

[11] Mads Jensen, Rasmus Pedersen, and Jeppe Juel Petersen. *Melting rate estimation and control for gas metal arc welding.* Department of control engineering, Aalborg university, 2010.

[12] Shre Chatterjee et al. “Exploring Strategies for Classification of External Stimuli Using Statistical Features of the Plant Electrical Response”. eng. In: *arXiv.org* 12.104 (2016), p. 20141225. ISSN: 17425662. URL: `http://search.proquest.com/docview/2080419927/`.

[13] DN Joanes and CA Gill. “Comparing measures of sample skewness and kurtosis”. In: *Journal of the Royal Statistical Society: Series D (The Statistician)* 47.1 (1998), pp. 183–189.

[14] Bo Hjorth. “EEG analysis based on time domain properties”. In: *Electroencephalography and clinical neurophysiology* 29.3 (1970), pp. 306–310.

[15] Lennart Råde and Bertil Westergren. *Beta [beta] mathematics handbook.* 1990.

[16] C Ding and H Peng. “Minimum redundancy feature selection from microarray gene expression data”. eng. In: *Computational Systems Bioinformatics. CSB2003. Proceedings of the 2003 IEEE Bioinformatics Conference. CSB2003.* IEEE, 2003, pp. 523–528. ISBN: 0769520006.

[17] Matlab. *fscmrmr.* Last accessed on 06-03-2020. URL: `https://www.mathworks.com/help/stats/fscmrmr.html`.

[18] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalizations.* CRC press, 2015.

[19] Matlab. *Train multiclass naive Bayes model.* Last accessed on 12-03-2020. URL: `https://www.mathworks.com/help/stats/fitcnb.html`.

[20] “Sequence Based Prediction of DNA-Binding Proteins Based on Hybrid Feature Selection Using Random Forest and Gaussian Naive Bayes: e86703”. eng. In: *PLoS ONE* 9.1 (2014), e86703. ISSN: 1932-6203. URL: `http://search.proquest.com/docview/1505329403/`.

[21] Matthias Kohl. “Performance Measures in Binary Classification”. In: *International Journal of Statistics in Medical Research* 1 (Jan. 2012), pp. 79–81. DOI: `10.6000/1929-6029.2012.01.01.08`.

[22] “A signal pattern matching and verification method using interval means cross correlation and eigenvalues in the nuclear power plant monitoring systems”. eng. In: *Annals of Nuclear Energy* 29.15 (2002), pp. 1795–1807. ISSN: 0306-4549.

[23] Sheldon M. Ross. “Chapter 2 - DESCRIPTIVE STATISTICS”. In: *Introduction to Probability and Statistics for Engineers and Scientists (Fourth Edition).* Ed. by Sheldon M. Ross. Fourth Edition. Boston: Academic Press, 2009, pp. 9 –53. ISBN: 978-0-12-370483-2. DOI: `https://doi.org/10.1016/B978-0-12-370483-2.00007-2`. URL: `http://www.sciencedirect.com/science/article/pii/B9780123704832000072`.

[24] Finn Aakre Haugen. *Automatic Control.* self published (pre relese copy), 2019.

[25] Christian Gebbe et al. “Feature Extraction and Classification of the Electric Current Signal of an Induction Motor for Condition Monitoring Purposes”. eng. In: *Applied Mechanics and Materials* 856 (2017), pp. 244–251. ISSN: 1660-9336.

[26] Christopher M Bishop. *Pattern recognition and machine learning.* Information science and statistics. New York, NY: Springer, 2006. URL: `https://cds.cern.ch/record/998831`.

[27] Chih-wei Hsu, Chih-chung Chang, and Chih-Jen Lin. “A Practical Guide to Support Vector Classification Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin”. In: (Nov. 2003).

[28] Matlab. *MATLAB.* Last accessed on 19-05-2020. URL: `https://www.mathworks.com/products/matlab.html`.

[29] Matlab. *Fit k-nearest neighbor classifier.* Last accessed on 19-05-2020. URL: `https://www.mathworks.com/help/stats/fitcknn.html`.

[30] Peter Kralchevsky and Kuniaki Nagayama. *Particles at Fluid Interfaces and Membranes.* Elsevier, 2001.

Appendix A

Project Proposal

Here is a copy of the original project proposal by Migatronic A/S.

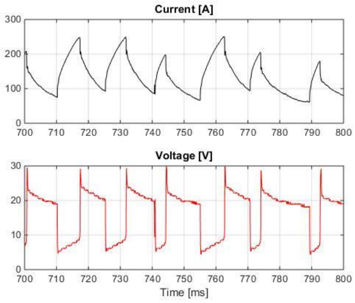
Weld process free burn estimator

For some years there has been a cooperation between the welding company Migatronic A/S and Aalborg University, which has resulted in several student-projects and an industrial PhD project. Migatronic has been manufacturing welding machines in Denmark since 1970, and today employs 300 people worldwide, with production facilities in Denmark, China and India.

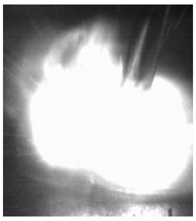
As the technology in the welding machines has evolved from simple power transformers to sophisticated microcontroller systems, more complex control of the process has been made available. Today's inverter technology, using a powerful DSP, assists even "unskilled" welders in creating high-quality welds – where the quality of a weld earlier was solely dependent on the skill of the welder.

The welding current is controlled with 50 kHz, all decisions are made based on measurements of weld current and voltage. To improve the welding it is crucial to have knowledge about different states in the weld process, preferable based on only the two before mentioned measurements. A number of different weld processes exists, each having their benefits and drawbacks. One such process, primary use for thin plates, is called short circuit welding. In this process a wire is fed forward to a weld pool, and an accurately controlled current, through the wire, causes the process to alternate between a short circuit and an arc between the wire tip and weld pool. The drawback of the short circuit process is however, a large explosion at the moment of free burn; that is where the process changes from being short circuited to ignite an arc.

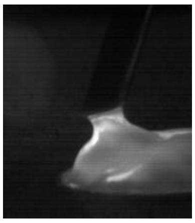
If an estimator could inform when the free burn is about to happen (before it happens), actions could be taken, to avoid a large explosion. In figure 1, an example of some measurements is shown. In the bottom the voltage is shown, where low voltage equals short circuit and high voltage equals arc. It is fairly simple to separate these two states, but the challenge is to estimate the free burn before it happens, and to make this estimator robust to an ever changing system.



