Graffiti detection using a variable temperature MOx sensor and Machine Learning

Christian Clemedtson,
Mikael Kristensson

Master's thesis
2021:E14

Lund Institute of Technology
Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Centrum Scientiarum Mathematicarum

Abstract

MOx sensors are widely used for sensing gases. In this thesis a variable temperature MOx sensor was used to detect graffiti spray paint and differentiate it from other substances using the sensors sensitivity to different gases at different temperatures using machine learning. The machine learning methods Support Vector Machines, k-Nearest Neighbour and neural networks were evaluated and compared. Different choices of features, including temperature sensitivity and derivatives, as well as different sensors and operating temperatures were tested. Data preprocessing, in particular standardization, was performed and its results evaluated. Data collection was done by spraying graffiti spray paint and applying other substances to the sensor outdoors.

This resulted in a graffiti spray paint detector that worked very well in differentiating graffiti spray paint from other substances. All three used machine learning methods performed well on the data, reaching a very good performance when there was a significant amount of gas reacting with the sensor. The inclusion of derivatives as a feature resulted in higher performance but a larger amount of false alarms due to the added noise.

Keywords: MOx sensor, variable temperature, machine learning, graffiti, spray paint, Support Vector Machine, k-Nearest Neighbour, neural network

Acknowledgements

This thesis was written during the fall and winter of 2020 and 2021 at the Centre of Mathematical Sciences at Lunds Tekniska Högskola.

We would like to express our gratitude towards our supervisor Maria Sandsten.

Mikael Kristensson and Christian Clemedtson

Contents

1 Introduction **5**

1.1 Background 5

1.2 Goals 5

1.3 Related work 6

2 Theory **7**

2.1 Gas sensors and gases 7

2.1.1 MOx sensors 7

2.1.2 Aerosol 8

2.1.3 Resistance curves 9

2.1.4 Features 10

2.1.5 Standardization 11

2.2 Machine Learning 12

2.2.1 Support Vector Machine (SVM) 13

2.2.2 k-Nearest Neighbour (kNN) 14

2.2.3 Neural Networks 15

2.2.4 Validation 18

2.2.5 Performance metrics 19

3 Preliminary experiments **21**

3.1 Provided data on propellant and solvent 21

3.2 More experiments on propellant and solvent 25

3.3 Different substances on wall indoors 26

3.4 Realistic measurements outdoors 28

4 Methodology **30**

4.1 Experimental setup 30

4.2 Data collection 31

4.3 Feature selection 32

4.3.1 Temperature sensitivity 32

4.3.2 Derivatives 33

4.4 Data preprocessing 34

4.4.1 Standardization . . . 34
4.4.2 Thresholding . . . 35
4.4.3 Dataset . . . 36
4.5 Model selection . . . 37
4.6 Real time classification . . . 38

5 Results **40**
5.1 Data visualization . . . 40
5.2 Effects of standardization . . . 43
5.3 Classification . . . 45
5.3.1 SVM . . . 45
5.3.2 kNN . . . 46
5.3.3 Neural Network . . . 46
5.4 Classification with derivatives . . . 47
5.5 Impact of different substances and inclusion of acetone . . . 49
5.6 Real time classification . . . 51

6 Discussion **53**
6.1 Data visualization and standardization . . . 53
6.2 Classification methods . . . 53
6.3 Effects of derivatives . . . 54
6.4 Impact of different substances and inclusion of acetone . . . 55
6.5 Additional remarks . . . 55

7 Conclusions and further research **56**
7.1 Summary . . . 56
7.2 Improvements and further research . . . 56

A Appendix - Resistance curves **57**

1 Introduction

1.1 Background

Gas sensors, or gas detectors, have long been used for the purposes of detecting combustible, toxic, flammable, environmentally damaging or otherwise undesirable gases. Cheap, reliable, small, low power-consuming and long-lasting gas sensors are desirable and allow for a wide range of applications [6]. Recently nanostructured Metal Oxide (MOx) gas sensors have been seeing more use because of their versatility and low production costs. These work by measuring the resistance of a semiconducting metal oxide, where changes to this resistance is indicative of the presence of gas. The semiconducting material is selected for having a large amount of electrons in its conduction band as well as oxygen vacancies leading to high reactivity between the materials surface and present gas [12].

According to Stockholm's city council the cost incurred by vandalism and defacement on the city's public transport in 2018 adds up to around 300 million SEK over the year [15]. Undesired graffiti has a risk of incurring damage to architectural heritage materials and its presence leads to losses in business and decreasing property values from the negative image it generates. Removing graffiti is a costly and time consuming process which is often environmentally damaging and gets more difficult the longer the paint stays on. Spray paint is generally removed using chemical solvents and paint strippers which may cause health risks, damage to the environment and permanent discoloration of underlying surfaces. Early detection of graffiti is desirable for several reasons. As mentioned it is easier to remove if done quickly and is not allowed to penetrate the underlying material but also sensors could be used to sound alarms and possibly stop the act early. The presence of graffiti also tends to attract more, worsening the issue if it is not removed promptly [21].

Using gas sensors to detect graffiti spray paint is particularly appealing as they make camera monitoring, which is expensive and privacy invading, redundant. Variable temperature MOx gas sensors in particular are cheap, small and require very little power. These do however have their own issues with humidity and temperature as well as a limited range and some out-of-the-box variance between sensors of the same model. The sensing material would need to be selected well for the sensor to have low to no cross-sensitivity between the specific gas it is meant to detect and any other to reduce the chance for false positives.

1.2 Goals

The goal of this thesis is to investigate the viability of using variable temperature Metal Oxide (MOx) gas sensors for the purpose of identifying graffiti spray paint from other gases and background. Differentiation is done using machine learning. Different machine learning methods are tested, analyzed and discussed.

Some challenges that have needed to be overcome to achieve this goal includes:

- Experimentation and analysis of the possibility to use one variable temperature MOx gas sensor to separate graffiti spray paint from other gas compounds.
- Finding a suitable representation of data to be put into machine learning algorithms.
- Finding suitable machine learning methods to apply to the data.
- Analyzing the performance of the machine learning methods Support Vector Machines, k-Nearest Neighbour and neural networks.
- Data collection with cans of graffiti spray paint and other substances.

- Construction of a real time prototype that can detect spray paint of unknown concentrations in real time in realistic conditions with as few false alarms as possible.

We have limited ourselves to detect graffiti spray paint because of the ease of gathering data and its practical area of use.

1.3 Related work

MOx gas sensor have been used in a wide range of applications. When they are used for the purpose of differentiating different gases the general approach is using machine learning methods. Most of the scientific experiments on the subject is done in laboratory conditions where the sensors are placed in enclosed environments and can react freely with gases for large amounts of time under known environmental conditions. A large part of experiments are performed using an array of multiple MOx sensors.

In a paper from 2015 [16] a comparison was made between different machine learning methods for odor classification. The evaluated algorithms were principal component analysis, linear discriminant analysis, k-Nearest Neighbour (kNN), Support Vector Machine (SVM) and three types of neural networks. It was concluded that kNN, SVM, and two variations of the neural network had the best performance with a flawless test error.

Neural networks were used in another experiment in 2013 for odor classification [18] where 14 sensors were used in an array. The maximum value of the resistance curve and simple approximations of derivatives were used as input to a backpropagation neural network. A similar paper was published in 2018 [3] where gas sensors with varying temperatures was used. Maximum value and an approximation of the derivative to this value were fed into a neural network which performed the classification. A comparison between different types of neural networks for the purpose for gas classification has also been made [26] where the backpropagation neural network performed better than other variants.

In a paper from 2019 [19] a combination Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) was proposed. The CNN extracts features from the input data, which are fed into a Long Short-Term Memory (LSTM) based recurrent layer which extracts the sequential information of the resistance curve. This implementation is capable of recognizing a gas species with an accuracy of 84.06% with a 0.5 second response time or an accuracy of 98.28% with a 4 second response time.

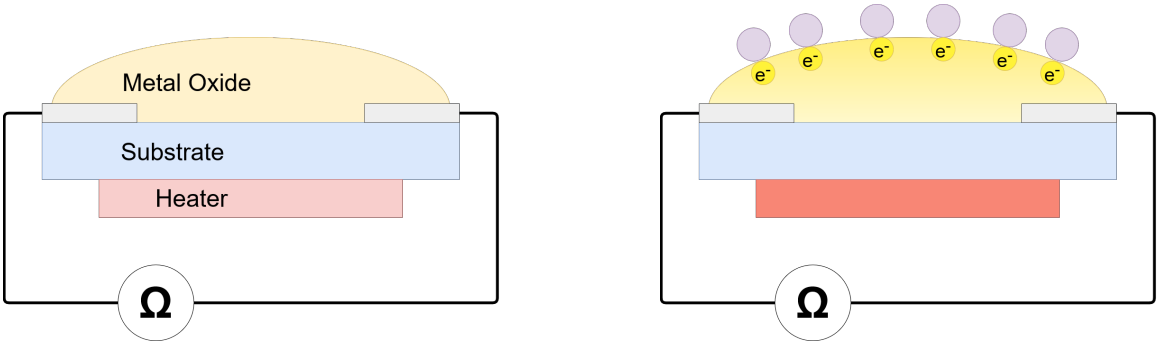
Another study of interest is a paper from 2017 [14] where no feature extraction was performed but instead only the first 60 points from when the gas started reacting with the sensor were classified. The points were measured with a MOx sensor array with 8 sensors and subsequently the points were directly fed into machine learning classifiers. It was concluded that classification could be done with 96.7% accuracy within six seconds, 12 times faster than if general features such as steady state value and derivatives would have been used.

2 Theory

This section will go over the basic operation and structure of variable temperature Metal Oxide (MOx) gas sensors as well as how to interpret and use their output. Then we go over feature extraction and standardization preprocessing and finally explain the theory of the machine learning methods used for this classification problem.

2.1 Gas sensors and gases

2.1.1 MOx sensors



(a) *(b)*

Figure 1: (a) Basic structure of a MOx gas sensor. The metal oxide acts as a resistor, whose resistance is measured. (b) At high temperatures in clean air, oxygen is adsorbed to the surface of the metal oxide, trapping electrons and increasing resistance.

A MOx gas sensor is made up of a heater element and a metal oxide chemiresistor on a silicon-based substrate, as shown in Figure 1a. At high temperatures, above around 100°C, oxygen adsorbs to the surface of the metal oxide trapping electrons as illustrated in Figure 1b and increases the measured resistance [5]. Adsorbtion is the adhesion of atoms, ions or molecules to a surface. The presence of combustive, reducing or oxidizing gases disturb this adsorbtion in a multitude of different ways depending on gas and metal oxide all of which result in resistance changes on the sensor [8]. The resistance measured is thus linked to the occurrence and concentration of these gases around the sensor. For the purpose of identifying specific gases it becomes apparent that resistance measurements of one sensor at one temperature is insufficient unless concentration is known as there are at least two unknown variables (type of gas and concentration) to one measurement (resistance).

In order to ascertain which substance the sensor is detecting one would need to add dimensions to the data. A common way to accomplish this is by using several sensors with different metal oxides together [18] [19] [16] as each will react slightly differently depending on the occurring gas and thus their readings together show unique fingerprints for any measured gas assuming these metal oxides are chosen well.

Another way to add dimensions to the data is through quickly varying the temperature of one sensor as there is a correlation between the operating temperature of MOx gas sensors and their sensitivity to different gases. A variable temperature MOx sensor, such as the one used in this thesis, has a heating element which will

quickly and precisely change temperatures, allowing for temperature sweeps where resistance measurements are taken at many specific temperature levels in very quick succession. A simple example to explain how this is useful follows: lets say a variable temperature MOx sensor is exposed to two gases, gas A and gas B, at two different times under similar conditions. The sensor sweeps over the temperatures 100 °C and 200 °C. At 100 °C we measure a data point with a resistance of 10Ω for each gas. During the same sweep the measurement at 200 °C for gas A is 15Ω but is 14Ω for gas B. Explaining the underlying chemical and physical causes for these differences are beyond the scope of this thesis, but what is important to note is this: a variable temperature MOx sensor will give different resistance values when exposed to different gases due to the properties of said gases. Because of this, it is possible to distinguish different gases using only one sensor despite not knowing gas concentrations.

MOx sensors are susceptible to environmental effects. One factor which greatly impacts the sensitivity of MOx sensors is humidity. Water molecules will bind to the sensing area on the surface of the metal oxide, resulting in a decreased surface area where the target gas can react with the metal oxide, affecting the sensitivity of the sensor and the measured resistance [2]. Temperature is another environmental factor that may affect the sensors. Another issue is that the innate resistance of any MOx sensor of the same model will vary slightly and may change over time.

2.1.2 Aerosol

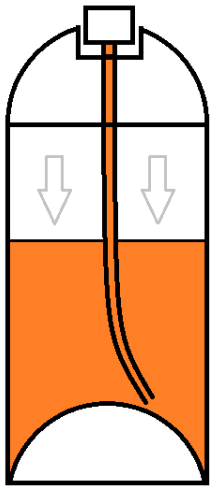


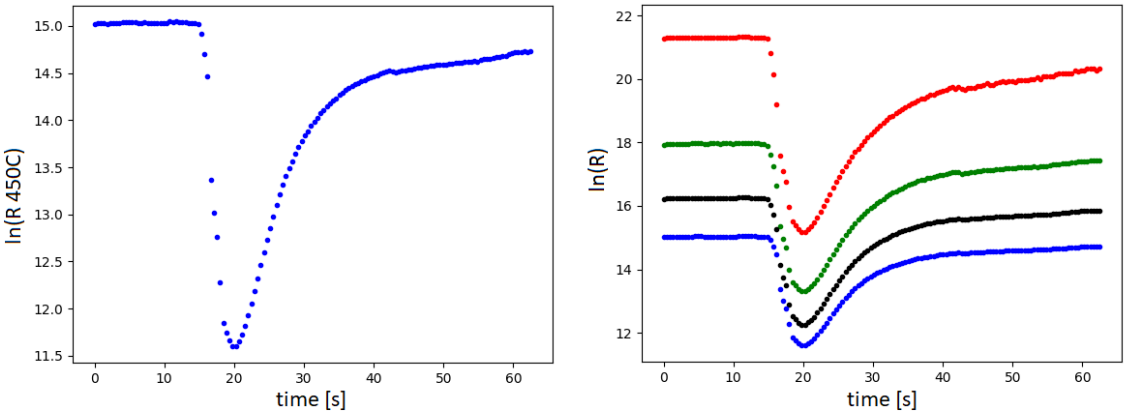
Figure 2: A general spray can. Propellant pushes the mixture down, forcing it into the dip tube. The dip tube leads to a push valve which opens when depressed.

Aerosol spray paint cans, shown in Figure 2, contain a mixture of propellant and solvent. The purpose of the propellant is to get pressure in the can to make it suitable for spraying and solvents are often used in paints to mix the paint components into a homogeneous solution. Inside the can, the mixture of propellant and solvent is being pushed down by propellant toward a dip tube. This tube leads to the top of the can where a valve button is located. When the button is depressed the valve opens, allowing solvent and propellant mixture to be released in an aerosol spray.

An aerosol is solid or liquid particles suspended in a gas. When spray paint is applied to a surface the

suspending gas will quickly spread and disperse while the suspended particles adhere to whichever surface was sprayed on. Some of these suspended substances, such as acetone and xylene will evaporate at and below room temperature, releasing fumes which may be detected using gas sensors for a long time after spraying [24] [23].

2.1.3 Resistance curves



(a) *(b)*

Figure 3: Log-resistance as a function of time when the sensor is exposed to gas. (a) shows the resistance curve at 450°C. (b) shows several curves from different temperatures. Red: log-resistance for 150°C. Green: log-resistance for 250°C. Black: log-resistance for 350°C. Blue: log-resistance for 450°C

The sensor response of MOx gas sensors are in the form of resistance values. Typically when a sensor is exposed to gas its resistance start to decrease. Figure 3a shows a typical resistance curve from when a sensor being exposed to some gas, in this case spray paint. Prior to 14 seconds the resistance values are fairly constant. This baseline resistance reflects the composition of the surrounding air when the sensor is not directly exposed to any substance. At approximately 14 seconds in the gas starts to react with the sensor and resistance quickly decreases until it reaches a minimum at 20 seconds, where the reacting gas concentration around it has reached a maximum. Afterwards the gas disperses and resistance values increase until the baseline is reached once more.

Figure 3a shows the measured resistances for only 1 temperature. The sensor that we have used has a heating element and does a temperature sweep. For every sample point the sensor cycles through a set of temperatures and registers a measurement at each temperature level. Figure 3b shows the same event as Figure 3a but with resistance values from every temperature in the sweep. This temperature sweep goes through 4 temperatures: 150°C, 250°C, 350°C and 450°C and registers a measured resistance point for each temperature.

The notation of a log-resistance point at index *i* in a resistance curve is defined as $x_n(i)$. The variable *n* is the index of the temperature, $n = 1, 2, 3$ or 4 depending on if it is measured in 150°C, 250°C, 350°C or 450°C.

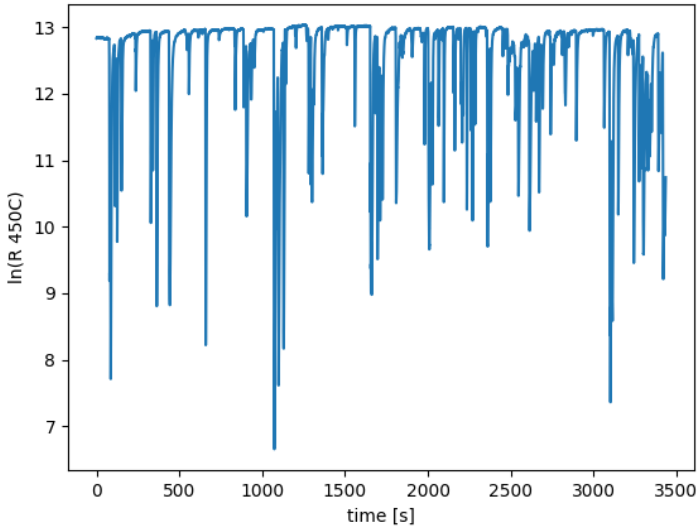


Figure 4: Log-resistance as a function of time when the sensor is exposed to multiple sprays of spray paint over almost an hour.

As previously mentioned, when a MOx sensor is exposed to a gas which it is sensitive to its internal resistance dips. Figure 4 shows a typical session where spray paint was repeatedly sprayed near a MOx sensor for almost an hour resulting in many such minima. The notation ln(R 450C) means that the log-resistance is measured at 450°C. Note that Figure 4 shows only the resistance curve for one temperature and as all measurements were taken with a variable temperature MOx sensor there would be several such curves, one for each temperature measured at.

There is a highly non linear relationship between the gas concentration and the measured resistance values from the sensor. Resistance values are logarithmized as a way to linearize the highly non linear relationship between gas concentration and the resistance. This may help if the variation of the concentration of the target gas is large [13]. The classification of different gases should be possible despite not knowing gas concentration.

2.1.4 Features

In the context of machine learning, a feature is a measurable property of observed data. For classification purposes it is desired that a combination of features will produce a pattern that can be used in separating different classes. The selection of features is an important part in any machine learning classification task. There are many types of features that can be extracted from MOx sensors. Some features are direct piecewise properties of a resistance curve, for example the minimum value, derivatives, area, rise time and fall time of the resistances curves [13].

When the sensor has been exposed to a significant amount of reacting gas the sensor can reach a steady state where the resistance stabilizes on a constant value. This steady state value is highly dependent on the concentration of the target gas and is determined by the minimum of the resistance curve. The negative

derivative to this point and the positive derivative after this point are also relevant features, representing the concentration change rate [25] and rate of reaction [13]. These features are shown in Figure 5.

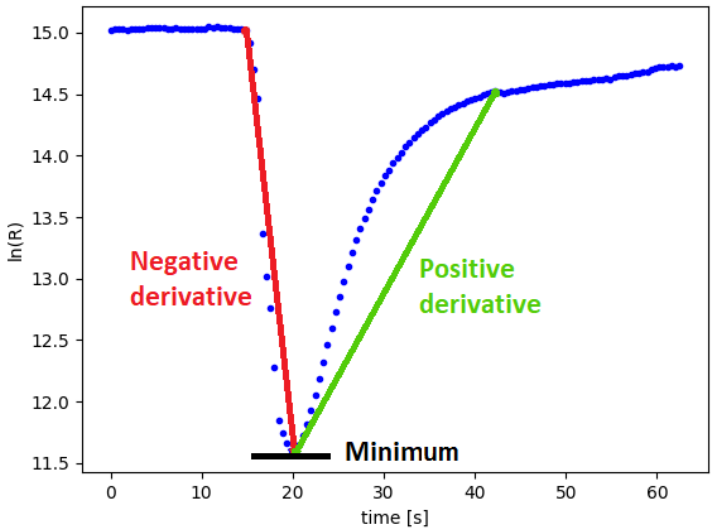


Figure 5: Log-resistance as a function of time when the sensor is exposed to some gas. Using the resistance minimum along with positive and negative derivative as features can be useful in controlled environments.

The steady state response and derivatives have been used multiple times in practice to succesfully differentiate gases [3] [18]. The minimal and maximal derivative between two points have also been used [13]. However, these applications are often performed in laboratory conditions where gas is pumped into an enclosed space with the sensor. The sensor can then continuously react with the target gas for multiple minutes and reach a stable value. These conditions will not be met if the sensor is placed in less controlled environments such as outside.

2.1.5 Standardization

External factors can have a large impact on measured resistances. Operating a MOx sensor outside makes it especially susceptible to environmental effects. The two major environmental factors which affect the sensors operation are humidity and ambient temperature, as described in Section 2.1.1. The sensor might also work differently depending on the model, if it's inside or outside or permanently change from long-term use or excessively high temperatures. These kinds of effects are also known as sensor drift. It is necessary to counteract these effects if classification with a MOx sensor is to work in realistic conditions. Standardization is a process of putting data in a standard form and can be used to counteract external effects on the sensor data. There are many standardization methods to standardize new data to the form of older data. This can also be viewed as a calibration of new data.

There are many methods that can be used for standardization. Different methods are suitable for different kinds of sensor drift. Some require manual calibration by exposure of a reference gas. Some take advantage of multiple sensors in a sensor array. Some simple common methods of standardization are [7]:

- Differential: $y_n(i) = x_n(i) - x_n(0)$
- Relative: $y_n(i) = \frac{x_n(t)}{x_n(0)}$
- Fractional: $y_n(i) = \frac{x_n(t) - x_n(0)}{x_n(0)}$

where $x_n(i)$ is a log-resistance point at index i in a resistance curve and *n* is the index of the temperature. $n = 1, 2, 3$ or 4 depending on if it is measured in 150°C, 250°C, 350°C or 450°C. $y_n(i)$ is the corresponding standardized point and $x_n(0)$ is the measured baseline. These are all relating the measured data points to the baseline, which often vary due to sensor drift. The differential method counteracts additive sensor drift and the relative method counteracts multiplicative sensor drift [7].

Another method of standardization is to use a mathematical function *f* to transform measured values from not standardized data points to some reference points. The objective is to find the function *f* to transfer the data points $x_n(i)$ from one sensor array to predict the data points $y_n(i)$ in the other sensor array for all values of i such as [4]:

$$y_n(i) = f(x_n(i))$$

A proposed functions was

$$y_n(i) = a_n x_n(i)$$

for a constant a_n. Another proposed function with an added intercept constant c_n was

$$y_n(i) = a_n x_n(i) + c_n$$

The constants a_n and c_n could for example be found by minimizing the sum of squared error between $y_n(i)$ and $f(x_n(i))$ for all i. It is a way of linearly transforming one set of data points to a form that resembles another set of data points. Another variant of this equation can be expressed by omitting a_n, resulting in

$$y_n(i) = x_n(i) + c_n \tag{1}$$

Another method of performing similar standardization is by using a transformation matrix F to map points from an uncalibrated system to a reference space [9], such as:

$$S_{reference} = S_{uncalibrated} F \tag{2}$$

where $S_{reference}$ and $S_{uncalibrated}$ are response matrices from a measurement which are used to calibrate an uncalibrated system to a reference system. A response matrix is a measured matrix where rows are data points used for the transformation and columns are features for each point [9]. The transformation matrix F could for example be computed with the pseudoinverse.

2.2 Machine Learning

Machine learning is a subset of artificial intelligence. It encompasses all algorithms which improve from experience. Training data is learned from to construct a model which is then used to classify or predict new data. Support Vector Machines, k-Nearest Neighbour and neural networks are all considered machine learning methods. The machine learning methods in this section will be used to classify individual data points **x** which corresponds to a measured 4-dimensional log-resistance point.

2.2.1 Support Vector Machine (SVM)

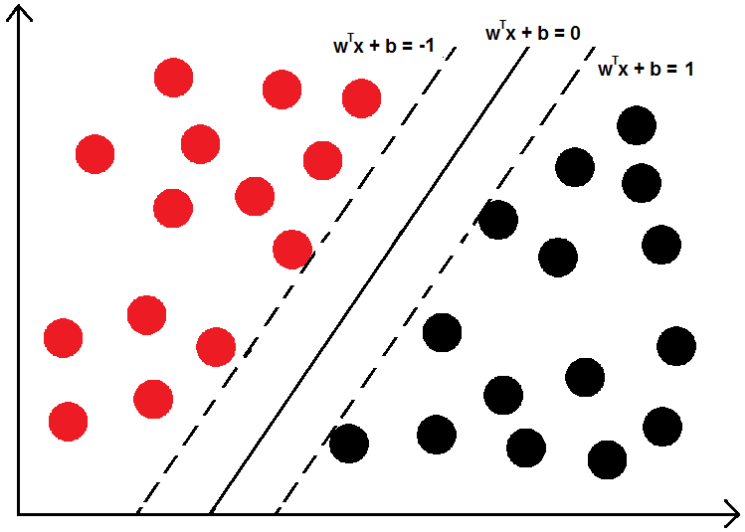


Figure 6: The principle of classification with an SVM. The line in the middle is a linear decision boundary $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ *which separates the red points from the black points.*

The Support Vector Machine (SVM) is a linear classifier that is used for both linear and non linear classification. The main concept of the SVM is that it produces a linear decision boundary that separates two classes, as seen in Figure 6. The line in the middle is the decision boundary that separates the red points from the black points. For a space with dimension n this line would be a hyperplane of dimension $n - 1$. One way to determine the separating hyperplane is to choose it so that it is as far away from the closest training points as possible. This is called large margin classification [10]. The dashed lines are determined by the closest training samples and are called support vectors. The margin is the space between the support vectors. It is called hard margin classification if the hyperplane is chosen in a way that all data points of one class is on one side of the hyperplane and all data points of the other class is on the other side. This means that there would be no margin violations, no points would lie within the margin or on the wrong side of the hyperplane.

Most of the time hard margin classification is not good enough, for example if the data is not linearly separable. An n-dimensional dataset is not linearly separable if it can not be perfectly separated by a hyperplane of $n - 1$ dimensions. For example data in two dimensions as in Figure 6 is linearly separate because it can be perfectly separated by a one-dimensional line. Soft Margin Classification [10] is a more flexible model. It allows for some training points to be classified on the wrong side of the separating hyperplane. The tradeoff between the width of the margin and the margin violations can be controlled with a hyperparameter called C. A small C leads to a wider margin with more margin violations. A large C leads to a more narrow margin with less margin violations. Generally a smaller C leads to a more generalizable model [10].

The classification is done by computing the decision function $\mathbf{w}^T\mathbf{x} + b$ where $\mathbf{x}$ is the point to be predicted, $\mathbf{w}$ are weights and b is a bias. Both $\mathbf{w}$ and b are parameters of the hyperplane and are determined during training. The decision function outputs either 0 or 1 depending on which class is predicted, such as

$$y = \begin{cases} 0 \text{ if } \mathbf{w}^T\mathbf{x} + b < 0 \\ 1 \text{ if } \mathbf{w}^T\mathbf{x} + b \geq 0 \end{cases}$$

where y is the predicted value [10]. The prediction is based on which side of the hyperplane **x** lies. In Figure 6 the two dashed lines, called support vectors, are given by $\mathbf{w}^T\mathbf{x} + b = -1$ and $\mathbf{w}^T\mathbf{x} + b = 1$. The line in the middle is the hyperplane $\mathbf{w}^T\mathbf{x} + b = 0$.

Determining the weights of the hyperplane is done by solving an optimization problem. For hard margin classification the following problem is solved [10]:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} \\ \text{subject to} \quad & t^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 \text{ for } i = 1, 2, \ldots, m \end{aligned}$$

where $t(i) = 1$ for points of the first class and $t(i) = -1$ for points of the other class. m is the total number of data points. The idea is to minimize $||\mathbf{w}||$ to get a large margin while not allowing any margin violations. $\frac{1}{2}\mathbf{w}^T\mathbf{w}$ has the same minima as $||\mathbf{w}||$ and is easier to minimze because of its differentiability. For soft margin classification a slack variable $\zeta^{(i)}$ is introduced. It determines how much data point i is allowed to violate the margin. The following optimization problem needs to be solved [10]:

$$\begin{aligned} \min_{\mathbf{w},b,\zeta} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{m}\zeta^{(i)} \\ \text{subject to} \quad & t^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \zeta^{(i)} \quad \text{and } \zeta^{(i)} \geq 0 \quad \text{for } i = 1, 2, \ldots, m \end{aligned}$$

The hyperplane can not straightforwardly separate non linear data. In the non linear case additional features are added to the data, so called similarity features. They can be computed using a similarity function of the original data points. This means that original data points are mapped to a much higher dimensional space where there exists a hyperplane that can satisfyingly separate the two classes. One example of a similarity function is the Gaussian radial basis function (RBF):

$$\phi_\gamma(\mathbf{x}, \ell) = \exp(-\gamma||\mathbf{x} - \ell||^2)$$

where γ is a hyperparameter. A small γ leads to a smoother decision boundary and a large γ leads to a more irregular decision boundary, more affected by each individual data point. In practice the process of adding similarity features is done using kernels making it much more computationally efficient [10]. A kernel function ϕ quantifies the similarities between two data points **a** and **b** by computing the dot product $\phi(\mathbf{a})^T\phi(\mathbf{b})$. The Gaussian RBF kernel is defined as

$$K(\mathbf{a}, \mathbf{b}) = \exp(-\gamma||\mathbf{a} - \mathbf{b}||^2)$$

Some other popular kernels are the linear kernel, polynomial kernel and the sigmoid kernel.

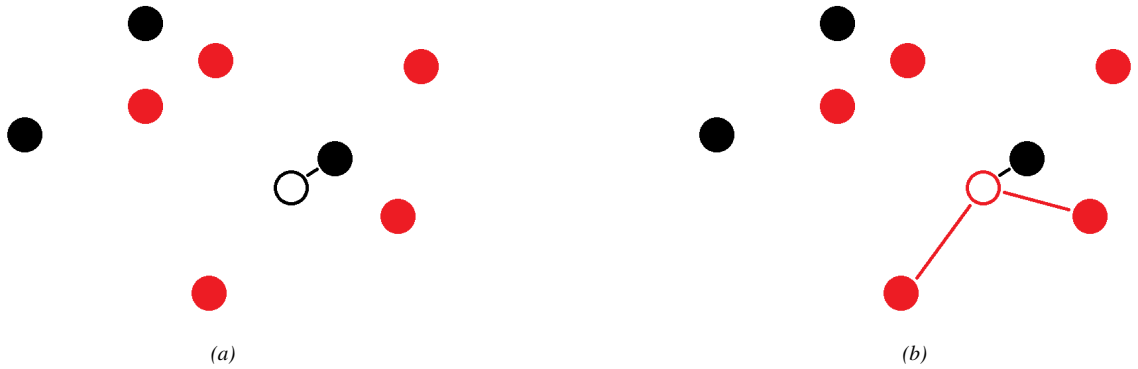
2.2.2 k-Nearest Neighbour (kNN)

The k-Nearest Neighbour (kNN) model is a fairly simple classification model that works well when there is plenty of data with few dimensions. Training data is labeled, meaning it is stored with information on which

class it belongs to. In the case of our data there are two labels, 'spray paint' and 'not spray paint'. To do classification, new data is designated the same label as the majority of the k nearest data points from the training data set. "Nearest" in this case is calculated using a *Minkowski distance*, or L^p norm, defined as

$$L^p(\mathbf{a}, \mathbf{b}) = (\sum_{i}^{n} |a_i - b_i|^p)^{\frac{1}{p}}$$

where **a** and **b** are two points defined in n dimensions [22]. For example, the euclidean norm and L^2 norm are the same.



(a) *(b)*

Figure 7: Representations of k-Nearest Neighbour classification. The hollow circle represents a new data point to be classified. Red and black dots represent two different classes in a binary classifier. left has $k = 1$ and right has $k = 3$

Figure 7 shows the *k*-nearest neighbour model working as a binary classifier. Different values of *k* will occasionally give different results, as shown in these images where in Figure 7a the circle is classified as black but when more neighbouring data points are taken into account, as shown in Figure 7b, it gets classified as red. To avoid ties, which is generally to be desired, *k* should be designated as an odd number.

2.2.3 Neural Networks

Multilayer perceptrons, also called feedforward neural networks or deep neural networks are the quintessential deep learning models. When used as a classifier, a feedforward network works by defining a mapping $y = f(\mathbf{x}; \theta)$ and learning the value of parameters θ which result in the best function approximation [11].

The most simple form of a neural network is the perceptron which made up of a number of input nodes leading to a single output node. Data from the input nodes are multiplied by *weights*, summarized along with a *bias* and parsed through an *activation function*. The activation function is predefined and remains unchanged but weights and bias are optimized during training in order to minimize the networks *loss function*. The structure of a perceptron is that of a neural network with only one node.

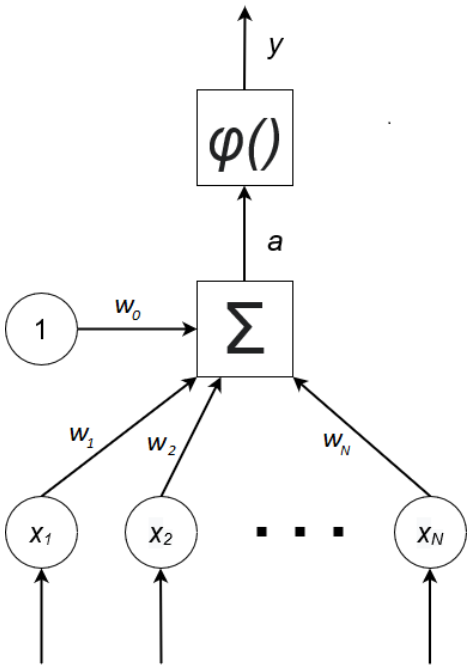


Figure 8: Visual representation of a perceptron. x are input values, w are weights, w_0 is bias, y is output.

Figure 8 is a basic one-node perceptron drawn up as a flowchart. The output y is calculated according to the following equation:

$$\begin{aligned} y &= \varphi(a) \\ a &= \sum_{n=1}^{N} w_n x_n + w_0 \end{aligned} \tag{3}$$

where φ is an activation function and a is the argument to the function, the sum of values in previous nodes multiplied by their weights. As mentioned before, the goal is to optimize the weights in order to minimize a loss function. A loss function is chosen to fit the problem the network is meant to solve. The loss function should always be differentiable and its value should decrease as the network gets better at whichever task it is meant to do. The value of a loss function should be higher the more incorrect a guess made by the network is. Examples of loss functions are:

$$\begin{aligned} Mean\ squared\ error \quad & E(\mathbf{w}) = \frac{1}{2N} \sum_{n}^{N} (y_n - d_n)^2 \\ Cross-entropy\ error \quad & E(\mathbf{w}) = -\frac{1}{N} \sum_{n}^{N} \big(y_n ln(d_n) + (1 - y_n) ln(1 - d_n) \big) \end{aligned} \tag{4}$$

where d_n is the predicted value and y_n is the actual value.

Mean squared error is used for regression tasks and minimizing this loss function is equivalent as solving a least squares regression problem. Cross-entropy is used in binary classification tasks where y_n is either 0 or 1 and d_n is bound in $[0, 1]$. It is apparent from Equation 4 that when every predicted value, d_n, is equal to the real solution, y_n, for every the cross-entropy loss is 0, but it will increase if this is not the case. A popular choice of optimizer to minimize the loss function is *stochastic gradient descent*. An important hyperparameter to adjust in the optimization is the *learning rate* which determines how much to change the weights in the case of errors.