Figur 1: Measurements from short circuit welding



Figur 2: Arc



Figur 3: Short circuit

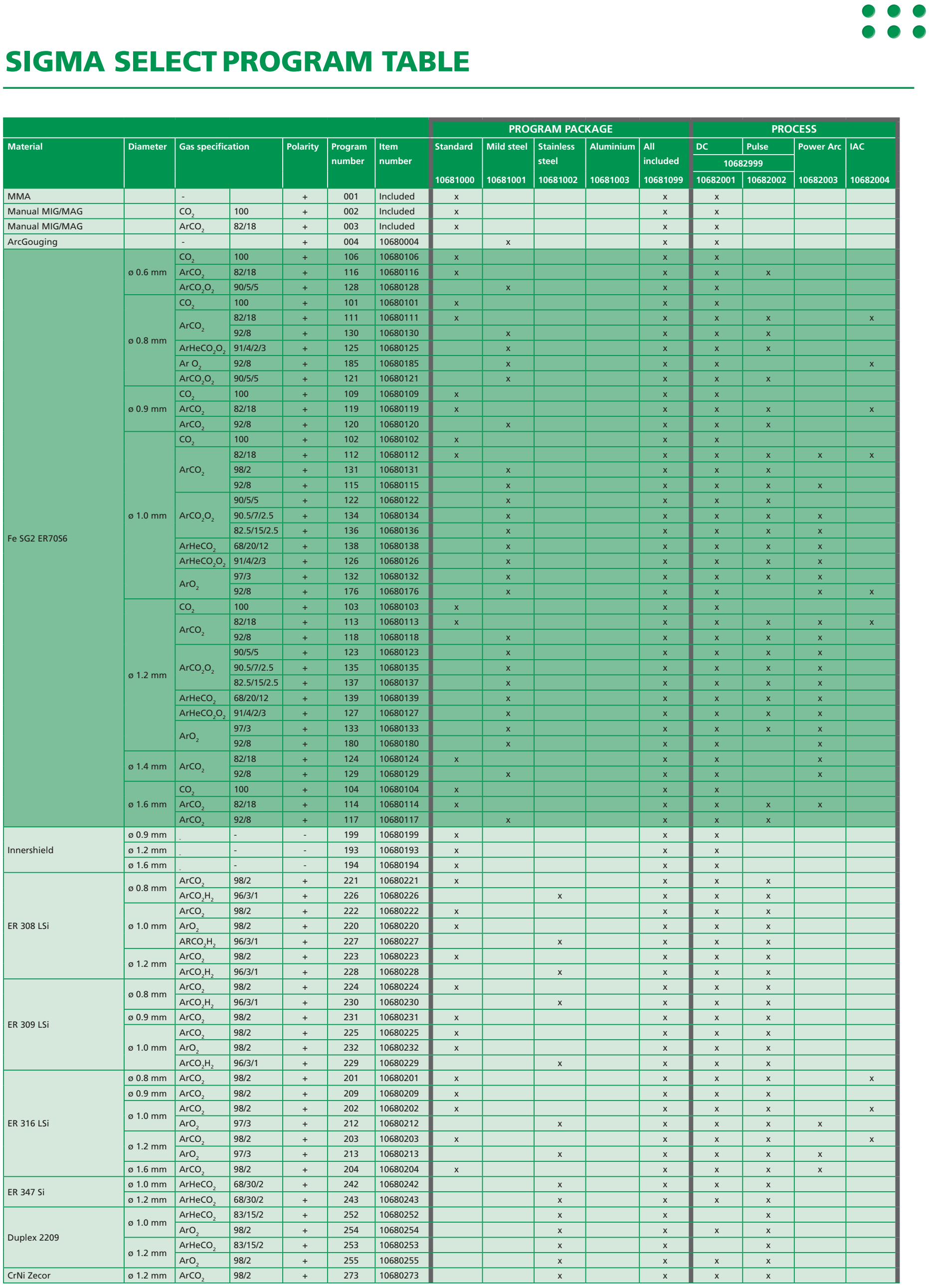
Project proposal:
Develop an estimator that, based on measurements of the weld current and voltage, can estimate when the weld process is about to free burn.
The following should be considered:

- Robustness for different settings of current / wire speed
- Robustness for different wire materials
- Ability to adapt to changing the setup (changing cable lengths, coiling the cables, etc.)
- It should be possible to implement the estimator and run it in real time

Jesper L. Skovfo, Control System Engineer, M.Sc.E.E., Migatronic A/S, JLS@migatronic.dk
Steffan K. Ovedal, Control System Engineer, M.Sc.E.E., Migatronic A/S, SKO@migatronic.dk
René Petersen, Control System Engineer, Ph.D., Migatronic A/S, RPE@migatronic.dk

Appendix B

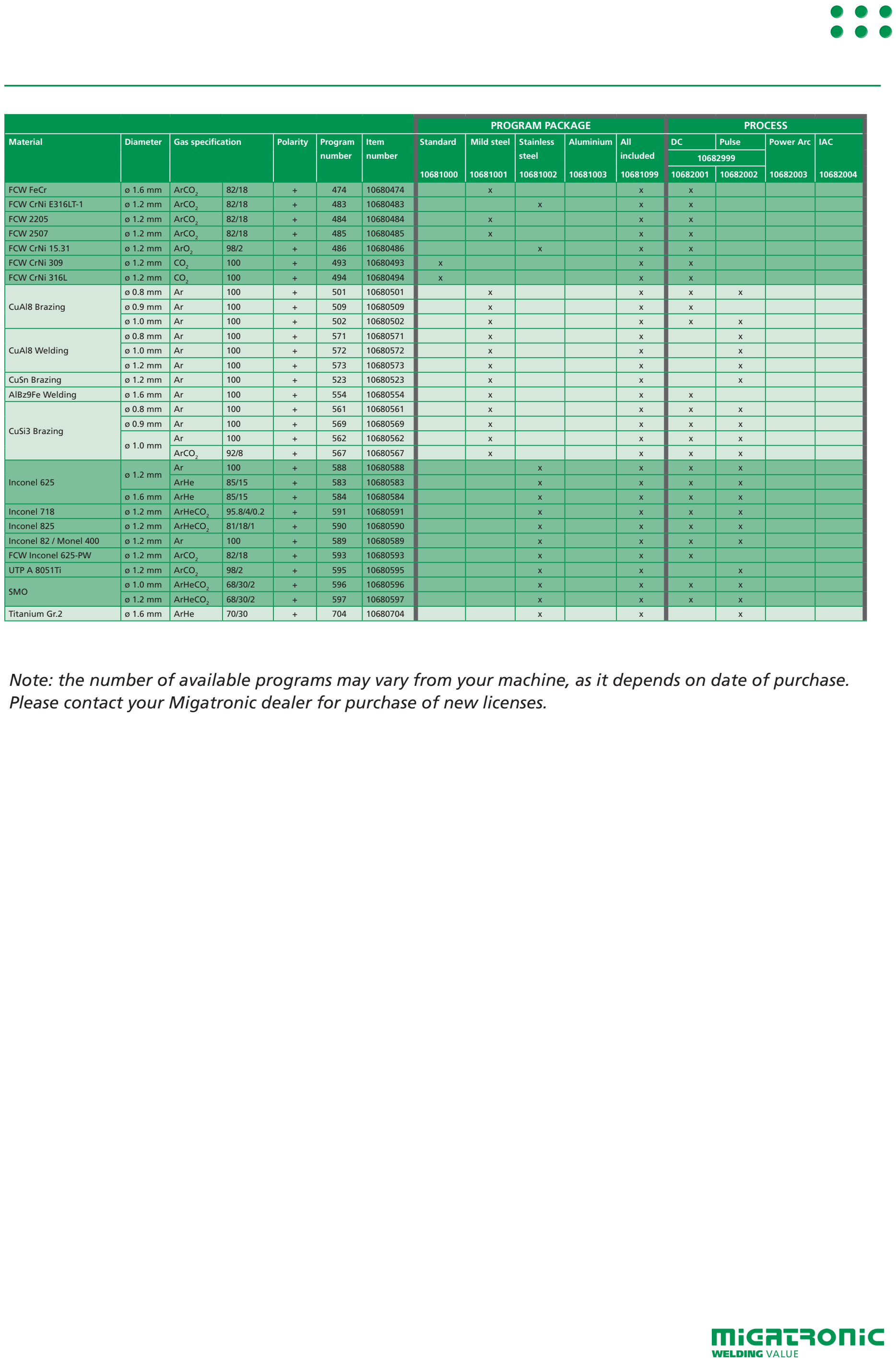
Sigma Select Program Table



SIGMA SELECT PROGRAM TABLE

							PROGRAM PACKAGE					PROCESS			
Material	Diameter	Gas specification		Polarity	Program number	Item number	Standard	Mild steel	Stainless steel	Aluminium	All included	DC	Pulse	Power Arc	IAC
												10682999			
							10681000	10681001	10681002	10681003	10681099	10682001	10682002	10682003	10682004
MMA		-		+	001	Included	x				x	x			
Manual MIG/MAG		CO$_2$	100	+	002	Included	x				x	x			
Manual MIG/MAG		ArCO$_2$	82/18	+	003	Included	x				x	x			
ArcGouging		-		+	004	10680004		x			x	x			
Fe SG2 ER70S6	ø 0.6 mm	CO$_2$	100	+	106	10680106	x				x	x			
		ArCO$_2$	82/18	+	116	10680116	x				x	x	x		
		ArCO$_2$O$_2$	90/5/5	+	128	10680128		x			x	x			
	ø 0.8 mm	CO$_2$	100	+	101	10680101	x				x	x			
		ArCO$_2$	82/18	+	111	10680111	x				x	x	x		x
			92/8	+	130	10680130		x			x	x	x		
		ArHeCO$_2$O$_2$	91/4/2/3	+	125	10680125		x			x	x	x		
		Ar O$_2$	92/8	+	185	10680185		x			x	x			x
		ArCO$_2$O$_2$	90/5/5	+	121	10680121		x			x	x	x		
	ø 0.9 mm	CO$_2$	100	+	109	10680109	x				x	x			
		ArCO$_2$	82/18	+	119	10680119	x				x	x	x		x
		ArCO$_2$	92/8	+	120	10680120		x			x	x	x		
	ø 1.0 mm	CO$_2$	100	+	102	10680102	x				x	x			
		ArCO$_2$	82/18	+	112	10680112	x				x	x	x	x	x
			98/2	+	131	10680131		x			x	x	x		
			92/8	+	115	10680115		x			x	x	x	x	
		ArCO$_2$O$_2$	90/5/5	+	122	10680122		x			x	x	x		
			90.5/7/2.5	+	134	10680134		x			x	x	x	x	
			82.5/15/2.5	+	136	10680136		x			x	x	x	x	
		ArHeCO$_2$	68/20/12	+	138	10680138		x			x	x	x	x	
		ArHeCO$_2$O$_2$	91/4/2/3	+	126	10680126		x			x	x	x	x	
		ArO$_2$	97/3	+	132	10680132		x			x	x	x	x	
			92/8	+	176	10680176		x			x	x		x	x
	ø 1.2 mm	CO$_2$	100	+	103	10680103	x				x	x			
		ArCO$_2$	82/18	+	113	10680113	x				x	x	x	x	x
			92/8	+	118	10680118		x			x	x	x	x	
		ArCO$_2$O$_2$	90/5/5	+	123	10680123		x			x	x	x	x	
			90.5/7/2.5	+	135	10680135		x			x	x	x	x	
			82.5/15/2.5	+	137	10680137		x			x	x	x	x	
		ArHeCO$_2$	68/20/12	+	139	10680139		x			x	x	x	x	
		ArHeCO$_2$O$_2$	91/4/2/3	+	127	10680127		x			x	x	x	x	
		ArO$_2$	97/3	+	133	10680133		x			x	x	x	x	
			92/8	+	180	10680180		x			x	x		x	
	ø 1.4 mm	ArCO$_2$	82/18	+	124	10680124	x				x	x		x	
			92/8	+	129	10680129		x			x	x		x	
	ø 1.6 mm	CO$_2$	100	+	104	10680104	x				x	x			
		ArCO$_2$	82/18	+	114	10680114	x				x	x	x	x	
		ArCO$_2$	92/8	+	117	10680117		x			x	x	x		
Innershield	ø 0.9 mm		-	-	199	10680199	x				x	x			
	ø 1.2 mm		-	-	193	10680193	x				x	x			
	ø 1.6 mm		-	-	194	10680194	x				x	x			
ER 308 LSi	ø 0.8 mm	ArCO$_2$	98/2	+	221	10680221	x				x	x	x		
		ArCO$_2$H$_2$	96/3/1	+	226	10680226			x		x	x	x		
	ø 1.0 mm	ArCO$_2$	98/2	+	222	10680222	x				x	x	x		
		ArO$_2$	98/2	+	220	10680220	x				x	x	x		
		ARCO$_2$H$_2$	96/3/1	+	227	10680227			x		x	x	x		
	ø 1.2 mm	ArCO$_2$	98/2	+	223	10680223	x				x	x	x		
		ArCO$_2$H$_2$	96/3/1	+	228	10680228			x		x	x	x		
ER 309 LSi	ø 0.8 mm	ArCO$_2$	98/2	+	224	10680224	x				x	x	x		
		ArCO$_2$H$_2$	96/3/1	+	230	10680230			x		x	x	x		
	ø 0.9 mm	ArCO$_2$	98/2	+	231	10680231	x				x	x	x		
	ø 1.0 mm	ArCO$_2$	98/2	+	225	10680225	x				x	x	x		
		ArO$_2$	98/2	+	232	10680232	x				x	x	x		
		ArCO$_2$H$_2$	96/3/1	+	229	10680229			x		x	x	x		
ER 316 LSi	ø 0.8 mm	ArCO$_2$	98/2	+	201	10680201	x				x	x	x		x
	ø 0.9 mm	ArCO$_2$	98/2	+	209	10680209	x				x	x	x		
	ø 1.0 mm	ArCO$_2$	98/2	+	202	10680202	x				x	x	x		x
		ArO$_2$	97/3	+	212	10680212			x		x	x	x	x	
	ø 1.2 mm	ArCO$_2$	98/2	+	203	10680203	x				x	x	x		x
		ArO$_2$	97/3	+	213	10680213			x		x	x	x	x	
	ø 1.6 mm	ArCO$_2$	98/2	+	204	10680204	x				x	x	x	x	
ER 347 Si	ø 1.0 mm	ArHeCO$_2$	68/30/2	+	242	10680242			x		x	x	x		
	ø 1.2 mm	ArHeCO$_2$	68/30/2	+	243	10680243			x		x	x	x		
Duplex 2209	ø 1.0 mm	ArHeCO$_2$	83/15/2	+	252	10680252			x		x		x		
		ArO$_2$	98/2	+	254	10680254			x		x	x	x		
	ø 1.2 mm	ArHeCO$_2$	83/15/2	+	253	10680253			x		x		x		
		ArO$_2$	98/2	+	255	10680255			x		x	x	x		
CrNi Zecor	ø 1.2 mm	ArCO$_2$	98/2	+	273	10680273			x		x	x	x		