The *multilayer perceptron* is constructed by adding one or more hidden layers to the perceptron. These hidden nodes serve as feature detectors, each being sensitive to different regions of the input space. Adding more layers helps the network to detect more complicated features and fit closer to more varied input data.

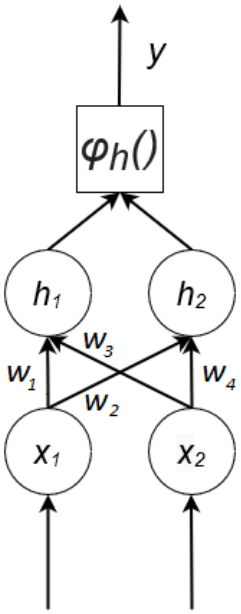


Figure 9: Visual representation of a multilayer perceptron with two input nodes and two hidden nodes, h_1 and h_2, in one hidden layer.

Figure 9 shows a visual representation of a multilayer perceptron with two input nodes and one hidden layer containing two hidden nodes. It is slightly more condensed compared to Figure 8. Each layer in a multilayer perceptron is fully connected, meaning each node in a layer connects to each node in the next. Note that there can be any number of nodes in a hidden layer, indiscriminate of the amount of input values. The equation for a general multilayer perceptron with one output would be as follows:

$$\begin{aligned} y &= \varphi_0\big(\sum_{i=1} \tilde{w}_i h_i + \tilde{w}_0\big) \\ h_i &= \varphi_h\big(\sum_{n=1} w_n x_n + w_0\big) \end{aligned} \tag{5}$$

Note that the value of a hidden node is calculated the same way as the output, y, is calculated for the percep-

tron in Equation 3. This remains true for any number of hidden layers, where the output from the previous layer is treated as the input to the new one. There is no limit to the amount of hidden layers that may be added but more nodes can add complications, such as increased training time and *overfitting*, which is explained in the next section.

The activation function φ is chosen based on the type of problem the network is meant to solve and different layers in the same network can have different activation functions. Examples of activation functions are:

$$\begin{aligned} Linear, &\quad \varphi(a) = a \\ Sigmoid, &\quad \varphi(a) = \frac{1}{1+e^{-a}} \\ ReLU, &\quad \varphi(a) = max(0, a) \end{aligned} \tag{6}$$

The principal utility of activation functions lies in adding non-linearity to a network, allowing deep network to learn high order polynomials beyond one degree. The linear activation function is essentially the absence of an activation function as it directly maps the summarized and weighted input to the output. This means the linear activation function is useless for hidden layers as any hidden layer with a linear activation function might as well just be removed. [17]

The sigmoid activation function, which can also be referred to as the logistic activation function, is a non-linear activation function most commonly used for binary classification tasks [17]. Its output is bound to the interval $[0, 1]$ meaning the sigmoid activation function can be used as input to cross entropy error.

Rectified Linear Unit, ReLU, function is the most widely used activation function [20]. It is quick to compute as it does not require the calculation of exponentials or divisions while still being non-linear.

The training of a neural network is generally done in several *epochs*. Training for one epoch means that all training data has been trained on once. Weights are saved between epochs meaning doing training over more epochs generally leads to the network fitting closer to training data.

2.2.4 Validation

For the purpose of determining the quality of any machine learning approach, in particular how it generalizes to new data, some form of validation is used. Typically the dataset is split into a training set and a test set. The training set contains the data that the machine learning methods are trained on and the test set contains the data that the machine learning methods are evaluated on to see how well they perform. The simplest approach of doing this is holdout cross-validation, meaning one randomly sets aside some portion of the training data as validation data. The network is then trained on the training data after which accuracy is calculated on the validation data. The limitation of this approach is that not all the data can be used in training. If too much validation data is set aside we might get a poor hypothesis as there is not enough training data and if too little is set aside we may not get a good accuracy estimate. However, it does not have to be a random split of training and test data. For example if the data is collected on different days it is good choice to put data from the same days in the training data and data from the other days in the test data.

In any of the methods used an issue called overfitting might arise. Overfitting is when a network is fitted too closely to its training data resulting in it being bad at generalizing to new data. A sign of overfitting is that there will be a significant difference in training accuracy and validation accuracy, meaning the network is much better at predicting data which is a part of its training set compared to data which is not.

In order to avoid overfitting in a neural network a method called *dropout* can be implemented. With this

method, nodes are randomly temporarily removed during training to prevent nodes collaborating too much and fit to irrelevant features of the training data [22].

2.2.5 Performance metrics

There are many metrics that can be used to determine the performance of a classifier. In the context of spray paint classification an observed point will be classified as either 'spray paint' or 'not spray paint'. In binary classification these classifications are typically called positive or negative. A point is classified as positive if it is classified as 'spray paint' and negative if it is classified as 'not spray paint'.

The *confusion matrix* is a matrix representation showing which observed points are being classified as what class. It is defined as a matrix C where element C_{ij} is how many points belonging to class i is being classified as class j. For binary classification a confusion matrix looks like:

$$\text{Confusion matrix: } \begin{pmatrix} \text{True positives} & \text{False negatives} \\ \text{False positives} & \text{True negatives} \end{pmatrix}$$

In the context of spray paint classification a true positive means that a 'spray paint' point is being classified as 'spray paint'. A true negative means that a 'not spray paint' point is being classified as 'not spray paint'. A false negative means that a 'spray paint' point is being classified as 'not spray paint'. A false positive means that a 'not spray paint' point is being classified as 'spray paint'. In the ideal case all points are either true positives or true negatives. This is when all points are correctly classified and the confusion matrix is a diagonal matrix.

The most straightforward performance metric is the *total classification accuracy*, defined as:

$$\text{Accuracy} = \frac{\text{True positives + True negatives}}{\text{True positives + True negatives + False positives + False negatives}}$$

This is a good evaluation of the overall classification performance. However if the dataset is imbalanced, where there are many more data points of one class compared to the other, total classification accuracy may not be a fair representation of the performance. Some metrics that can be evaluated on each class individually are precision, recall and F1-score [1], defined as:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives + False positives}}$$

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives + False negatives}}$$

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision + Recall}}$$

Precision quantifies what proportion of positive classifications are correct. *Recall* quantifies what proportion of positive data points are correctly classified. *F1-score* is defined as the harmonic mean of precision and recall. In the ideal case they are all equal to 1 and in the worst case they are equal to 0. The precision, recall and F1-score do not take true negatives into account, making these suitable metrics for an imbalanced dataset where true negatives are of less importance.

Another metric that is relevant when false positives are to be especially avoided is the *False positive rate*. It describes the proportion of false positives and all points that could result in a false positive. It is defined as:

$$\text{False positive rate} = \frac{\text{False positives}}{\text{False positives + True negatives}}$$

Ideally the false positive rate should be as small as possible. For example, if the sensor registers 100 points of 'not spray paint' and only 5 of them are being misclassified as 'spray paint', this will result in a false positive rate of 0.05.

3 Preliminary experiments

The following section has a slightly unusual structure. We have split the process of initial data collection and analysis in four parts, each with their own subsection. Each subsection contains background followed by results followed by conclusions. The reason for this is that a large part of the work done on this project was in experimenting with sensors and we find it important to explain our reasoning and provide context to how our later data collection and analysis was done.

Initially it was unknown if it was even possible to achieve any satisfying results. We could not find papers describing this kind of sensor being used in this way before. This section follows the performed experiments, starting from trying to differentiate propellant and solvent in controlled environments, to more realistic measurements both indoors and outdoors. To be clear, none of the data collected from experiments in this section is used after this section. Our data collection leading to results is detailed in Section 4.

3.1 Provided data on propellant and solvent

As mentioned in Section 2.1.2 an aerosol spray is composed of two components, a liquid or solid *solvent* which is suspended in a *propellant* gas. Experiments were carried out to see if it is possible to distinguish one from the other. The combination could then be recognized to identify spray paint specifically while avoiding false alarms from only either propellant or solvent reaching the sensor.

As a starting point data was provided to us from several experiments performed using two different spray cans. ”Can A” was used to act as propellant and contains 50-100% dimethylether, 10-25% acetone, 10-25% aliphatic polyisocyanate, 5-10% butyl acetate and 1-2,5% 2-methoxy-1-methyl acetate. While this mixture is mostly dimethylether which is a widely used propellant in aerosol sprays it also contains the ubiquitous solvent acetone.

The other can, ”Can B”, was used to act as a general solvent while also containing propellant. It contains 50-75% dimethylether, 20-25% butanone, 20-25% acetone and 2,5-5% 2-methoxy-1-methyl acetate. This can contains less of the propellant dimethylether and more of the solvent acetone. It also contains the solvent butanone. While Can A is mostly dimethylether, Can B is meant to more closely represent the content of a regular can of spray paint without the actual paint.

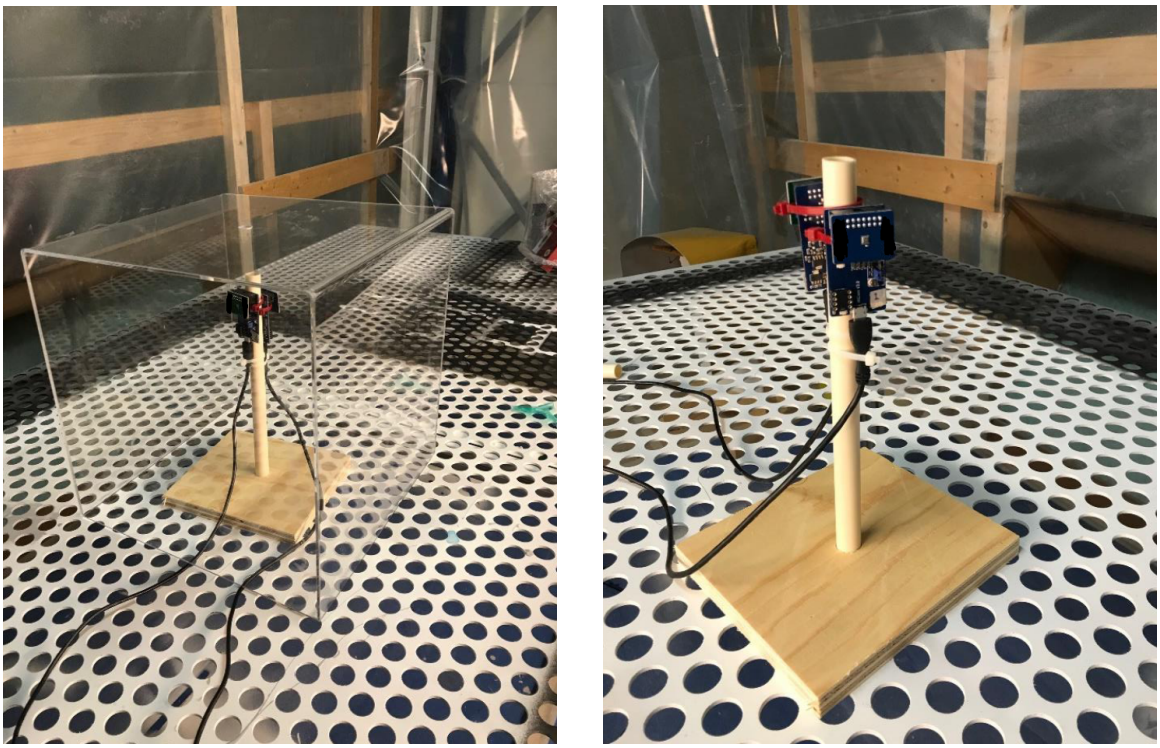


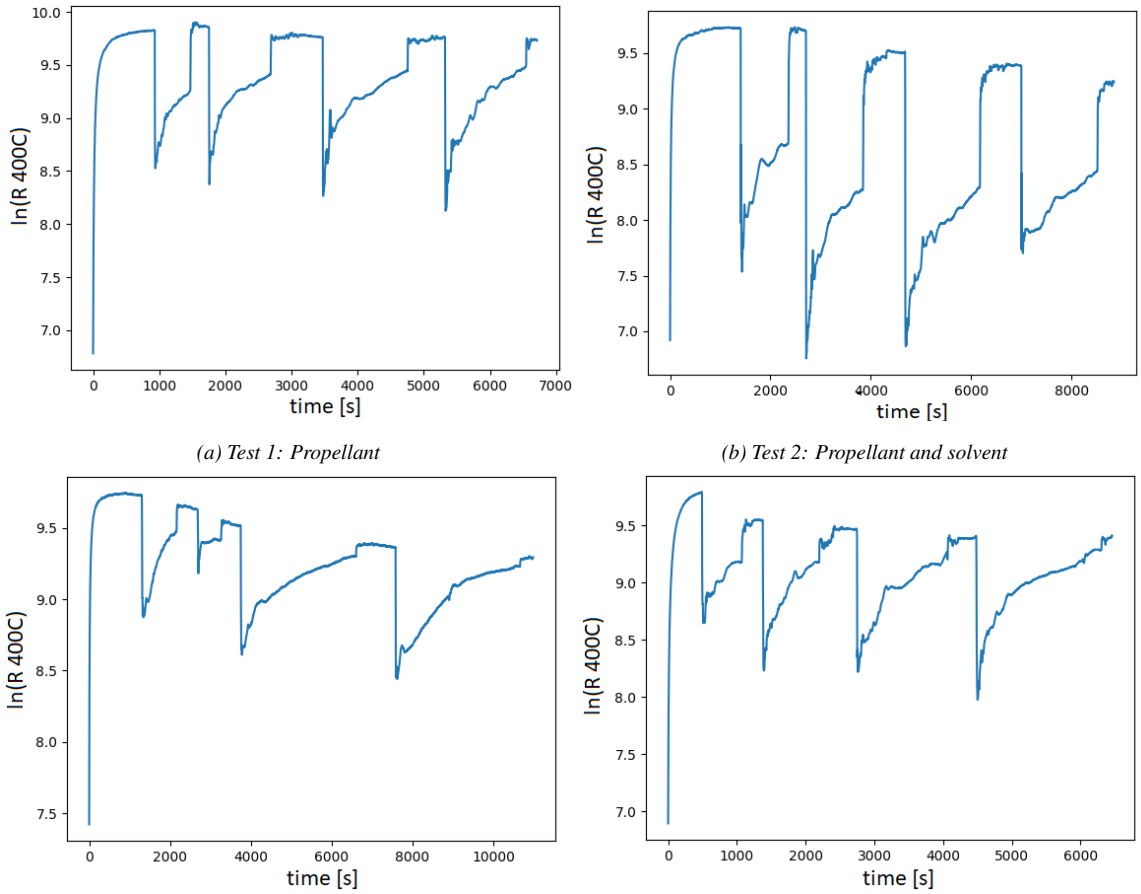
Figure 10: Sensors placed inside a box. Gas is applied from under the box through the perforated plate and stays a long while before dispersing due to limited airflow.

Experiments were done with the following setup: a 300×300×300 mm box with an open bottom and a $12mm$ diameter hole on top was placed on a perforated plate. A sensor was placed inside the box, close to the center, as shown in Figure 10. The sensor was set to sweep over temperatures between $100°$C and $400°$C taking resistance measurements on every 50th degree.

If the spray cans are sprayed on a piece of paper the propellant will be released at the time of the spray and disperse quicker than the solvent that remains on the paper. This is the reasoning when doing the following experiments which were performed on four different days:

- Test 1: Propellant. Can A is sprayed on a paper underneath the box for 1, 3, 5 and 10 seconds respectively. After each spray the paper is removed and the sensor is allowed to reset to baseline.
- Test 2: Propellant and solvent. Can B is sprayed on a paper underneath the box for 1, 3, 5 and 10 seconds respectively. After each spray the paper is removed and the sensor is allowed to reset to baseline.
- Test 3: Solvent. Can B is sprayed for 3 seconds on a paper in another room. The paper is then waved under the box for 1, 3, 5 and 10 seconds respectively. After each waggling the paper is removed and the sensor is allowed to reset to baseline.
- Test 4: Solvent then propellant. Can B is sprayed for 3 seconds on a paper in another room. The paper is then waved under the box for 1, 3, 5 and 10 seconds respectively. After each waggling Can A is

sprayed for 2 seconds on another paper underneath the box. Both papers are then removed and the sensor is allowed to reset to baseline.



(a) Test 1: Propellant

(b) Test 2: Propellant and solvent

(c) Test 3: Solvent

(d) Test 4: Solvent then propellant

Figure 11: Log-resistance curves at $400°C$ taken from four initial tests using the MOx sensor.

What is shown in Figure 11 are logarithmic resistance values from a sensor plotted against time. We can clearly see that resistance values dip when exposed to the contents of either Can A or Can B in all tests. This indicates that the sensor is sensitive to the contents of both cans. The results from Test 3 as shown in Figure 11c imply that the sensor is sensitive to some component in Can B which is not the propellant, though this does not preclude that the sensor might also be sensitive to propellant. In an attempt to distinguish propellant from solvent resistance values at different temperatures were compared.