							PROGRAM PACKAGE					PROCESS			
Material	Diameter	Gas specification		Polarity	Program number	Item number	Standard	Mild steel	Stainless steel	Aluminium	All included	DC	Pulse	Power Arc	IAC
												10682999			
							10681000	10681001	10681002	10681003	10681099	10682001	10682002	10682003	10682004
Super Duplex 2509	ø 1.0 mm	ArCO$_2$	98/2	+	262	10680262			x		x		x		
		ArHeCO$_2$	83/15/2	+	256	10680256			x		x		x		
			68/30/2		274	10680274			x		x		x		
		ArHeO$_2$	69/30/1	+	267	10680267			x		x		x		
		ArO$_2$	98/2	+	264	10680264			x		x	x	x		
	ø 1.2 mm	ArCO$_2$	98/2	+	263	10680263			x		x		x		
		ArHeCO$_2$	83/15/2	+	257	10680257			x		x		x		
			68/30/2		275	10680275			x		x		x		
		ArO$_2$	98/2	+	265	10680265			x		x	x	x		
AlMg5 ER5356	ø 0.8 mm	Ar	100	+	311	10680311	x				x	x			
	ø 0.9 mm	Ar	100	+	319	10680319	x				x	x	x		
	ø 1.0 mm	Ar	100	+	312	10680312	x				x	x	x		
		ArHe	70/30	+	316	10680316				x	x	x	x		
			50/50	+	332	10680332				x	x	x	x		
	ø 1.2 mm	Ar	100	+	313	10680313	x				x	x	x		
		ArHe	70/30	+	317	10680317				x	x	x	x		
			50/50	+	333	10680333				x	x	x	x		
	ø 1.6 mm	Ar	100	+	314	10680314	x				x	x	x		
		ArHe	70/30	+	318	10680318				x	x	x	x		
AlMg4.5MnZr ER5087	ø 1.2 mm	Ar	100	+	321	10680321				x	x	x	x		
AlMg4.5 ER5183	ø 1.2 mm	ArHe	20/80	+	323	10680323				x	x	x	x		
	ø 1.6 mm	ArHe	70/30	+	328	10680328				x	x	x	x		
AlMg3Mn ER5554	ø 0.9 mm	Ar	100	+	369	10680369				x	x	x			
AlSi5 ER4043	ø 0.9 mm	Ar	100	+	359	10680359	x				x	x			
	ø 1.0 mm	Ar	100	+	352	10680352	x				x	x	x		
		ArHe	70/30	+	356	10680356				x	x	x	x		
	ø 1.2 mm	Ar	100	+	353	10680353	x				x	x	x		
		ArHe	70/30	+	357	10680357				x	x	x	x		
	ø 1.6 mm	Ar	100	+	354	10680354	x				x	x	x		
		ArHe	70/30	+	358	10680358				x	x	x	x		
Al99.5 ER1100	ø 1.2 mm	Ar	100	+	363	10680363				x	x	x	x		
	ø 1.6mm	Ar	100	+	364	10680364				x	x	x	x		
AlSi12 ER4047	ø 1.0mm	Ar	100	+	372	10680372	x				x	x	x		
	ø 1.2 mm	Ar	100	+	373	10680373	x				x	x	x		
AlSi10 MG ER4046	ø 1.2 mm	ArHe	50/50	+	383	10680383				x	x	x	x		
AlSi10Cu4 ER4145	ø 1.2 mm	Ar	100	+	393	10680393				x	x	x			
FCW FeRutil 215	ø 1.0 mm	ArCO$_2$	82/18	+	402	10680402	x				x	x			
	ø 1.2 mm	ArCO$_2$	82/18	+	403	10680403	x				x	x			
	ø 1.6 mm	ArCO$_2$	82/18	+	404	10680404	x				x	x			
		CO$_2$	100	+	444	10680444	x				x	x			
FCW FeRutil 217	ø 1.2 mm	ArCO$_2$	82/18	+	453	10680453	x				x	x			
FCW FeRutil 713R	ø 1.2 mm	ArCO$_2$	82/18	+	454	10680454		x			x	x			
FCW FeRutil 15.14	ø 1.2 mm	CO$_2$	100	+	443	10680443		x			x	x			
FCW FeRutil Nittetsu SF-1A	ø 1.2 mm	ArCO$_2$	82/18	+	407	10680407		x			x	x			
FCW FeRutil Nittetsu SF-1E	ø 1.2 mm	CO$_2$	100	+	448	10680448		x			x	x			
FCW FeRutil Nittetsu SF-3AM	ø 1.2 mm	ArCO$_2$	82/18	+	408	10680408		x			x	x			
	ø 1.4 mm	ArCO$_2$	82/18	+	409	10680409		x			x	x			
FCW FeRutil PZ6111	ø 1.6 mm	ArCO$_2$	82/18	+	405	10680405		x			x	x			
FCW FeRutil PZ6111-HS	ø 1.6 mm	CO$_2$	100	+	446	10680446		x			x	x			
FCW FeRutil 6114	ø 1.2 mm	ArCO$_2$	82/18	+	458	10680458		x			x	x			
FCW FeRutil PZ6114S	ø 1.2 mm	CO$_2$	100	+	447	10680447		x			x	x			
FCW FeRutil PZ6138	ø 1.2 mm	ArCO$_2$	82/18	+	457	10680457		x			x	x			
FCW FeRutil DWA 55E	ø 1.2 mm	ArCO$_2$	82/18	+	451	10680451		x			x	x			
FCW FeRutil Tubrod 15.14	ø 1.2 mm	ArCO$_2$	82/18	+	452	10680452		x			x	x			
		CO$_2$	100	+	495	10680495		x			x	x			
FCW FeMetal 6104	ø 1.2 mm	ArCO$_2$	82/18	+	456	10680456		x			x	x			
FCW FeMetal MX 100T	ø 1.2 mm	ArCO$_2$	82/18	+	429	10680429		x			x	x			
FCW FeMetal 115	ø 1.2 mm	ArCO$_2$	82/18	+	423	10680423	x				x	x	x	x	
	ø 1.4 mm	ArCO$_2$	82/18	+	425	10680425	x				x	x			
	ø 1.6 mm	ArCO$_2$	82/18	+	424	10680424	x				x	x			
		CO$_2$	100	+	445	10680445	x				x	x			
FCW FeMetal 235M	ø 1.2 mm	ArCO$_2$	82/18	+	420	10680420		x			x	x	x		
FCW FeMetal 710M/742M	ø 1.0 mm	ArCO$_2$	82/18	+	426	10680426		x			x	x			
FCW FeMetal 710M	ø 1.2 mm	ArCO$_2$	82/18	+	427	10680427		x			x	x			
FCW FeMetal 742M	ø 1.2 mm	ArCO$_2$	82/18	+	428	10680428		x			x	x			
FCW FeMetal MC-RS	ø 1.2 mm	ArCO$_2$	82/18	+	418	10680418		x			x	x			
FCW FeMetal 700 T-MC	ø 1.2 mm	ArCO$_2$	92/8	+	461	10680461		x			x	x	x		
FCW FeMetal HL 51-MC	ø 1.0 mm	ArCO$_2$	92/8	+	462	10680462		x			x	x	x		
FCW FeMetal HL 51-MC	ø 1.2 mm	ArCO$_2$	92/8	+	463	10680463		x			x	x	x		
FCW FeMetal 700MC	ø 1.0 mm	ArCO$_2$	92/8	+	464	10680464		x			x	x	x		
FCW FeMetal 700MC	ø 1.2 mm	ArCO$_2$	92/8	+	465	10680465		x			x	x	x		
FCW FeBasic 515	ø 1.2 mm	ArCO$_2$	82/18	-	413	10680413		x			x	x			
FCW FeBasic 6130	ø 1.4 mm	CO$_2$	100	+/-	435	10680435		x			x	x			
		ArCO$_2$	82/18	+/-	415	10680415		x			x	x			
FCW FeBasic 6125	ø 1.0 mm	ArCO$_2$	82/18	-	417	10680417		x			x	x			
FCW FeBasic 15.00	ø 1.4 mm	CO$_2$	100	+/-	433	10680433		x			x	x			