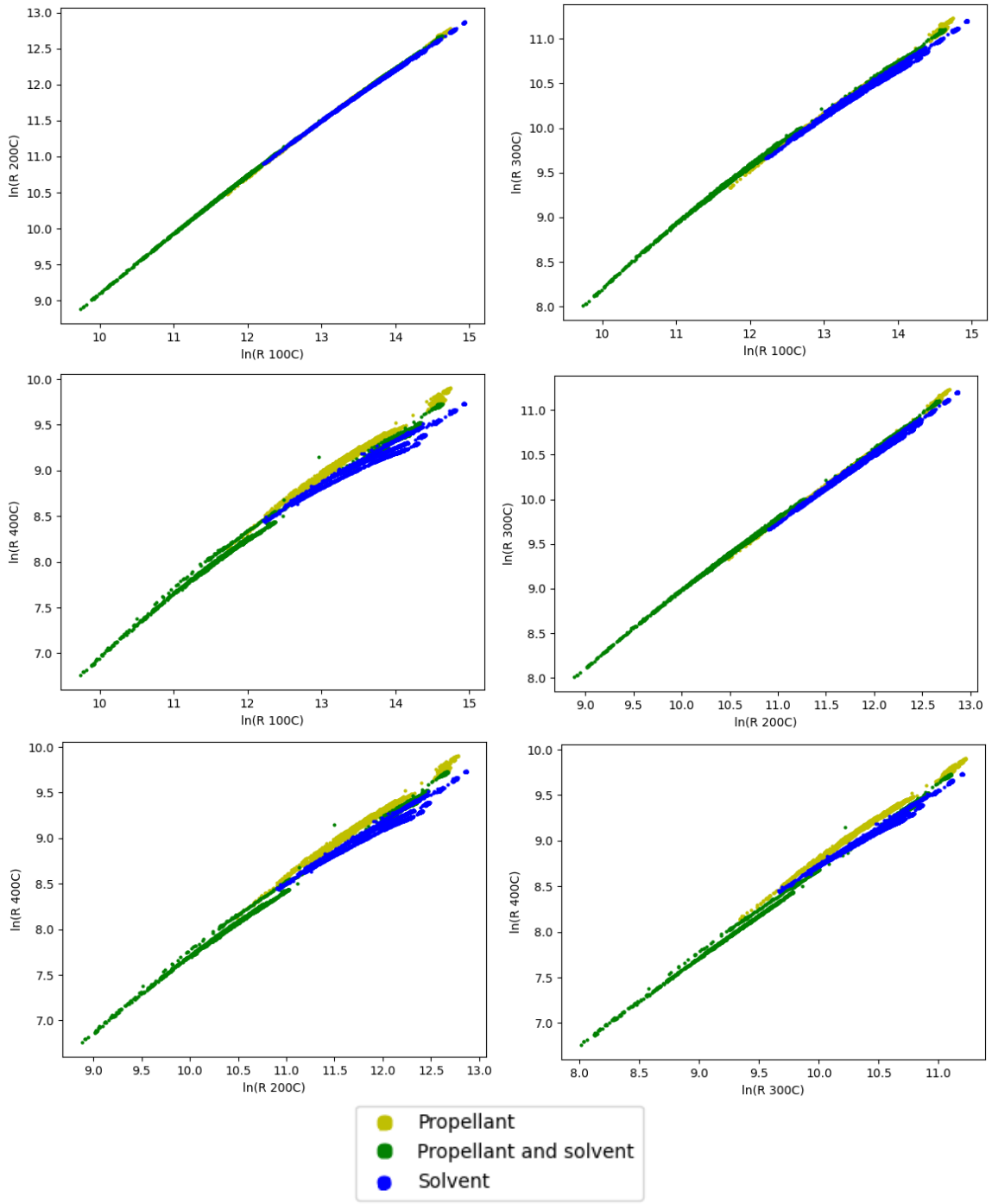


Figure 12: 6 correlations between resistance values at four different temperatures. Data taken from tests 1, 2 and 3. Yellow is propellant, green is propellant and solvent and blue is solvent.

The plots from Figure 12 were made using data from Tests 1, 2 and 3. What we hoped to see when comparing

resistance values at different temperatures for different gas compounds is separation. Separation in this context would mean that the differently coloured dots do not lie on the same line in Figure 12. In the figure 12 there does not seem to be any separation between propellant and solvent data at low resistance values, which is where one would expect the separation to be most pronounced as that is where the concentration of the interesting gas is at its highest. This would mean that the points would be separated at around the middle of the curves, where there are measurements from all three classes.

There is however some kind of separation for high resistance values, which is close to or at the baseline where no or very little gas is interacting with the sensor. The reason for this is that the tests were done over several hours and the baseline shifted over that time. The baseline is also slightly shifted because of the different environmental conditions for the different days, such as temperature and humidity. This would result in a misalignment in the top right in the graphs in Figure 12. This phenomena is therefore not useful in distinguishing propellant from solvent.

We now know that the reason there is no separation in this data is because the solvent acetone, which the sensor is extremely sensitive to, is present in both Can A and Can B. It is unknown if the sensor is sensitive to the propellant used, dimethylether. We did not know this at the time and theorized that we might be able to separate propellant and solvent by analyzing minima in resistance plots over time. Specifically we hoped that the minimum value, at the peak, together with the minimum derivative before a peak and the maximum after within the span of a few seconds might provide separation. Other studies have been made previously which support this hypothesis, as mentioned in Section 2.1.4. We were hoping that either one or both of the derivatives might be unusually high for propellants as they spread and vanish much more quickly than most gases and fumes.

3.2 More experiments on propellant and solvent

More measurements were carried out in a similar manner as the previous experiment. The same setup with the box as shown in Figure 10 was used to contain the aerosol in the box to have as much of a controlled environment as possible. The objective was still to separate propellant, solvent and the combination of the two. We needed more than four samples of each class to draw statistically significant conclusions from the data. The same cans A (for propellant) and B (for propellant and solvent) were applied to the sensor in the same way as the previous experiment. The difference being that there were more variation in the distances of the paper and the sensor, sometimes being up to 50 cm away and sometimes being right next to the sensor.

There were difficulties in having the aerosol interact with the sensor, occasionally to the point where there would be no reaction at all. In particular when Can A was sprayed 50 cm underneath the sensor there was no resistance change distinguishable from noise. We considered that this could be due to a lack of airflow into the sensor or that the room was getting contaminated over time. In general, Can B provided more significant readings than Can A.

At this point we knew that the raw resistance data we were using was insufficient to separate propellant from solvent. Some more complex features were needed for the separation. There seemed to be a difference in derivative between propellant and solvent tests. For every one resistance minimum resulting from a gas exposure, three features were extracted:

- The resistance value of the minimum divided by the baseline
- Approximation of the negative derivative before the minimum
- Approximation of the positive derivative after the minimum

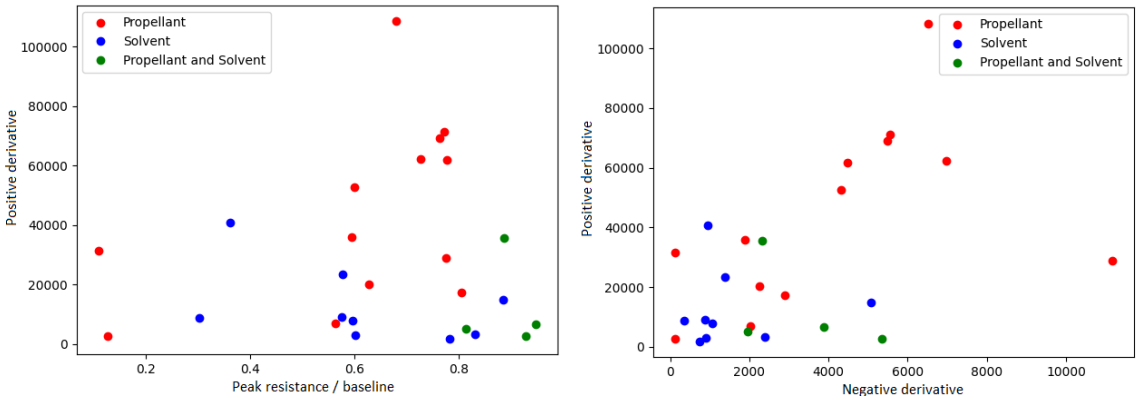


Figure 13: Approximated derivative after a minimum plotted against either (a) the resistance value of the minimum divided by the baseline or (b) the approximated derivative before the minimum.

Taking these features at every measured instance of gas being sprayed resulted in the relations in Figure 13a and Figure 13b. These figures seem to indicate that there is some separation for the different classes using these features, though we can not draw any conclusions from such a small sample. It does seem possible that this approach may separate the different classes in some manner, although not anywhere near close to good enough in this state.

At this point we did have reasons to believe that this kind of separation is not optimal. First off, all three features are dependent on the concentration of the sprayed gas. Classification should be made for any unknown gas concentration as our goal is to use this sensor outdoors in applications where concentration can not be controlled. Secondly, it is likely that these features are not intrinsic properties of the gas itself but instead an artifact of the way the gas was applied. For example, spraying the gas close to the sensor will yield lower resistance peaks and possibly larger derivatives than simply placing a pre sprayed piece of paper beneath the sensor. These assumptions combined with the fact that we had some trouble getting the gas, especially propellant, to reach the sensor led us on the path to do some more realistic experiments.

3.3 Different substances on wall indoors

In order to increase airflow compared to the box we cut up a plastic tube and hung two sensors inside. We set this up on a wooden movable wall and attached another sensor directly to the wall next to the tube. Notably, the two sensors inside the tube are of different models, meaning they will likely react in slightly divergent ways to any gas. One sensor was also put outside as a comparison. The wall was set up in the same environment as the box we used for previous experiments so the atmosphere is consistent between experiments. Using this setup we wanted to figure out at what range we can expect the sensors to operate and also if restricting the airflow by use of the tube impacts results. We eventually want to be able to use these sensors outside where they need to be protected from the elements so it is important to figure out how much we can restrict the airflow without negatively impacting results. Pictures of this setup follows in Figure 14.

Figure 14: The wall. The right image shows how our sensors were set up for these experiments. This setup is meant to resemble realistic conditions more closely than the box. Two different kinds of sensors are inside the tube and one is outside. This lets us compare the different sensors while also finding out if having sensors in a makes for different results compared to having them out in the open.

We were also interested in observing how the sensors react to different substances. Substances with strong smells are particularly interesting to test with this kind of sensor. Initially we were curious about ethanol, hair spray and window cleaner containing ammonia. However, as we were experimenting we kept trying as many different substances as time would allow, testing whatever we had on hand at our test site. We did tests over several days using the wall indoors.

First we did some tests with Can A (propellant) and Can B (propellant and solvent) as before. We found that there are difficulties getting results when spraying even a short distance away from the wall. The propellant did not even show up when sprayed a meter away from the sensors. We found no significant difference between the results of the sensors in the and the sensor next to the tube.

We then did tests with ammonia-based window cleaner and hairspray. Both gave significant readings on the sensors, similar to what we had seen with the spray cans previously. At this point we were still unsure of what features to use to best differentiate what the sensors react to.

Our main issue, which would have to be resolved before any other at this point was that we still had significant issues getting the sensors to pick up anything at all. Spray paint, which the sensors are very sensitive to, had to be sprayed only up to about a meter away to be picked up and would be inconsistent even then. This gets worse as over time as the air becomes saturated with fumes and the baseline drifts and sensitivity decreases.

3.4 Realistic measurements outdoors

The wall was moved outdoors to see if having cleaner air would make it easier for sensors to interact with the gas. Another reason for moving the wall outdoors was that having the sensors work under these conditions was the stated goal of this project. The wall outdoors is shown in Figure 15.

Figure 15: The wall outdoors. There are four boxes each containing two different sensors. This setup lets us test in realistic conditions.

Moving the wall outdoors provided a significant change in results. Compared to indoors, the sensors became more sensitive to every gas. The range of detection was increased and the obtained resistance curves had less oscillations. Another difference in doing tests outside was that one spray of spray paint oftentimes resulted in multiple peaks. There was one initial peak shortly after the spray occurred followed by multiple peaks as fumes from the spray paint naturally flowed to the sensor. We believed that this improved sensing had two explanations. First, the surrounding air is much cleaner. Second, there is more natural air flow outside compared to a closed room.

Lots of measurements were taken with graffiti spray paint and other substances such as ethanol and hairspray. After exploring different settings of the sensor, switching which temperatures to take measurements at and increasing the sampling rate, we noticed something we had not seen before. Plotting all resistance values for a high temperature against the resistance values in a lower temperature results in clearly separated clusters for different substances.

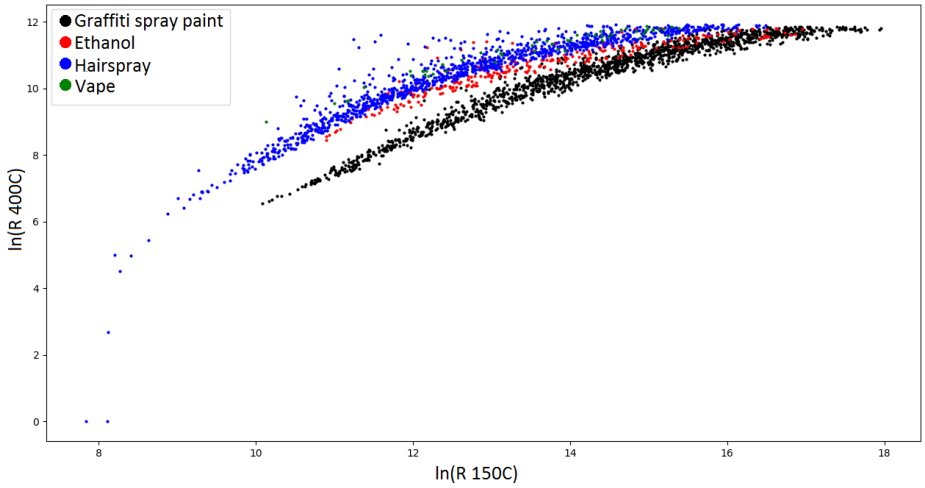


Figure 16: Log-resistances in 400°C plotted against resistance values in 150°C for different substances.

This behaviour is seen in Figure 16 where the different measured substances are in clearly separated curves. This was a small breakthrough and something that utilized the different gases sensitivity in different temperatures. This seemed like intrinsic properties of gases expressed in a way that could be fed into machine learning methods. Data in this form was something that we ended up using in our final classifiers.

More measurements were carried out outdoors. There was a lot of experimentation with different settings on the sensor. Deciding which temperature levels to use in the temperature sweep was particularly important. Registering values at more temperatures result in higher dimensional data points at the cost of lower sampling rate. In temperature sensitivity graphs such as Figure 16, we tried to maximize the separation between all measured temperatures at all points throughout the curves. After a lot of experimentation some settings that looked satisfying were found. We committed to the temperature sweep with four temperature levels: 150°C, 250°C, 350°C, and 450°C.

We also did experiments with additional substances to get a better understanding of the sensor response to spray paint. Two of the most popular solvents in graffiti spray paints are acetone and xylene. The sensor is only mildly sensitive to xylene but extremely sensitive to acetone. In fact, out of all substances that were tested acetone was the only substance that gave somewhat similar curve as spray paint in temperature sensitivity plots such as Figure 16.

Because every spray of spray paint results in multiple resistance minima (for example as in Figure 4) we also tried to separate the initial points after every spray from the subsequent points. The idea was that the subsequent points consisted of no propellant from the initial spray as opposed to the first points. We found no discernible difference in the temperature sensitivity between these two sets of points. This, combined with the sensor response to acetone, led us to believe that it is very hard to differentiate the propellant and solvent with this kind of sensor, as per our initial idea. However, it looked very promising to separate spray paint from all other substances using the temperature sensitivity and possibly combine it with our earlier findings of derivatives. Thus it was time to start collecting data and build a real dataset.

4 Methodology

This part describes what we have done and motivates why it has been done when going from data collection to machine learning classification. Section 4.1 shows a detailed design of the experimental setup where the data was collected and the process of the data collection is described in Section 4.2. Section 4.3 is about the interpretation of the form of the data and which features that are to be used. Subsequently Section 4.4 is about processing the data and selecting the important parts of the data to be included in the dataset. Section 4.5 is about the choice of machine learning models and parameters that were used in classification. Lastly, the classification was also done in real time which is described in Section 4.6 showing how it could work in practical applications.

4.1 Experimental setup

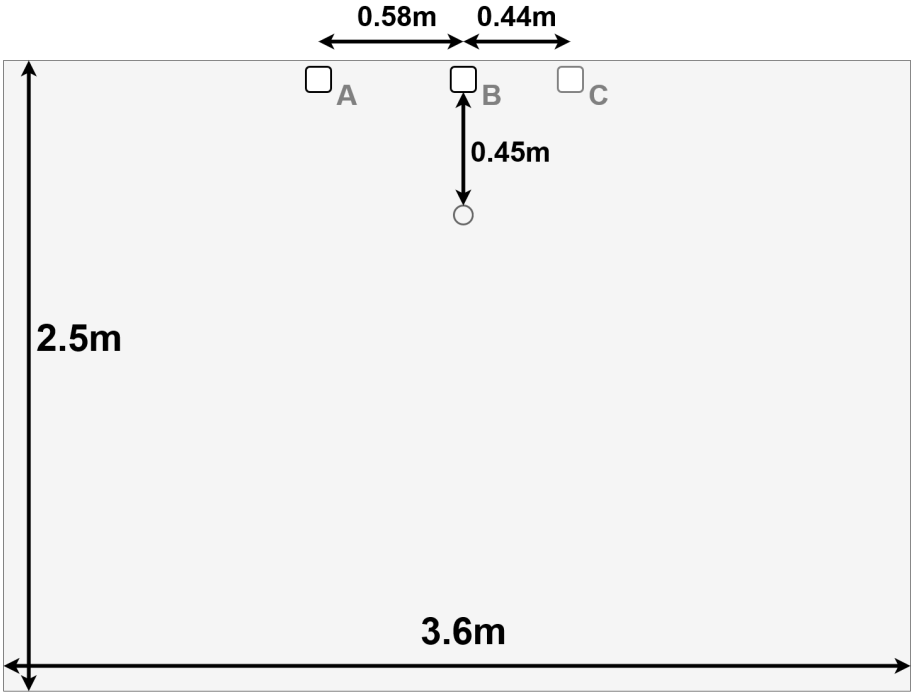


Figure 17: Diagram of the wall including distance measurements. A, B and C are positions of the boxes containing sensors. The circle beneath B is where a hook was positioned on which we hung liquid-soaked rags holding for example ethanol and window cleaner.

The data collection was done by mounting sensors at the top of a large wall, depicted in Figure 17. Sensors were placed on position A, B and C in the diagram. In the figure the placement of a hook is marked underneath the sensors. Each sensor was placed in a small case with dimension $60\times60\times30$ *mm*. The top and bottom of the cases where open for the gases to flow through.

4.2 Data collection

The main data set consists of measurements from three different days. All data collection was done using the experimental setup with the wall outdoors from Section 4.1. The data was measured outside because of the resemblance of realistic conditions and the cleanliness of the air. Many different substances were applied to the sensor, chosen based on if they had strong reactions with the sensor and substances which could realistically interact with the sensor in real world applications. Because of the nature of the substances they were applied in various ways to the sensor. Note that there was little consideration for consistently applying substances in the same ways. The reason for this is that we want the sensors to work on intrinsic properties of whatever they are sensing and randomizing application reduces the chance that a classifier becomes overfitted to irrelevant data. The following lists shows which substances were measured and in which way they were applied to the sensor:

- Graffiti spray paint (Montana Black) - Was sprayed on different positions on the wall. The spray paint often remained on the wall for a few minutes to half an hour after the initial spray, resulting is multiple successive reactions with the sensor. Various colours of spray paint were used.
- Ethanol - A piece of paper was soaked and hung on a hook in the center of the wall. A bottle was squeezed underneath the sensors.
- Hairspray - A piece of paper was soaked and hung on a hook in the center of the wall.
- Smoke - A stick with a piece of paper soaked in ethanol was lit on fire and waved under the sensors. Also some sticks and pieces of paper were lit on fire in a bucket.
- Window cleaner (ammonia based) - A piece of paper was soaked and hung on a hook on the center of the wall. A bottle was squeezed underneath the sensors.
- Vape - A vape pen was blown against the sensors to produce vape smoke.
- Cigarette smoke - Smoke from a cigarette was blown against the sensors
- Paint removal - A bottle of paint removal was squeezed underneath the sensors.
- Acetone (pure) - A piece of paper was soaked and hung up on a hook in the center of the wall. A bottle was squeezed underneath the sensors.
- Xylene - A bottle was squeezed underneath the sensors.

	Day 1	Day 2	Day 3
Graffiti spray paint	11470	5992	4950
Ethanol	17017	1237	-
Hairspray	12158	2026	-
Smoke	6814	-	-
Window cleaner	9454	1528	715
Vape	1175	-	-
Cigarette smoke	-	402	-
Paint removal	3261	-	-
Xylene	-	-	132
Acetone	-	-	2421

Table 1: Amount of data points for different days in the data set.

Table 1 shows the amount of measured data points for all different substances during the different measurement days. The data points that were included were those that had a logarithmic resistance in 450°C less than 12.55. This is roughly the points where there is a significant decrease in resistance, disregarding noise and measurements at the baseline where no gas is being applied. Three sensors were collecting data simultaneously, placed on position A, B and C in Figure 17 in Section 4.1.

The dates and conditions of the three measurement days is shown in the following table. Note that the conditions were only measured once during the day and could have changed throughout the day.

	Day 1	Day 2	Day 3
Date	201109	201110	101125
Temperature	10.4 °C	7.9 °C	7.6°C
Relative humidity	70.5%	76.0%	80.8%

4.3 Feature selection

4.3.1 Temperature sensitivity

Every measured sample point is measured at four different temperatures: 150°C, 250°C, 350°C and 450°C. The resistance values at all four temperatures can be used to take advantage of the temperature sensitivity of the sensor for different temperatures. Plotting resistance values in two different temperatures result in a plot like Figure 18 where resistance values in 450°C are plotted against resistance values in 150 °C.

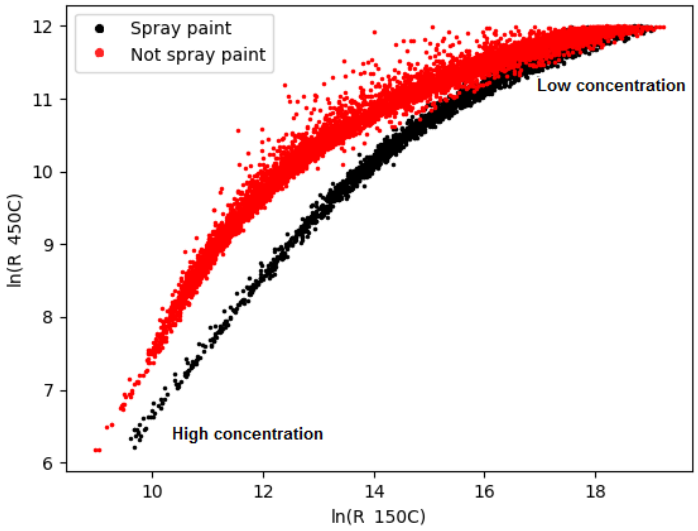


Figure 18: Log-resistance values in 450°C plotted against log-resistance values in 150 °C for labelled data with two classes. The 'Not spray paint' class is a mixture of different gases from the dataset.

Plots like Figure 18 should be interpreted as different temperature sensitivity for different applied gases. In the upper right in Figure 18 there is low concentration of gas being exposed to the sensor. This is where the

resistance is high, close to the baseline. Naturally there is less of a separation for low gas concentrations. Increasing the concentration decreases the resistances in the graph, following the curve towards the lower left. A typical spray of spray paint, for example the one in Figure 3 in Section 2.1.3, consists of many sample points, often between 30 and 80 with the sampling rate used in our sensors. These points are spread throughout the curve in Figure 18, from the baseline in the upper right, to an end point in the bottom left and back to the baseline in the top right.

The plot in Figure 18 only shows two dimensions. Every sample point is measured in four temperatures which means that each sample point has four dimensions. When determining the settings of the sensor, such as registered temperatures and sampling rate, we tried to maximize the separation at different points throughout the curves when plotting resistances at different temperatures against each other. The temperature sensitivity is being taken advantage of in the classification where every single four dimensional point is being fed into a machine learning classifier and classified as either 'spray paint' or 'not spray paint'. This means that one single spray of spray paint results in many sample points which are all being classified one by one.

4.3.2 Derivatives

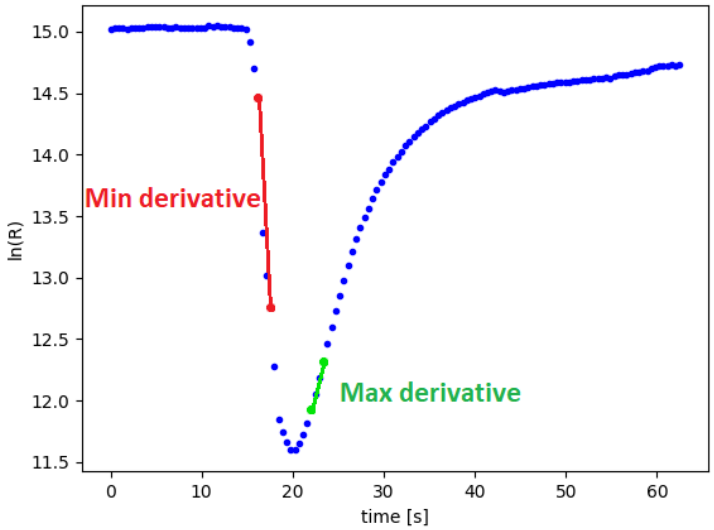


Figure 19: Minimal and maximal derivative on a resistance curve.

Additional features are used in the form of derivatives. The derivatives of interest are called the maximal and the minimal derivative. The minimal derivative is defined as the lowest derivative that exists before the resistance minimum and the maximal derivative is defined as the largest derivative that exists after the resistance minimum. The derivatives are shown in Figure 19.

The sampling speed can affect the position of individual points. Thus a derivative is averaged over four points to give a more smooth estimate of the slope. The derivative of a point is calculated as

$$\frac{x_4(i) - x_4(i-3)}{t(i) - t(i-3)}$$

where $x_4(i)$ is the resistance value measured in 450°C for the point at index i. $t(i)$ is the corresponding time at this index, measured in seconds. It is desired that all points on the same downwards and upwards slope share the same derivatives. That is why the derivatives are calculated on 60 points to the left and on 60 points to the right of each point. The maximal and minimal derivatives are chosen as the maximal and minimal of these derivatives. Most curves from the baseline to a minimum and back to the baseline are in the range of 30-80 points. 60 points on either side of a point is a suitable number of points to take into account in the derivative calculations.

4.4 Data preprocessing

4.4.1 Standardization

Standardizing the data set is crucial because the data is measured on different days. External conditions such as surrounding temperature, humidity and innate resistance of sensors will cause the measured resistance to drift when comparing different days and different sensors, as described in Section 2.1.5.

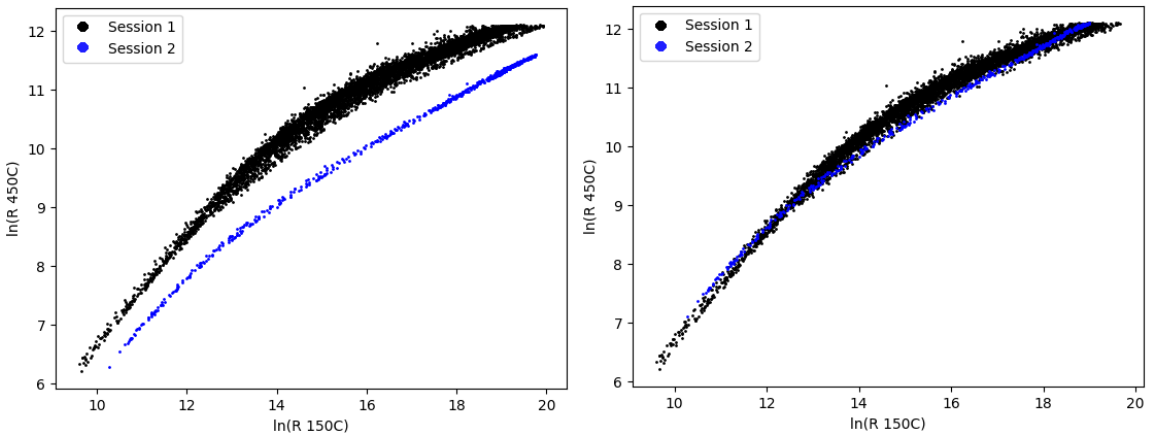


Figure 20: Log-resistances from two different measurement sessions. Session 1 consists of spray paint measurements from one sensor during three days of measurements. Session 2 consists of the same kind of spray paint measured with another sensor in drastically colder and dryer weather. (a) shows resistance values as received by the sensor. (b) shows the same data points linearly shifted using standardization.

Figure 20 shows an extreme case of how the sensor drift could impact the measured resistances in practice. The figure shows measured spray paint from three days of tests during November (Session 1) and some sprays of spray paint in drastically colder and dryer weather in late January, measured with another sensor (Session 2). It is the same type of spray paint used in both sessions so it is desirable that these two curves appear as similar as possible. Note that this is en extreme example to show the point of standardization. The effect is much less extreme for the data that is collected in our dataset.

The log-resistance value in 450°C plotted against the log-resistance values in 150°C, as in Figure 20, suggests almost a linear shift in the ln(R 450C) axis. Using the theory in Section 2.1.5 the points can be standardized with Equation 1:

$$y_n(i) = x_n(i) + c_n$$

where $x_n(i)$ is a log-resistance point at index i. *n* is the index of the temperature, $n = 1, 2, 3$ or 4 depending on if it is measured in 150°C, 250°C, 350°C or 450°C. $y_n(i)$ is the corresponding standardized point and c_n is a standardization constant. Figure 20b shows how this constant has been added to the log-resistance points in 450°C to linearly shift the data points, aligning the two curves.

This standardization is performed for the log-resistances in 250°C, 350°C and 450°C. The goal of the standardization is to align the unshifted data points to be closer to some choice of reference data points. These reference data points were chosen as the spray paint measurements from sensor A during the first day of measurements during the data collection.

The standardization constant c_n is an estimate of the shift of the resistance in that temperature compared to the reference data points. Calculating c_4 for the resistances in 450°C is done by:

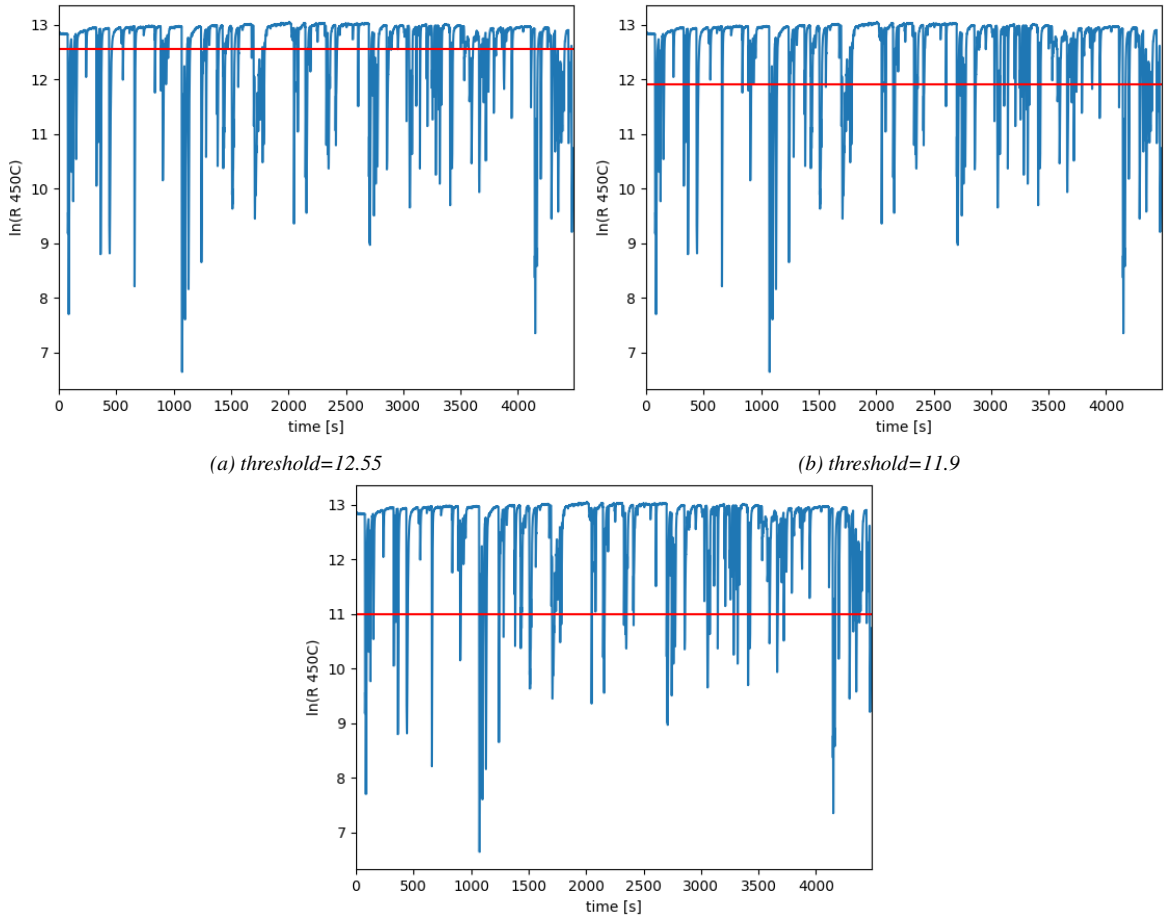
- Choose a constant value of the log-resistance in 150°C near the baseline where all the curves are to be aligned. In our data set we chose 19.0.
- Take the corresponding value for the log-resistance in 450°C for both the reference data points and the unshifted data points.
- Take the mean of all points a distance of 0.1 of the log-resistance in 150°C within this value.
- Calculate the shift c_4 as the difference between this values for the reference data and the unshifted data.

c_n is computed for each measurement performed in 250°C, 350°C and 450°C. The mean is computed because there is variation in the data. For example in Figure 20a, the curve from Session 1 is thick and there are multiple points where ln(R 150C) is equal to 19.0.

The choice where the data is to be aligned for the log-resistances in 150 °C was chosen as 19.0. This is a point which is close to the baseline. Because the external conditions affect all resistance values, including the baseline, the idea is to align the baselines for different sessions with differing external conditions. The standardization is also a form of baseline correction. This method also shares the concept of the differential standardization defined in Section 2.1.5 but the value of the baseline is from a previous reference measurement.

4.4.2 Thresholding

It is not ideal to classify every measured data point. It is necessary to set thresholds for which data points to include in the classification. Many of the data points will lie on the baseline which are of no interest to classify because there is no applied gas to the sensor. Furthermore, points that lie very close to the baseline are harder to differentiate because there is very little gas reacting with the sensor. These points are more interesting to classify when comparing different machine learning models. Optimizing classification of low gas concentration points are important for quicker and more sensitive spray paint detection.



(a) threshold=12.55

(b) threshold=11.9

(c) threshold=11.0

Figure 21: Threshold set at different levels for log-resistances in 450°C for one session of spray paint data

The classifications that are evaluated in this thesis are done using three different thresholds. One threshold is set fairly close to the baseline, determined by points with log-resistances in 450°C less than 12.55. This is slightly under the noise level of the baseline. Another threshold is determined by points with log-resistances in 450°C less than 11.9. This results in classifying data points where there is a significant amount of gas reacting with the sensor, showing classification performance in slightly nicer conditions. The third threshold is set fairly far away from the baseline, where the log-resistance in 450°C are less than 11.0. This is when there is a large amount of gas reacting with the sensor. The thresholds are set for the standardized data and are shown in Figure 21.

4.4.3 Dataset

The data is collected on three different days so the data is divided into training data from Day 1 and test data from Day 2 and Day 3. This results in the following split of data points in Table 2 where the threshold is set at different levels:

	Training data points	Test data points
Graffiti spray paint	11470	10942
Other substances	49879	6040

(a) threshold=12.55

	Training data points	Test data points
Graffiti spray paint	8394	5718
Other substances	30922	2781

(b) threshold=11.9

	Training data points	Test data points
Graffiti spray paint	5005	2814
Other substances	19804	1381

(c) threshold=11

Table 2: Number of data points in the dataset for different choice of thresholds.

This distribution of training and test data is fair choice because the testing is performed on points on slightly different environmental conditions that has not been trained on. All three sensors have also been used in simultaneous data collection. All this has been done to not make the dataset unrealistically easy to classify.

Note that acetone has not been included in the above dataset. We have instead constructed a new separate dataset where acetone is included. The reason for this is that acetone is the component of spray paint which the sensor is most sensitive to. Acetone was only recorded on the third day of measurements. In this new separate dataset roughly half of the acetone data points are being included in the training data while the rest are being included in the test data. This means that for the training data, 1182, 740 and 444 acetone points are added for the threshold 12.55, 11.9 and 11.0 respectively. For the test data, 1239, 866 and 546 points are added for the threshold 12.55, 11.9 and 11.0 respectively. This points are being added in addition to the data in Table 2. This dataset is only being used when it is specifically specified that acetone is being included.

Another important part of data preprocessing that can be used to decrease false positives is *class weighting*. This means that the classes can be assigned specific weights if they are not of equal importance. For example, weighting the 'not spray paint' class twice as high as 'spray paint' will force the classifiers to correctly classify the 'not spray paint' class more than the 'spray paint' class. In practice this results in less false positives. The downside of this approach is that the total accuracy will decrease because the data has been manipulated to not be optimized over accuracy. That is why it is not used for all the results but only in a part of Section 5.4.