							PROGRAM PACKAGE					PROCESS			
Material	Diameter	Gas specification		Polarity	Program number	Item number	Standard	Mild steel	Stainless steel	Aluminium	All included	DC	Pulse	Power Arc	IAC
												10682999			
							10681000	10681001	10681002	10681003	10681099	10682001	10682002	10682003	10682004
FCW FeCr	ø 1.6 mm	$ArCO_2$	82/18	+	474	10680474		x			x	x			
FCW CrNi E316LT-1	ø 1.2 mm	$ArCO_2$	82/18	+	483	10680483			x		x	x			
FCW 2205	ø 1.2 mm	$ArCO_2$	82/18	+	484	10680484		x			x	x			
FCW 2507	ø 1.2 mm	$ArCO_2$	82/18	+	485	10680485		x			x	x			
FCW CrNi 15.31	ø 1.2 mm	ArO_2	98/2	+	486	10680486			x		x	x			
FCW CrNi 309	ø 1.2 mm	CO_2	100	+	493	10680493	x				x	x			
FCW CrNi 316L	ø 1.2 mm	CO_2	100	+	494	10680494	x				x	x			
CuAl8 Brazing	ø 0.8 mm	Ar	100	+	501	10680501		x			x	x	x		
	ø 0.9 mm	Ar	100	+	509	10680509		x			x	x			
	ø 1.0 mm	Ar	100	+	502	10680502		x			x	x	x		
CuAl8 Welding	ø 0.8 mm	Ar	100	+	571	10680571		x			x		x		
	ø 1.0 mm	Ar	100	+	572	10680572		x			x		x		
	ø 1.2 mm	Ar	100	+	573	10680573		x			x		x		
CuSn Brazing	ø 1.2 mm	Ar	100	+	523	10680523		x			x		x		
AlBz9Fe Welding	ø 1.6 mm	Ar	100	+	554	10680554		x			x	x			
CuSi3 Brazing	ø 0.8 mm	Ar	100	+	561	10680561		x			x	x	x		
	ø 0.9 mm	Ar	100	+	569	10680569		x			x	x	x		
	ø 1.0 mm	Ar	100	+	562	10680562		x			x	x	x		
		$ArCO_2$	92/8	+	567	10680567		x			x	x	x		
Inconel 625	ø 1.2 mm	Ar	100	+	588	10680588			x		x	x	x		
		ArHe	85/15	+	583	10680583			x		x	x	x		
	ø 1.6 mm	ArHe	85/15	+	584	10680584			x		x	x	x		
Inconel 718	ø 1.2 mm	$ArHeCO_2$	95.8/4/0.2	+	591	10680591			x		x	x	x		
Inconel 825	ø 1.2 mm	$ArHeCO_2$	81/18/1	+	590	10680590			x		x	x	x		
Inconel 82 / Monel 400	ø 1.2 mm	Ar	100	+	589	10680589			x		x	x	x		
FCW Inconel 625-PW	ø 1.2 mm	$ArCO_2$	82/18	+	593	10680593			x		x	x			
UTP A 8051Ti	ø 1.2 mm	$ArCO_2$	98/2	+	595	10680595			x		x		x		
SMO	ø 1.0 mm	$ArHeCO_2$	68/30/2	+	596	10680596			x		x	x	x		
	ø 1.2 mm	$ArHeCO_2$	68/30/2	+	597	10680597			x		x	x	x		
Titanium Gr.2	ø 1.6 mm	ArHe	70/30	+	704	10680704			x		x		x		

Note: the number of available programs may vary from your machine, as it depends on date of purchase. Please contact your Migatronic dealer for purchase of new licenses.

MIGATRONIC
WELDING VALUE

Appendix C

High Speed Footage of Metal Bridge

To further the understanding of the formation and collapse of the metal bridge, high speed footage of the GMAW short circuit process is used. The goal of this is to figure out if there is some features/patterns that can be used to describe the development of the bridge. The full high speed clip can be in appendix D.

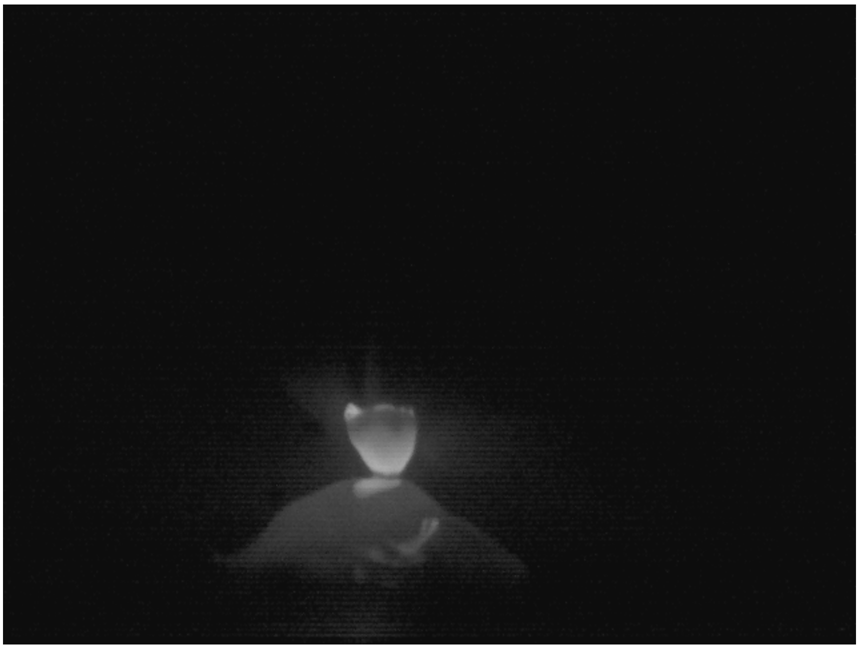


Figure C.1: Still frame of the formation of the metal bridge, just after formation

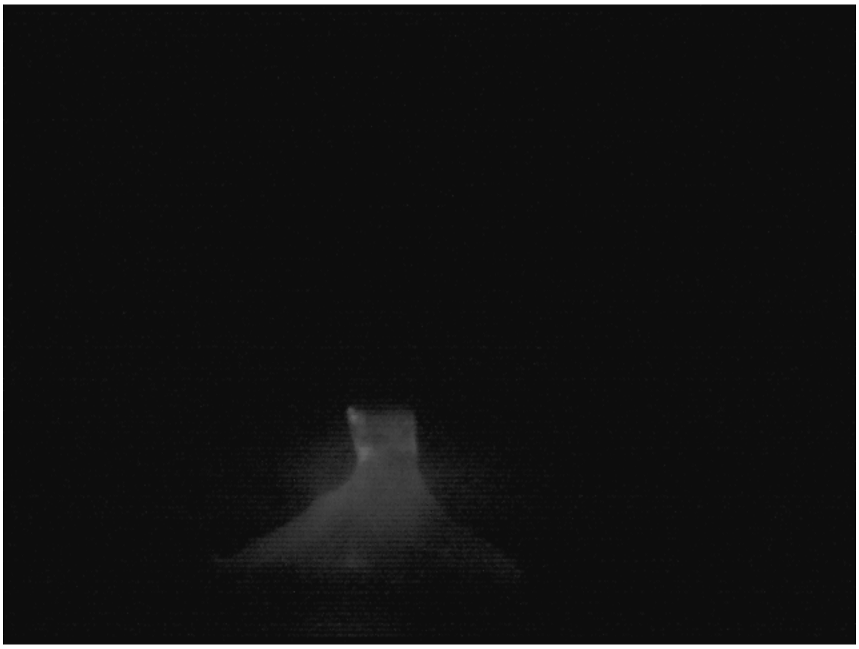


Figure C.2: Still frame of the formation of the metal bridge

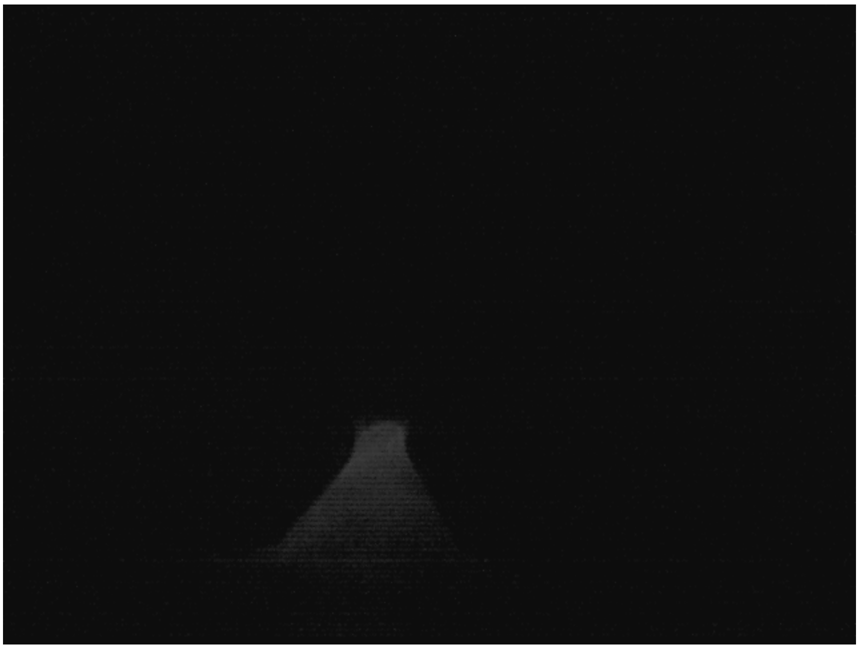


Figure C.3: Still frame of the formation of the metal bridge

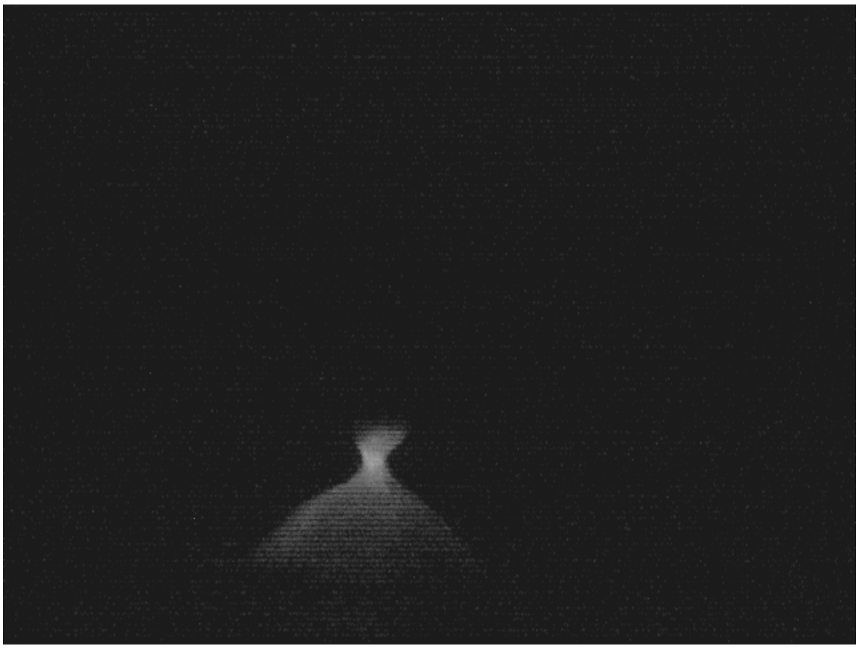


Figure C.4: Still frame of the formation of the metal bridge, just before collapse

Figure C.1 though C.4 shows the development of the metal bridge, from formation to just before collapse. The bridge can be roughly described as a cylinder, where the sides goes from a convex shape to a concave one. The top and bottom diameter of the cylinder does not change, while the middle gets pinched.

The time it takes for the bridge to go though its cycle varies, which correlates with the current and voltage measurements. The time between bridge formation is also not constant, due in turn to "waves" or oscillations in the weld pool caused by the collapse of the former bridge. These waves also tend to cause small deformations in the shape of the bridge.

From the high speed footage is it also clear that the pinching of the bridge is not linear. this means that the pinching of the bridge start slow and then the speed increases as the neck gets thinner. The footage also shows that the distance form the electrode to the workpiece remains quit uniform.

As for finding a suitable model for the bridge there are several options and this appendix will cover two of these models: one that is based on the physical properties of the bridge and another that is based on the electrical properties. The first model is to describe the bride as a capillary bridge. A capillary bridge [30] is a term used for a liquid "membrane" that spans the gap between to objects. This membrane can be described by 4 parameters: h, the distance between to two surfaces, r_0, the radii of the neck, r_c, the contact radii and φ_c, the contact angle. The shape and the membrane is also tied to the pressure difference between outside and the inside of the membrane, denoted as P_1 and P_2. Comparing figure C.5 with figures C.1 though C.4, shows that this is a suitable model for describing the shape of the molten metal bridge. To add the pinching of the neck, a fitting rate of change for the neck radii has to be found.

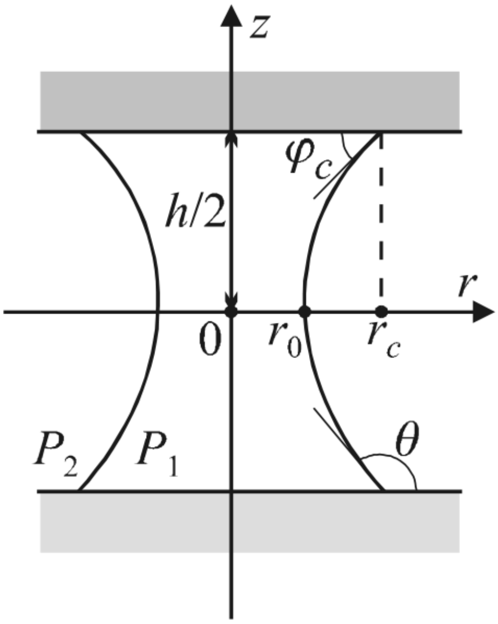


Figure C.5: Example of a capillary bridge between 2 flat surfaces[30].

Another approach is to model the bridge as a piece of wire with changing resistance, as this is electrically how it acts, when it is in short circuit mode. The resistance of a piece of wire is given by:

$$R = \frac{\rho L}{A} \tag{C.1}$$

Where:

ρ =	resistivity of the wire material	$[Wm]$
L =	lengths of the wire piece	$[m]$
A =	cross-sectional area of the wire	$[m^2]$

By multiplying the equation with $I(t)$, it can replace U_a in the electrical model of the welding system, when the system is short circuited, as the bridge is "replacing" the arc. From looking at the high speed footage, it is clear that the change in resistance in the molten metal bridge, manly comes from the change in the cross-sectional area of the bridge. By incorporating this into the model, it should be possible to model the bridge from an electrical standpoint. The lengths of the wire is the distance form the electrode to the workpiece. As the available data for the welding system is electrical, it makes the most sense to use this model to describe the molten metal bridge.