In the last step of the preprocessing the data needs to be scaled to have zero mean and unit variance. This means that every point is subtracted by the mean and divided by the standard deviation of all data points. This step is especially important for the SVM to function but is also done for overall computational purposes.

4.5 Model selection

Machine learning methods, especially supervised learning where you are working with labelled data, is a suitable approach to differentiate the different gases since all of the data that has been collected has been labelled. Also there is no explicit model at hand describing the behaviour of the measured resistances. This is not needed because the machine learning methods can find relevant patterns in the data. The choice of specific methods is based on the machine learning theory in Section 2.2.

The k Nearest Neighbour classifier was chosen because it works well when there is much data available with few dimensions. These criteria are fulfilled by our collected and preprocessed data. At its core the data set consists of only four dimensions. This simple yet powerful method is a promising choice.

The Support Vector Machine was chosen because it fundamentally works by determining a linear decision boundary that separates data points. Plotting log-resistance values against each other for different temperatures (for example as in Figure 18 in Section 4.3.1) there is a clearly visible decision boundary between the two classes. This is an indication that a SVM is a suitable method for this kind of data. Also, SVM is a common method with proven success on similar classification tasks, as described Section 1.3.

The neural network is a good choice for classification tasks because of its great generalization ability and proven success on similar classification tasks. Compared to the SVM, neural networks are more flexible with many different designs and parameters to vary.

The kNN had the easiest hyperparameter to optimize since it only had one, k, the amount of nearby points to take into consideration. This was chosen as k=5 but any choice in the same range would have resulted in similar results. The SVM was chosen with an RBF kernel and $C = 20$. A larger C generally resulted in better performance metrics but it was limited to 20 for a more generalizable model. The γ parameter was chosen as $1/(N \cdot \text{var(X)})$ where N is the number of features (four or six) and var(X) is the total variance of all training data points.

The architecture of the neural network was chosen with four hidden layers with 48 nodes in the first layer, 36 nodes in the second layer, 24 nodes in the third layer and 12 nodes in the fourth layer. In between the hidden layers there were three dropout layers with a dropout of 0.5, 0.4 and 0.3 respectively. The loss function was binary cross entropy which was minimized with stochastic gradient descent with a learning rate of 0.3. This structure was chosen as it performed fairly well on the data for all threshold.

4.6 Real time classification

The classification was also performed in real time to show how it would work as a detector in real life applications. It was tested by applying different substances near the sensor, including graffiti spray paint, ethanol, hairspray and window cleaner. When it classifies the measured points as spray paint or not spray paint it shows a positive or negative image on the screen depending on the result of the classification.

The idea of the real time algorithm is to continuously evaluate points from the sensor. If the resistance value of the measured points falls below a set threshold, all the points up to a maximum of 40 will be individually classified. If 75% of the points are being classified as spray paint it is concluded that spray paint was being sprayed near the sensor. Otherwise the conclusion is that another substance was reacting with the sensor. This gives room for some misclassifications of individual points while still maintaining a high overall accuracy.

Pseudocode for the real time classification is shown in Algorithm 1. 4-dimensional resistance points x are continuously received from the sensor via the *receive()* function. $x[4]$ is equal to the resistance value of the point at 450°C. Points below the threshold are classified with the pretrained machine learning classifier *clf* which outputs 0 if the point is classified as 'spray paint' and 1 if it is classified as 'not spray paint'. The points also need to be scaled to zero mean unit variance with the same scaler used on the training data. We used a SVM classifier and classifying points below 11.0 when testing the real time classifier.

Algorithm 1 Real time classification without standardization

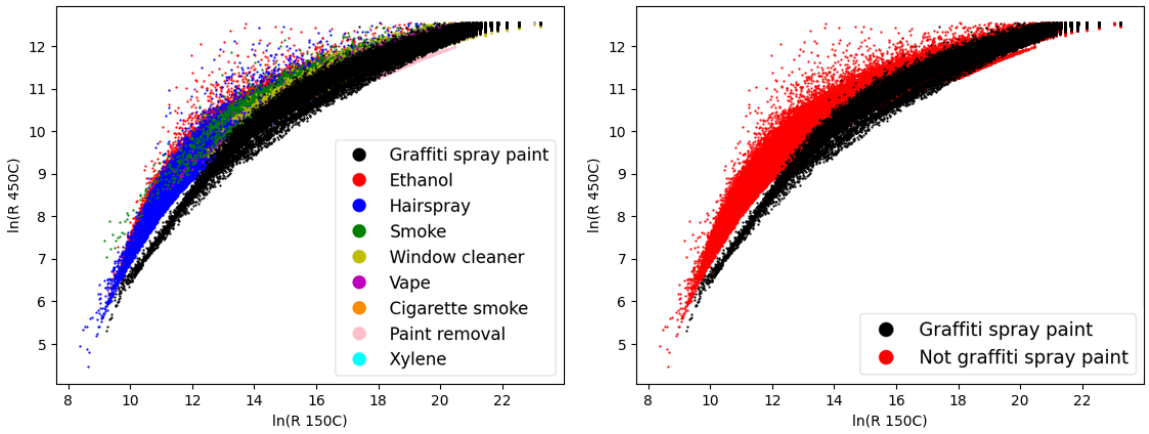
```
while True do
    x ← receive()
    if x[4] < threshold then
        points ← []
        for i in range(40) do
            x ← receive()
            if x[4] > threshold then
                Break
            end if
            points.append(x)
        end for
        predictions ← []
        for x in points do
            x ← scaler.transform(x)
            predicted_class ← clf.predict(x)
            predictions.append(predicted_class)
        end for
        if predictions contains > 75% 0:s then
            Alarm! This was classified as spray paint!
        else
            This was not classified as spray paint.
        end if
    end if
end while
```

5 Results

This section is focused on the results of the classification using the dataset that was established in the previous section. In the first part the measured data points are visualized, showing what the data looks like. Subsequently standardization is applied to the data and the transformed dataset is visualized. In Section 5.3 classification is performed on 4-dimensional points only consisting of resistance values at different temperatures. In Section 5.4 derivatives are included in the data points. In Section 5.5 the impact of individual substances is shown as well as the effects of including acetone in the dataset.

5.1 Data visualization

Figure 22 shows the points of all different substances from all three sensors measured on the three measurement days. Figure 23 shows the same data points but all six interactions between the four resistance values. Afterwards, Figure 24 displays the same thing but only the points from one of the sensors. Directly measured resistance curves are shown in Appendix A which are the basis of the data in this section.



(a) *(b)*

Figure 22: All measured data points with a log-resistance less than 12.55. (a) has assigned colours according to which substance is measured. (b) is coloured according to the binary classifier.

Figure 22 shows the data from every sensor collected in one graph. Note that (a) and (b) contain the same data points, the only difference between the two subfigures being that (b) defines every non-graffiti data point as 'not graffiti', which is how the data is treated in our machine learning algorithms. While the data is separated at high gas concentrations, showing as low resistance values, toward the bottom left of the graph, the point clusters start to overlap for $\ln(R\ 150C)$ values over 12.

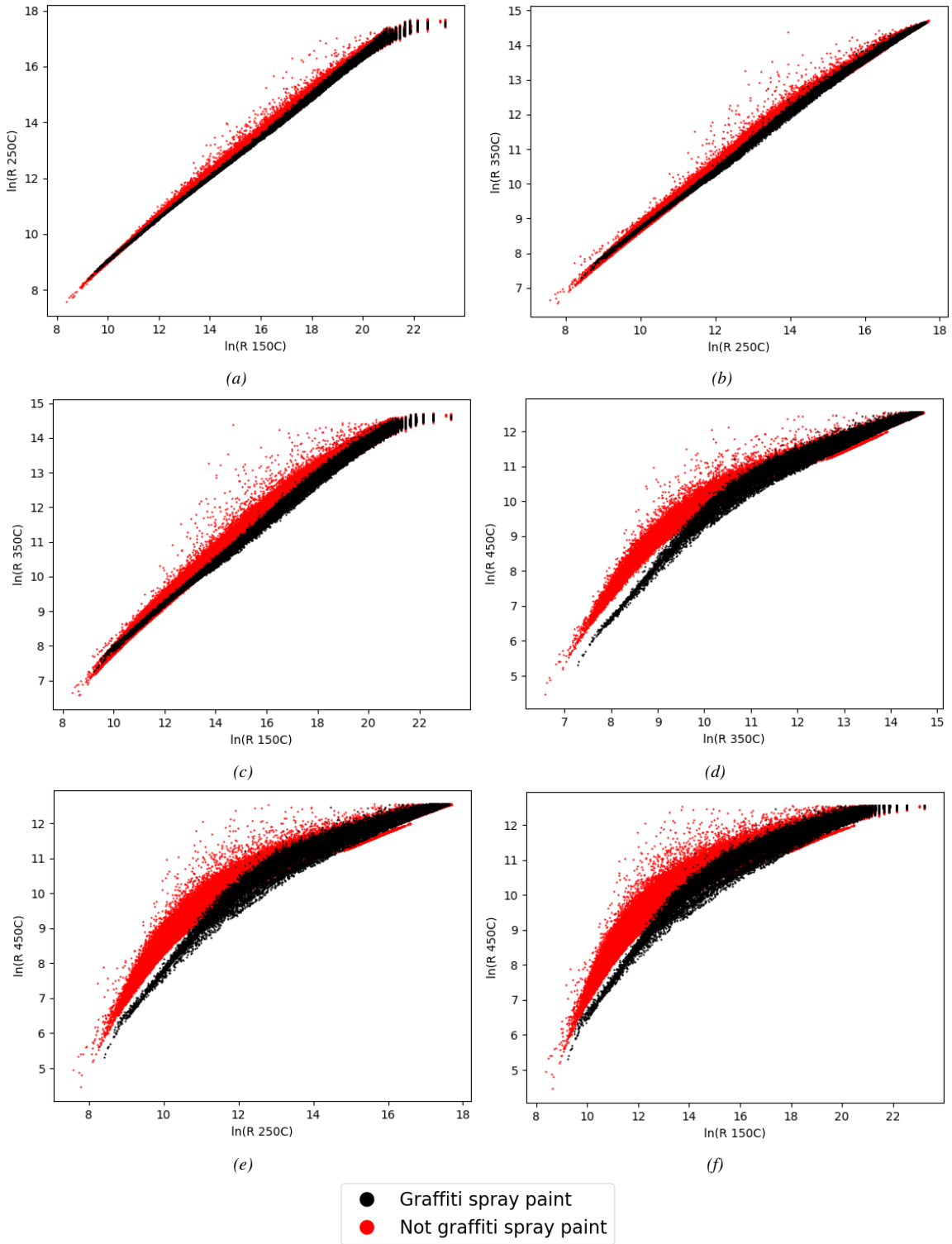


Figure 23: The 6 correlations between all measured resistance values at 4 different temperatures.

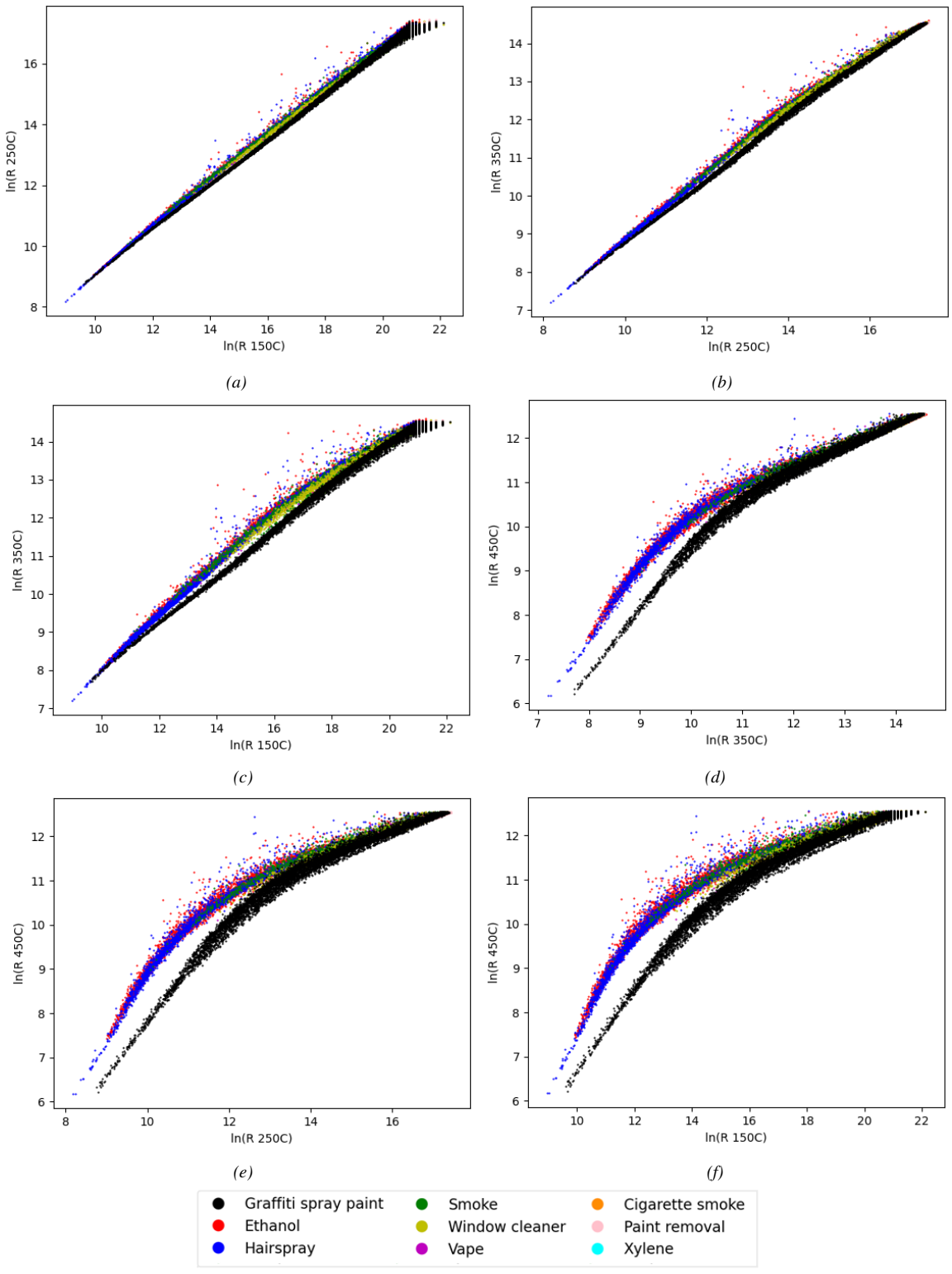
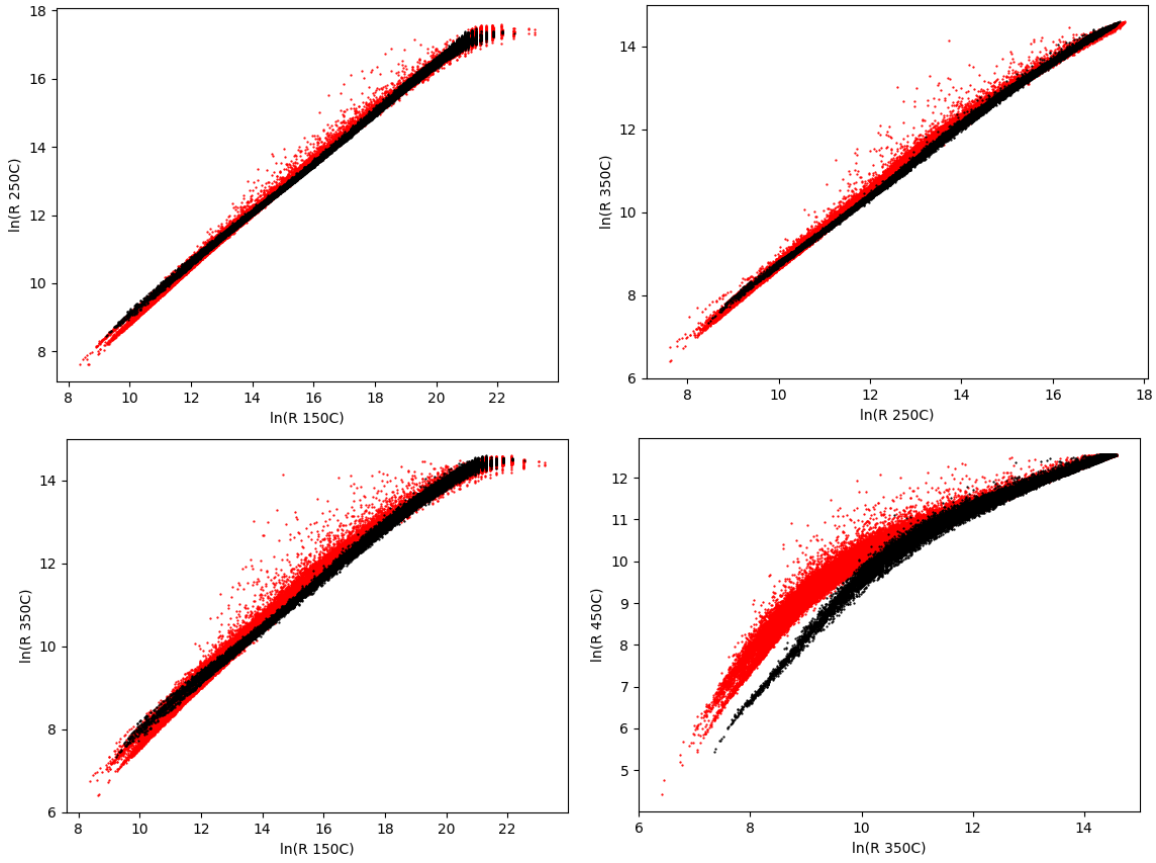


Figure 24: The 6 correlations between measured resistance values at 4 different temperatures from only sensor A during all three measurement days.