Appendix D

Migatronic Data Sets

This is attached as a folder containing all data received by Migatronics A/S for this thesis. The contents of the folder is:

- Multiple data sets, for 3 welding programs and various current and voltage settings, each containing current and voltage measurements for roughly 7 sec of welding
- The welding table, also seen in Appendix B
- The presentation given by Migatronic A/S, outlining and expanding upon the project proposal seen in appendix A

Appendix E

Classification Comparison Confusion Matrices

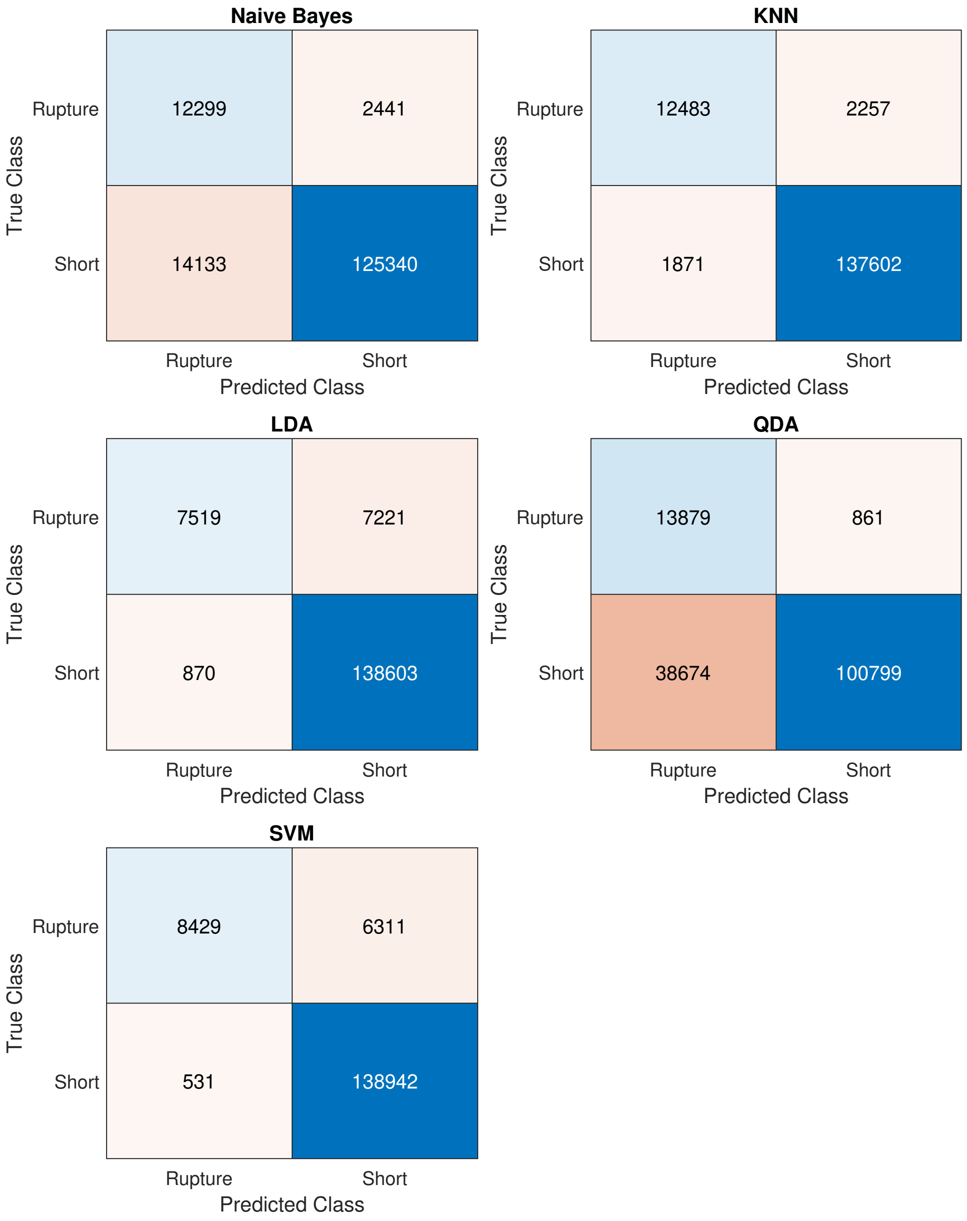


Figure E.1: Confusion matrices for classification comparison using feature set F

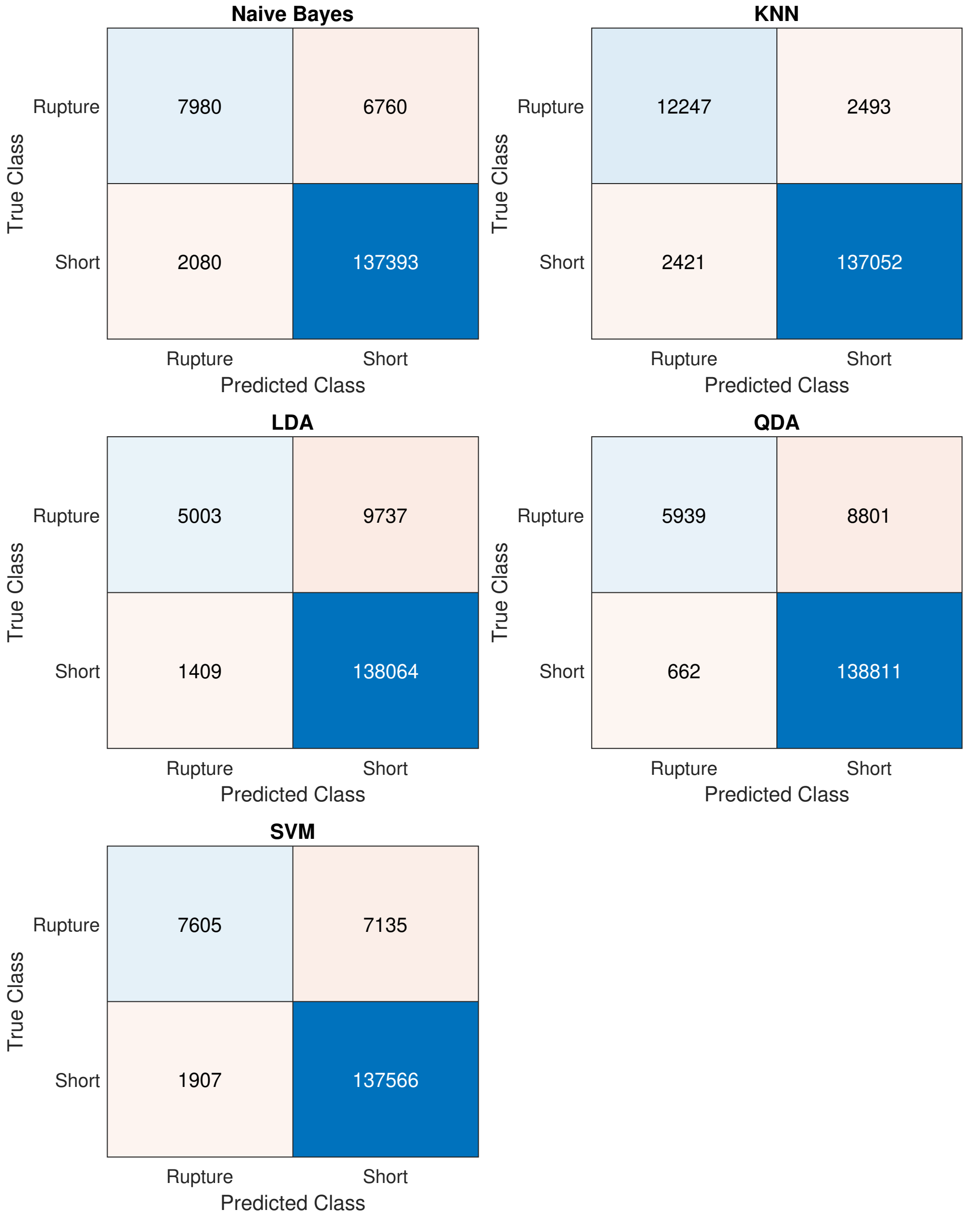


Figure E.2: Confusion matrices for classification comparison using feature set F_1