Figures 23 and 24 show how significantly different results from sensors of the same model can be. Figure 23 contains every temperature sensitivity plot we can make from the collected data using *all 3 sensors*. Figure 24 shows the data collected from only *one sensor*. The difference is significant. Looking at Figure 23f we can see that there is only a small range where the clusters are completely separated, blending together for values over ln(R 150C)=12 as well as merging slightly toward the bottom where resistance is at its lowest. If we instead look at the data from only one sensor as in Figure 24f the clusters do not ever blend for ln(R 150C) values below 15 and there is no merging at all for extremely low resistance values. Further, there is some separation for lower temperature comparisons when using only one sensor, as shown in Figure 24c which is not the case at all in Figure 23c.

5.2 Effects of standardization

Figure 25 shows all points of all different substances from all three sensors measured on all three measurement days, same as the previous Figure 23. In this case the data points have been standardized with the method from Section 4.4.1. Subsequently a comparison between standardized and not standardized data points is shown in Figure 26.



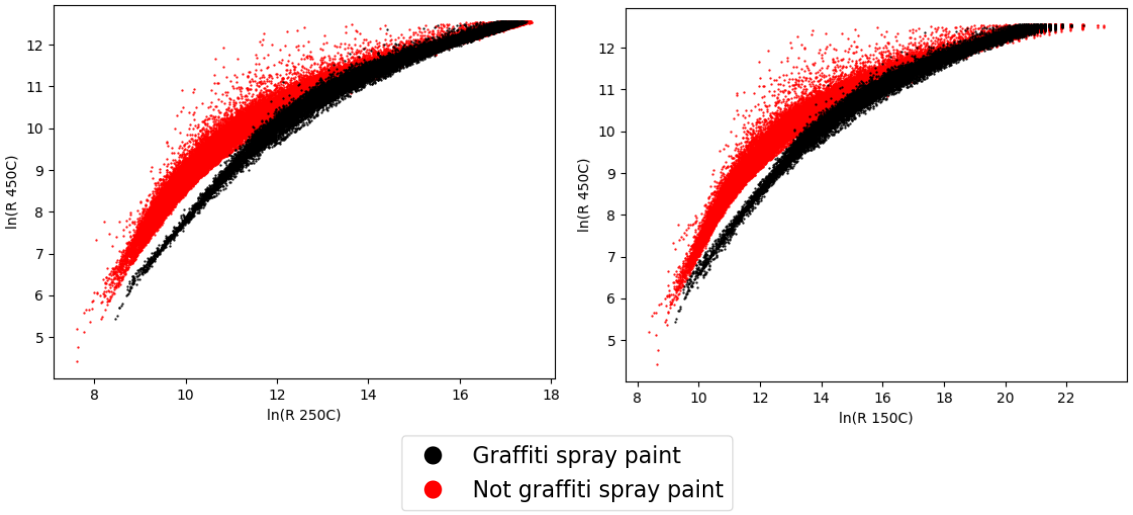
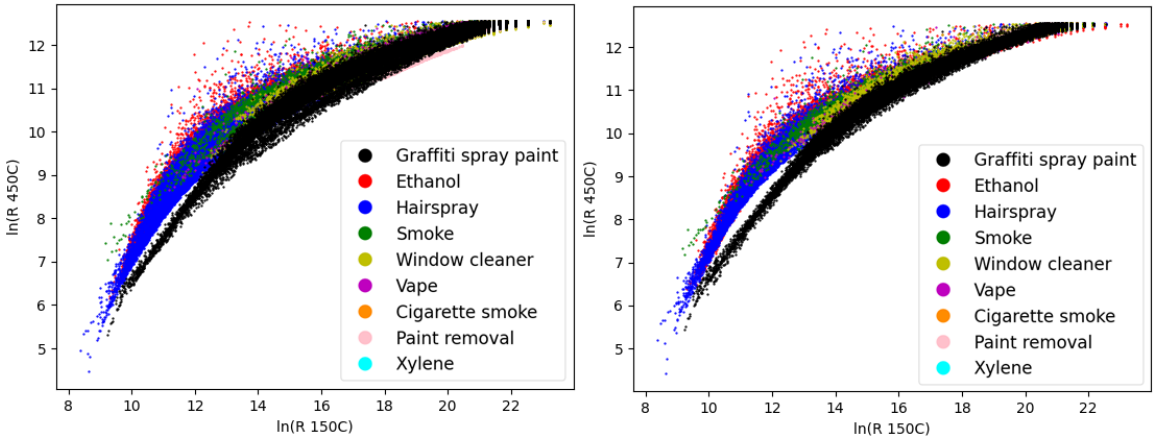


Figure 25: The 6 correlations between standardized measured resistance values at 4 different temperatures.



(a) Not standardized

(b) Standardized

Figure 26: All measured data points with a log resistance less than 12.55.

The goal of standardization is essentially to transform data from what is presented in Figure 23 to what is presented in Figure 24. Our best attempt can be seen in Figure 25 and while the data points are not as separated as in Figure 24 they are more separated than in Figure 23. A direct comparison between the not standardized and standardized data is shown in Figure 26. Note that the only significant difference being compensated for in this example is out-of-the-box differences between different sensors as well as some slight changes in weather between the three consecutive days. Recording data in drastically different temperatures or humidity adds even more noise. An extreme example of this is shown in Figure 20a where the two sessions are done at different locations months apart with different sensors, one of which might be damaged from use.

5.3 Classification

All the classifications in this part are done on 4-dimensional points consisting of measured resistances at four temperatures: 150°C, 250°C, 350°C and 450°C. Classification is done using three machine learning classifiers. The classification is performed on three variants of the dataset when including points under a low, medium or high threshold as constructed in Section 4.4.2. The results of the SVM is shown in Table 3, results of the kNN is shown in Table 4 and results of the neural network in Table 5. The total accuracy for the different classifiers as a function of the threshold is shown in Figure 27.

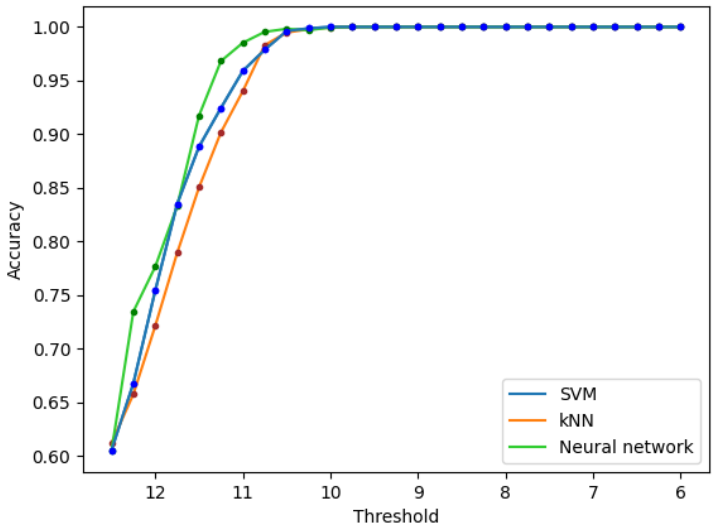


Figure 27: Total accuracy of classifiers for varied threshold between 12.5 and 6.

Note that the performance of the classifiers is perfect for lower thresholds in our dataset as seen in Figure 27. The subsequent results are focused on the thresholds where the differentiation is harder to make an interesting evaluation of the classifiers.

5.3.1 SVM

The results in this part is obtained with an SVM with the parameter $C = 20$ and an RBF kernel.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.97	0.38	0.54	0.97	0.71	0.82	0.99	0.95	0.97
Not spray paint	0.46	0.98	0.63	0.61	0.95	0.75	0.90	0.98	0.94
False positive rate	0.02			0.05			0.02		
Total accuracy	0.59			0.79			0.96		
Confusion matrix	$\begin{pmatrix} 4132 & 6810 \\ 137 & 5903 \end{pmatrix}$			$\begin{pmatrix} 4043 & 1675 \\ 126 & 2655 \end{pmatrix}$			$\begin{pmatrix} 2664 & 150 \\ 22 & 1359 \end{pmatrix}$		

Table 3: Results of classification using SVM.

5.3.2 kNN

The results in this part is obtained with a kNN classifier with $k = 5$.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.95	0.41	0.57	0.96	0.65	0.77	0.99	0.92	0.95
Not spray paint	0.47	0.96	0.63	0.57	0.95	0.71	0.85	0.99	0.92
False positive rate	0.04			0.05			0.01		
Total accuracy	0.60			0.75			0.94		
Confusion matrix	$\begin{pmatrix} 4438 & 6504 \\ 257 & 5783 \end{pmatrix}$			$\begin{pmatrix} 3699 & 2019 \\ 139 & 2642 \end{pmatrix}$			$\begin{pmatrix} 2583 & 231 \\ 20 & 1361 \end{pmatrix}$		

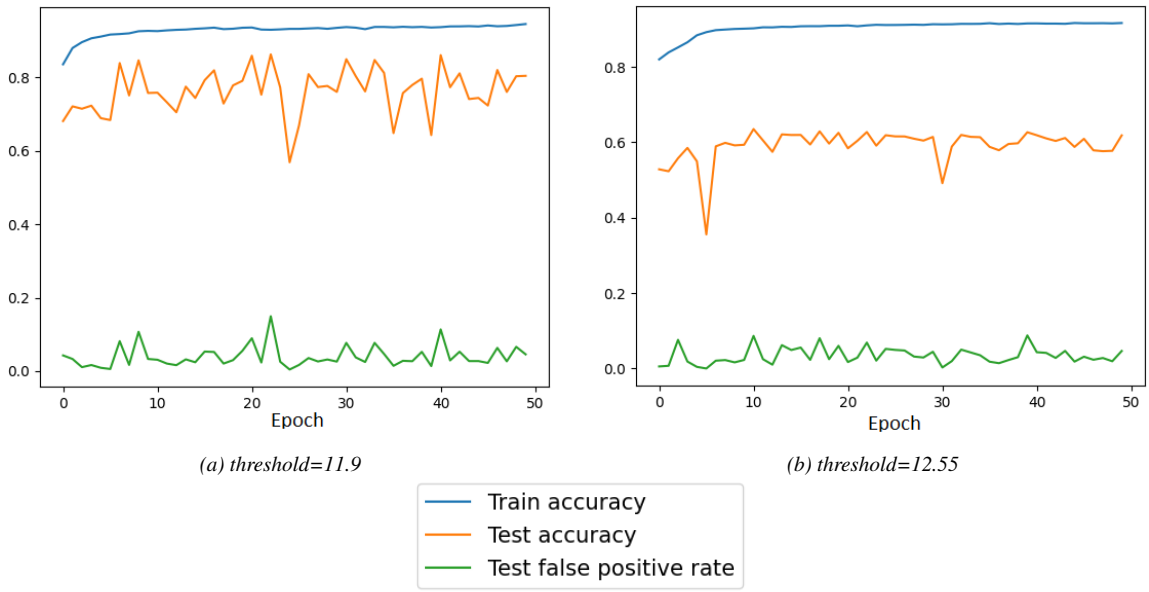
Table 4: Results of classification using kNN

5.3.3 Neural Network

The results in this part is obtained with a neural network trained by minimizing the loss function binary cross entropy with stochastic gradient descent with a learning rate of 0.3 The architecture consisted of 4 hidden layers with 48-36-24-12 nodes respectively and three dropout layers in-between them with a dropout of 0.5-0.4-0.3 respectively.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.94	0.43	0.59	0.97	0.73	0.83	0.99	0.96	0.98
Not spray paint	0.48	0.95	0.64	0.63	0.95	0.76	0.92	0.99	0.95
False positive rate	0.05			0.05			0.01		
Total accuracy	0.62			0.80			0.97		
Confusion matrix	$\begin{pmatrix} 4742 & 6200 \\ 281 & 5759 \end{pmatrix}$			$\begin{pmatrix} 4186 & 1532 \\ 127 & 2654 \end{pmatrix}$			$\begin{pmatrix} 2693 & 121 \\ 17 & 1364 \end{pmatrix}$		

Table 5: Results of classification using a neural network



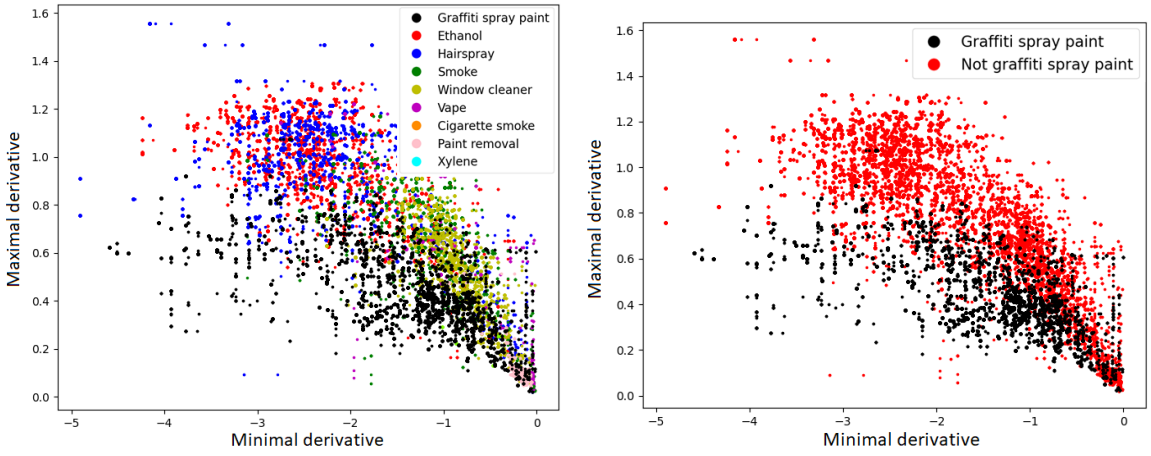
(a) threshold=11.9

(b) threshold=12.55

Figure 28: Training accuracies, test accuracies and test false positive rates for the neural network trained for two different thresholds.

5.4 Classification with derivatives

The classifications in this part are done on 6-dimensional points. The first four dimensions consists of measured resistances at four temperatures, 150°C, 250°C, 350 °C and 450°C. The last two dimensions are the minimal and the maximal derivative for each point. The derivatives plotted against each other is shown in Figure 29 and the result of classification using an SVM is shown in Table 6. Then class weighting has been used in Table 7 as a way to decrease false positives.



(a) *(b)*

Figure 29: Maximal derivatives plotted against minimal derivatives. (a) has assigned colours according to which substance is measured. (b) is coloured according to the binary classifier.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.89	0.84	0.87	0.92	0.97	0.95	0.97	1.0	0.98
Not spray paint	0.74	0.81	0.77	0.94	0.83	0.88	1.0	0.93	0.96
False positive rate	0.19			0.17			0.07		
Total accuracy	0.83			0.93			0.98		
Confusion matrix	$\begin{pmatrix} 9237 & 1705 \\ 1141 & 4899 \end{pmatrix}$			$\begin{pmatrix} 5572 & 146 \\ 460 & 2321 \end{pmatrix}$			$\begin{pmatrix} 2811 & 3 \\ 95 & 1286 \end{pmatrix}$		

Table 6: Results of classification when including derivatives using an SVM.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.95	0.62	0.75	0.98	0.88	0.92	0.98	0.99	0.98
Not spray paint	0.57	0.94	0.71	0.79	0.96	0.87	0.97	0.96	0.97
False positive rate	0.06			0.04			0.04		
Total accuracy	0.73			0.90			0.98		
Confusion matrix	$\begin{pmatrix} 6731 & 4211 \\ 346 & 5694 \end{pmatrix}$			$\begin{pmatrix} 5025 & 693 \\ 123 & 2658 \end{pmatrix}$			$\begin{pmatrix} 2778 & 36 \\ 53 & 1328 \end{pmatrix}$		

Table 7: Results of classification when including derivatives using an SVM and using class weighting to suppress false positives. The 'not spray paint' class was weighted 5 times more for the threshold of 12.55 and 50 times more for the threshold of 11.9 and 11.0.

5.5 Impact of different substances and inclusion of acetone

In this part acetone is included in the dataset which is a problematic substance because it is a large part of what the sensor reacts with in the graffiti spray paint. Figure 30 shows how a temperature sensitivity plot looks like when acetone is included in the dataset. Besides the acetone this is the same data points shown in Figure26b.

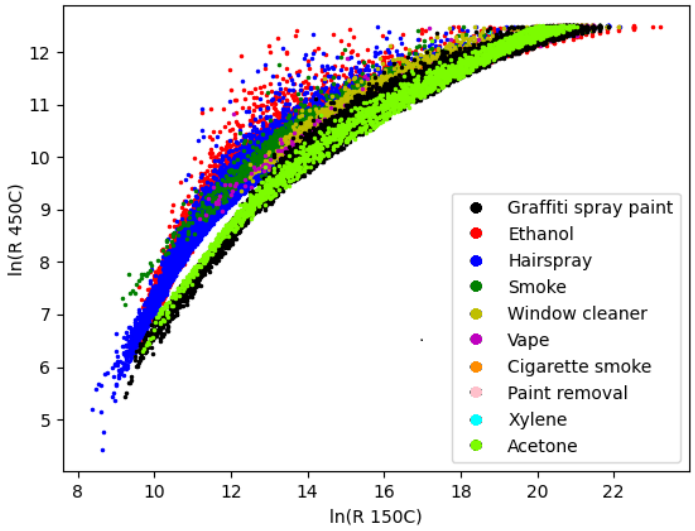


Figure 30: Log-resistance in 450°C plotted against log-resistance in 150°C when acetone is included in the dataset. It behaves very similar to graffiti spray paint.

When acetone is included in the dataset, the impact of different substances for the classification of 4-dimensional points with an SVM is shown in Table 8. The results of this classification is shown in Table 9. Note that the amount of false positives for different substances is essentially unchanged when including or not including acetone in the dataset. Afterwards derivatives are included and the impact of different substances is shown in Table 10 the result of the classification done with an SVM is shown in Table 11.