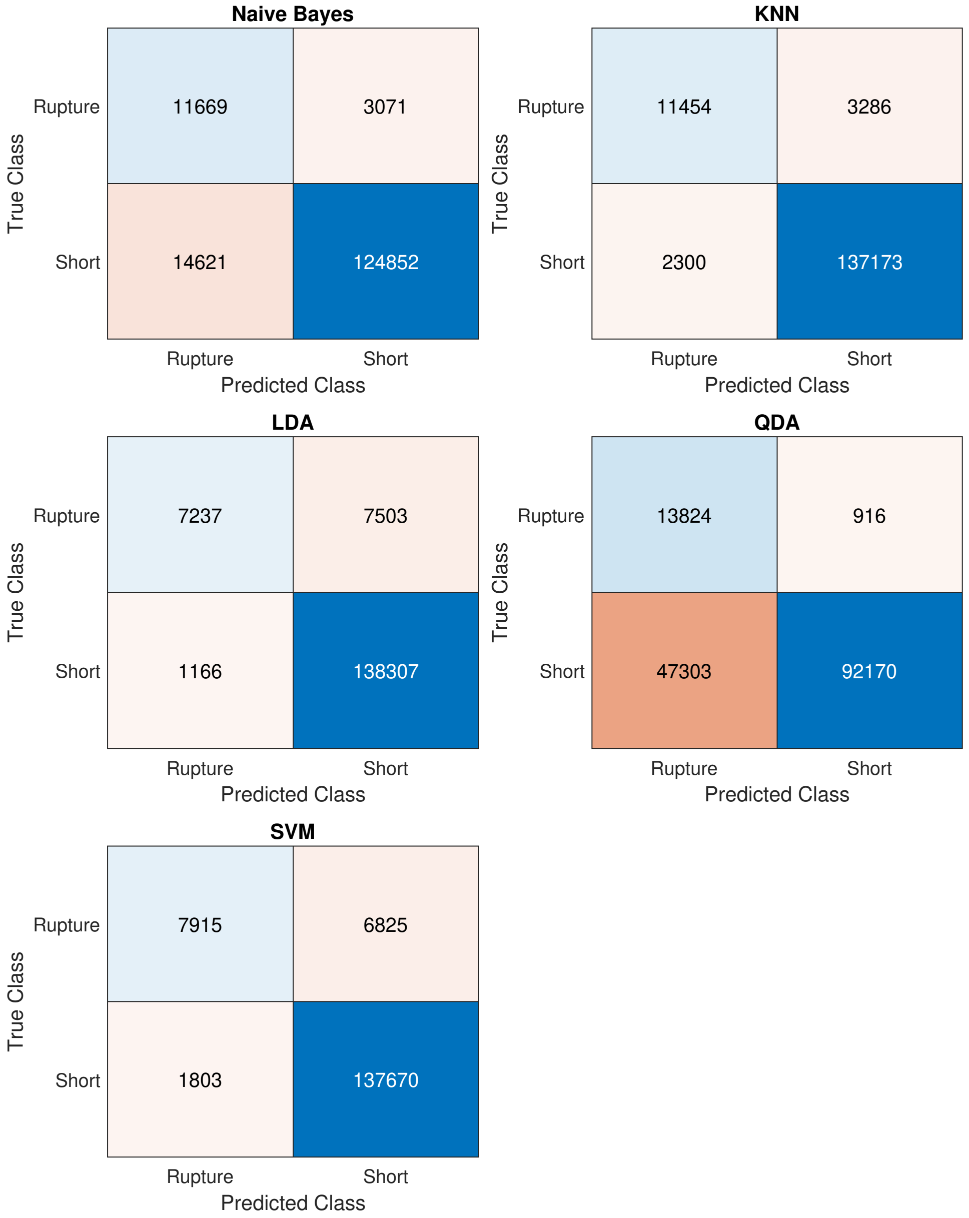


Figure E.3: Confusion matrices for classification comparison using feature set F_2

Appendix F

Thesis Summary - Dansk

Denne rapport undersøger hvorvidt der kan opstilles et system der er i stand til at forudse hvornår kortslutnings-fasen er ved at ophøre i en konventionel GMAW Short-Circuit svejseprocess, også kendt som et free burn eller rupture. Ved at forudse hvornår hvornår denne fase ophører, kan svejsestrømmen reguleres således at voldsomme brud på svejseelektroden undgås når kortslutningen ender, hvilket betyder at fjernelse af spatter fra svejseprocessen er mindre nødvendigt.
Problemet bliver opstillet som et binært klassifikationsproblem, hvor i ønskes at klassificere målingsdata fra svejseprocesser i kortslutnings-fasen som enten tilhørende "Rupture" klassen, hvis målingen er blandt de sidste målinger i kortslutnings-fasen, eller som tilhørende "Short" klassen, hvis målingen er blandt de resterende forkomne målinger i kortslutnings-fasen.
Strømstyrke- og spændingsmålinger bliver brugt til klassifikationsproblemet, og der ud over bliver der brugt en række sample-by-sample og statistiske features, som primært bliver ekstraheret fra spændingsmålinger fra svejseprocessen. Dertil bliver der opstilt en model simpel for tværsnitsarealet af svejseelektroden gennem kortslutnings-fasen, som det bliver forsøgt at estimere ved brug at en extended kalman-filter. Estimatet viser sig dog ikke at være realistisk. I stedet bliver residualet fra spændingsmålinger og kalman-filterets spændingsestimat tilføjet som en feature til klassifikationsproblemet.
For at finde den bedste klassifikationsmodel til problemet, bliver der lavet en sammenligning af forskellige modeller, baseret på en række præstationsmål udvundet fra forvirringsmatricer for de forskellige modeller i kryds-validering på træningsdata. Her bliver det vurderet at K-Nearest Neighbor modellen (KNN) har den bedste præstation, hvorefter optimale hyperparameter bliver fundet til at være Mahalanobis-afstand med 1 nabo.
I test af KNN-modellen bliver det vurderet at præstationen er acceptabel på testdata af samme afart som modellen træningsdata, hvorimod præstationen på testdata fra svejseprocesser med andre programmer og indstillinger sammenlignet med trænings-

dataet ikke var acceptabel. Ved at træne modellen på træningsdata fra en række svejseprocesser med forskellige programmer og indstillinger, og derefter lave en test, bliver det vurderet, at for at opnå en acceptabel præstation på et testdatasæt fra en svejseproces af en givet afart, skal klassifikationsmodellen være trænet med træningsdata fra en svejseproces af samme afart.