Substance	Number of points	False positive rate
Ethanol	1237	0.06
Hairspray	2026	0.01
Window cleaner	2243	0
Cigarette smoke	402	0.05
Xylene	132	0.06
Acetone	1239	0.29

(a) threshold=12.55

Substance	Number of points	False positive rate
Ethanol	823	0.10
Hairspray	1087	0.02
Window cleaner	694	0
Cigarette smoke	136	0.19
Xylene	41	0
Acetone	866	0.36

(b) threshold=11.9

Substance	Number of points	False positive rate
Ethanol	536	0.03
Hairspray	658	0.01
Window cleaner	179	0
Cigarette smoke	7	0.14
Xylene	1	0
Acetone	546	0.33

(c) threshold=11

Table 8: False positives rates for different substances using an SVM on 4-dimensional resistance points.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.88	0.32	0.47	0.88	0.58	0.70	0.91	0.73	0.81
Not spray paint	0.48	0.93	0.63	0.57	0.88	0.70	0.69	0.90	0.78
False positive rate	0.07			0.12			0.10		
Total accuracy	0.56			0.70			0.80		
Confusion matrix	$\begin{pmatrix} 3489 & 7453 \\ 486 & 6793 \end{pmatrix}$			$\begin{pmatrix} 3333 & 2385 \\ 434 & 3213 \end{pmatrix}$			$\begin{pmatrix} 2050 & 764 \\ 202 & 1725 \end{pmatrix}$		

Table 9: Results of classification when including acetone in the dataset using an SVM on 4-dimensional resistance points.

Substance	Number of points	False positive rate
Ethanol	1237	0.11
Hairspray	2026	0.10
Window cleaner	2243	0.16
Cigarette smoke	402	0.34
Xylene	132	0.70
Acetone	1239	0.45

(a) threshold=12.55

Substance	Number of points	False positive rate
Ethanol	823	0.17
Hairspray	1087	0.07
Window cleaner	694	0.11
Cigarette smoke	136	0.48
Xylene	41	0.46
Acetone	866	0.57

(b) threshold=11.9

Substance	Number of points	False positive rate
Ethanol	536	0.10
Hairspray	658	0.03
Window cleaner	179	0.4
Cigarette smoke	7	0.43
Xylene	1	0
Acetone	546	0.42

(c) threshold=11

Table 10: False positives rates for different substances using an SVM when including derivatives.

	Threshold=12.55			Threshold=11.9			Threshold=11.0		
	Precision	Recall	F1-score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Spray paint	0.86	0.80	0.83	0.86	0.93	0.89	0.89	0.89	0.89
Not spray paint	0.73	0.80	0.76	0.87	0.76	0.81	0.84	0.84	0.84
False positive rate	0.20			0.24			0.16		
Total accuracy	0.80			0.86			0.87		
Confusion matrix	$\begin{pmatrix} 8747 & 2195 \\ 1472 & 5807 \end{pmatrix}$			$\begin{pmatrix} 5295 & 423 \\ 874 & 2773 \end{pmatrix}$			$\begin{pmatrix} 2509 & 305 \\ 316 & 1611 \end{pmatrix}$		

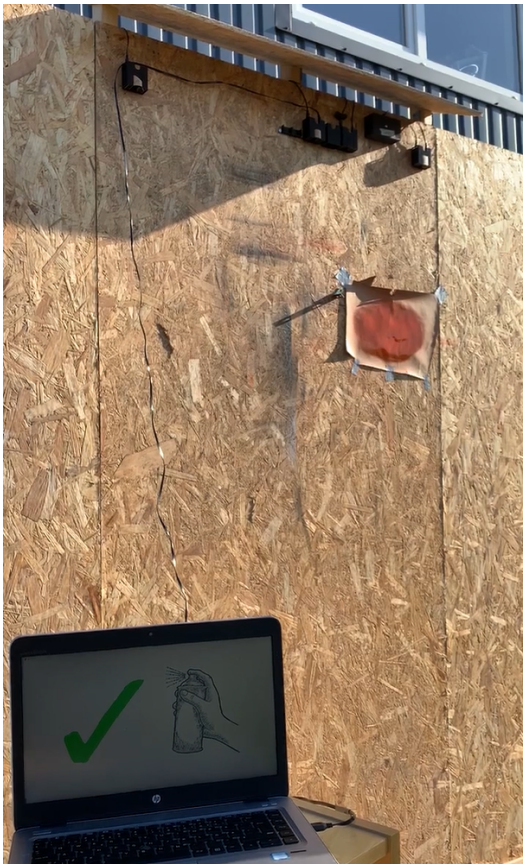
Table 11: Result of classification using an SVM when including acetone in the dataset and including derivatives.

5.6 Real time classification

The classification was performed in real time with graffiti spray paint, ethanol, hairspray and window cleaner as seen in Figure 31. The classification was done with the same SVM trained on points below the threshold of 12.55 used in the previous results. This works as a real time graffiti paint detector by identifying graffiti spray paint in roughly ten seconds after it has been sprayed. It was tested and worked fine in ranges up to two

meters to the side and below the sensor.

(a) Window cleaner classified as 'not spray paint'.



(b) Graffiti spray paint classified as 'spray paint'.

Figure 31: The classification performed in real time by correctly classifying window cleaner and graffiti spray paint. The active sensor is only the one in the center.

6 Discussion

In this section we will further explain the results given in the previous section and discuss our findings. We start by discussing the results of the visualized data followed by the results of the performance of the machine learning classification. Then we discuss the effect of derivatives followed by the impact of acetone. Lastly there is some general discussion with some additional remarks.

6.1 Data visualization and standardization

In Section 5.1 several temperature sensitivity plots are shown made from the data collected over three days. Even though all sensors used are of the same model there is some drift in the data from differences between sensors which can be counteracted to an extent using standardization. Reducing noise concentrates the clusters shown in Figure 22 to tighter lines, such as in Figure 26. The importance of counteracting this noise cannot be overstated as any step taken toward making the data easier to differentiate will be rewarded manyfold in both classification accuracy and sensitivity.

The standardization method used to produce our results was a form of linear shift, as described in Section 4.4.1. This method is quite simplistic but gave good results for our dataset. It is well known that temperature and humidity introduce significant noise to MOx sensors, though exactly how this noise behaves is dependant on the sensors structure [2]. Having low temperature and humidity variance in our dataset was decided to be a reasonable constraint given the scope and goals of this work. It is probable that there are variables beyond what we are aware of which would need to be accounted for in a more comprehensive dataset and we cannot discount the possibility of there being significant non-linearity to the noise introduced from the combination of several sources. A larger dataset with more variance in terms of temperature, humidity and atmosphere would have to be collected in order to account for said variance through more complex standardization.

Appendix A consists of a selection of unprocessed resistance curves. It can be seen that there is a substantial variation in the appearance of these curves. This is one of the reasons that we classified each point individually to only utilize the sensors sensitivity to different gases at different temperatures and not worry about this variance.

6.2 Classification methods

When classifying 4-dimensional points individually as in Section 5.3 all machine learning methods perform very well for the medium and lowest thresholds. When including the upper threshold of 12.55 they all struggle which is to be expected. Looking at the visualized data in Section 5.1 it can be seen that there is a high density of overlapping points near the baseline where there is a low concentration of gas reacting with the sensor. For classification purposes it is not necessarily important to use points that are this close to the baseline when the performance at lower thresholds is significantly better. All performance metrics at the lowest threshold of 11.0 are almost perfect on our dataset for all methods, as seen in Figure 27. It is evident from the Visualization section that the data is trivially separated for lower thresholds. It is the higher thresholds that are of interest when evaluating the machine learning classifiers. Using a higher threshold means that the sensor can differentiate gases at greater distances and of lower concentrations. A threshold of around 11.9 seems reasonable to use in practice.

The first machine learning classifier that is being evaluated is the SVM in Table 3. The results look promising with a total accuracy of 0.79 and a false positive rate of 0.05 for the medium threshold. For the higher

threshold the performance is subpar with a fairly low total accuracy and F1-score. The false positive rate of 0.02 is very low which is extremely good. Overall the SVM does a fine job of the classification while being very safe in producing false positives which is also evident from the lower recall of the 'spray paint' class.

The kNN classifier produces very similar results as the SVM, as seen in Table 4. The most notable difference is the slightly lower performance for the medium threshold with slightly lower accuracy and F1-score because of the larger amount of false negatives. For such a simple method the kNN classifier performs remarkably well. One downside of using the kNN in this case is the imbalanced form of the dataset. In the training data there is an abundance of 'not spray paint' points and in the test data there is instead an abundance of 'spray paint' points. The dataset is not imbalanced to an extreme degree and the kNN still performs fairly well but is outclassed by both the SVM and the neural network.

Lastly the neural network in Table 5 outclasses both the SVM and the kNN classifiers. Overall the neural network reaches a higher performance than both the SVM and the kNN classifiers. Total accuracies and F1-scores are better for the neural network for all three thresholds. The neural network performs especially well on the medium threshold. However, there is a tradeoff between accuracy and false positive rate. The increased accuracies and F1-scores also lead to a slight increase in false positive rate. There is a correlation in test accuracy and test false positive rate. This can also be seen in Figure 28b where the test accuracy and the training accuracy is shown during training. Figure 28b also looks like a typical case of overfitting where the training accuracy is much higher than the test accuracy. This is also an artefact of having all training data from the same day while the test data is from two other days.

When the threshold is set at 12.55 the points are very hard to differentiate and it would seem that this kind of classification is not enough to produce satisfying results in this case. This is definitely not a bad result, points this close to the baseline can not be expected to be differentiated. This can be seen in the data visualization where there is a total overlap for these points, for example Figure 25. It is reasonable to only use the points for the lower thresholds where there is better differentiation. It is important to remember that it is not necessary to have perfect accuracy. In practice no conclusions can be drawn from the classification of a singular point. It is the combined classifications of multiple points that are of importance. A single spray of graffiti spray paint will result in roughly 30-80 points and it is the majority prediction of all these points that really matters.

6.3 Effects of derivatives

There is definitely a distinct pattern in the derivatives as seen in Figure 29 where the minimal derivative is plotted against the maximal derivative. The different substances are somewhat divided into overlapping clusters where ethanol and hairspray, two substances which the sensor is very sensitive to, tend to have large derivatives. Most points of graffiti spray paint have a maximal derivative less than 0.8 and a minimal derivative greater than -2.5.

The derivatives are more noisy compared to the plain 4-dimensional resistance points which are almost perfectly separated at low thresholds for our dataset. This added noise is reflected in the increased quantity of false positives when including derivatives as seen in Table 6. The false positive rate is increased from 0.02, 0.05 and 0.02 to 0.07, 0.17 and 0.19 for the low, medium and high threshold respectively. Even though the precision for 'spray paint' is decreased, including the derivatives increases the F1-score and the total accuracy. The biggest increase is for the high threshold where the total accuracy is increased from 0.59 to 0.83 when including derivatives. This is because the points close to baseline are almost impossible to differentiate using only the resistance values at four temperatures. The derivatives give these points properties of the surrounding points.

In order to decrease the false positives that arise from the added noise of the derivatives a class weighting

was used. The points of the 'not spray paint' class were weighted higher, making them more important for the classifier to classify correctly and produce less false positives. This worked to a degree as seen in Table 7 where the false positives were significantly decreased at the cost of the total accuracy and F1-score.

It is not clear to what extent the derivatives are useful. On the one hand they do increase most aspects of classification performance but on the other hand they also introduce noise and false positives, even though that can be counteracted by manipulating the data with class weighting. The noise would not have as large impact if there were more features at our disposal. Normally when classifying higher dimensional data there are much overlap and noise between the features. This is not as big of a problem if the machine learning classifiers can find higher dimensional patterns from all features collectively. In the case of our low dimensional data the noise has a greater impact. Another thing worth considering is that when only resistance values at different temperatures is used the differentiation is based on the sensors sensitivity to different gases at different temperatures. This is based on intrinsic properties of individual gases. The derivatives are on the contrary dependant on the gas concentration. This is necessarily not a problem because from our measurements of spray paint all derivatives lie in the same range so they are still useful. However, the increased false positive rate can be a problematic because the purpose of the classification is making a detector where false positives are a big problem.

6.4 Impact of different substances and inclusion of acetone

There is a large discrepancy in differentiating different substances, as seen in Table 8. When derivatives are not included ethanol, hairspray, window cleaner, and xylene have very low false positive rates. Cigarette smoke has a false positive rate of 0.19 for the medium threshold which is slightly larger but not unacceptable. However acetone is a problem. This is expected because acetone is the largest component of the graffiti spray paint and a large part of what gives graffiti spray paint its measured properties. First off the false positive rate of acetone is greater than 0.29 which is problematic. Secondly, including acetone in the training data set impacts the classification negatively compared to not including it as seen in Table 9 where there is an increase in false negatives in addition to the false positives which are mostly a result of the acetone. A large part of the graffiti points that would otherwise have been correctly classified will now be classified as 'not spray paint'

The inclusion of derivatives is not a good solution to this problem. As seen in Table 11 every performance measure is worse compared to not including acetone. Also the false positive rates are significantly greater compared to when derivatives were not included. This is also seen in Table 10 where the false positive rate is unacceptably high for window cleaner, xylene, cigarette smoke and especially acetone. The introduced noise from derivatives are detrimental on the classification when acetone is included in the dataset. The false positive rate could be reduced by class weighting at the cost of total accuracy, if deemed a problem.

6.5 Additional remarks

The classification worked as expected when implemented as a real time detector. There are many parameters to vary for this kind of classification. The threshold to start classifying points can be increased to make the sensor more sensitive and work at greater ranges and lower concentrations at the cost of including points that are harder to differentiate. The amount of points to take into consideration can also be adjusted for quicker response times at the cost of less certainty of the predicted class. Also the majority percentage that is needed to sound an alarm can also be changed to adjust this confidence.

The results are highly dependent on the choice of sensor. The material and characteristics of MOx sensors have a great effect on the sensitivity to different gases. Different sensors have been tested during the work of

this thesis and only one of them resulted in the presented results. This is a well known phenomenon in the industry and sensors are often chosen based on the specific substance one is interested in. This means that the results obtained in this thesis may be hard to reconstruct with other types of MOx sensors.

7 Conclusions and further research

7.1 Summary

Using a variable temperature MOx sensor worked very well for the differentiation of graffiti spray paint from other substances. The classification of individual data points consisting of resistance values measured at 150°C, 250°C, 350°C and 450°C proved to be an effective way to utilize the sensors sensitivity to each gas at different temperatures. It was determined that acetone is a large part of what the sensor is reacting with in the graffiti spray paint as it behaves very similar to the temperature sensitivity characteristics of graffiti spray paint.

The SVM, kNN and neural network all performed very well when there was a significant amount of gas reacting with the sensor. The neural network performed slightly better than the SVM and kNN on most performance metrics. Closer to the baseline, where there is less gas reacting with the sensor, the points got much harder to differentiate. The inclusion of derivatives helped in the differentiation of these points by giving them properties of the surrounding points, at the cost of added noise and an increase of false positives.

7.2 Improvements and further research

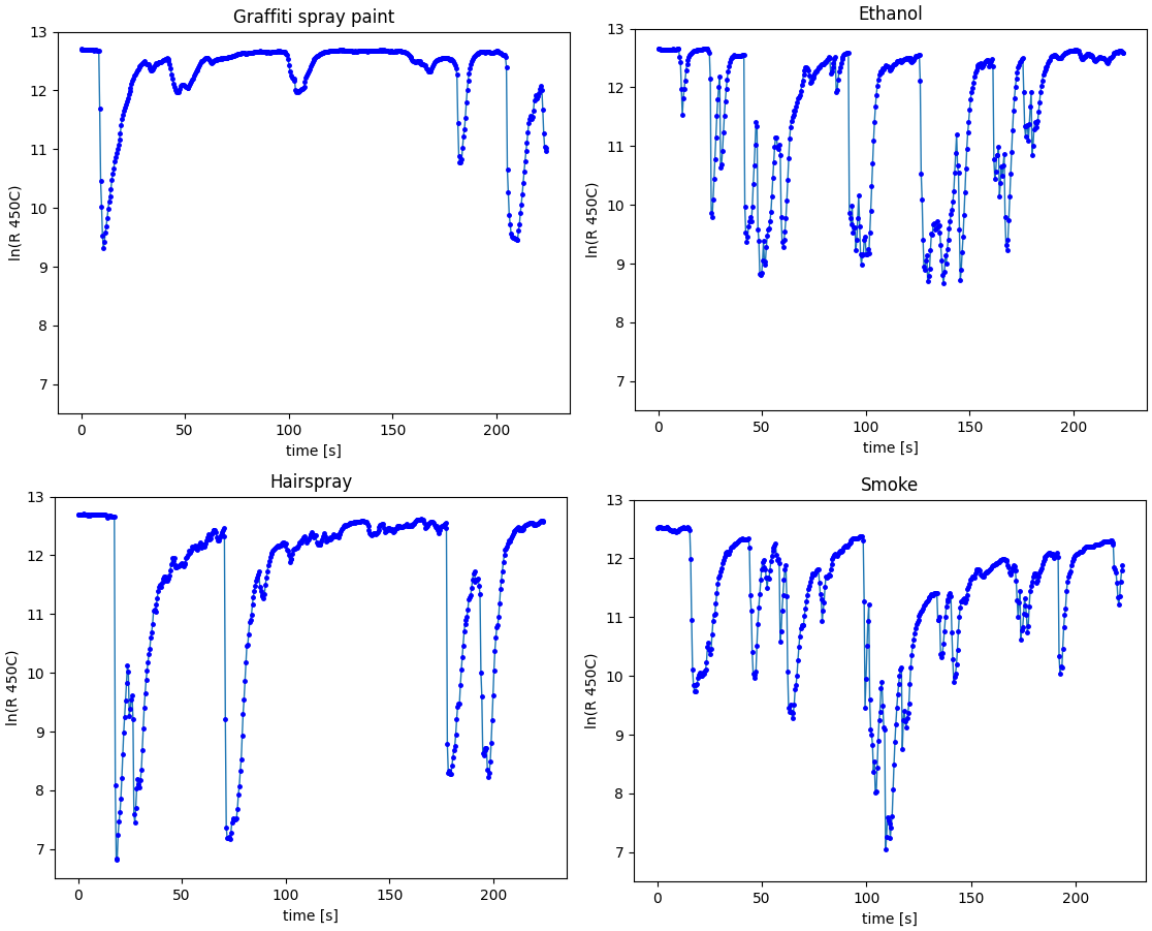
The part of our work with most apparent potential for improvement is standardization. Linear shift, which we used is quite basic and developing more comprehensive preprocessing should be a priority for any further development, both to be able to use the sensors in varying environments and in particular to have them work indiscriminate of weather conditions. Standardization can be incredibly complicated and much more work could be done here. As mentioned in Section 6.1 any improvements done in this area will improve accuracy and sensitivity both while also making training easier. Perhaps machine learning could be used to optimize standardizing the data.

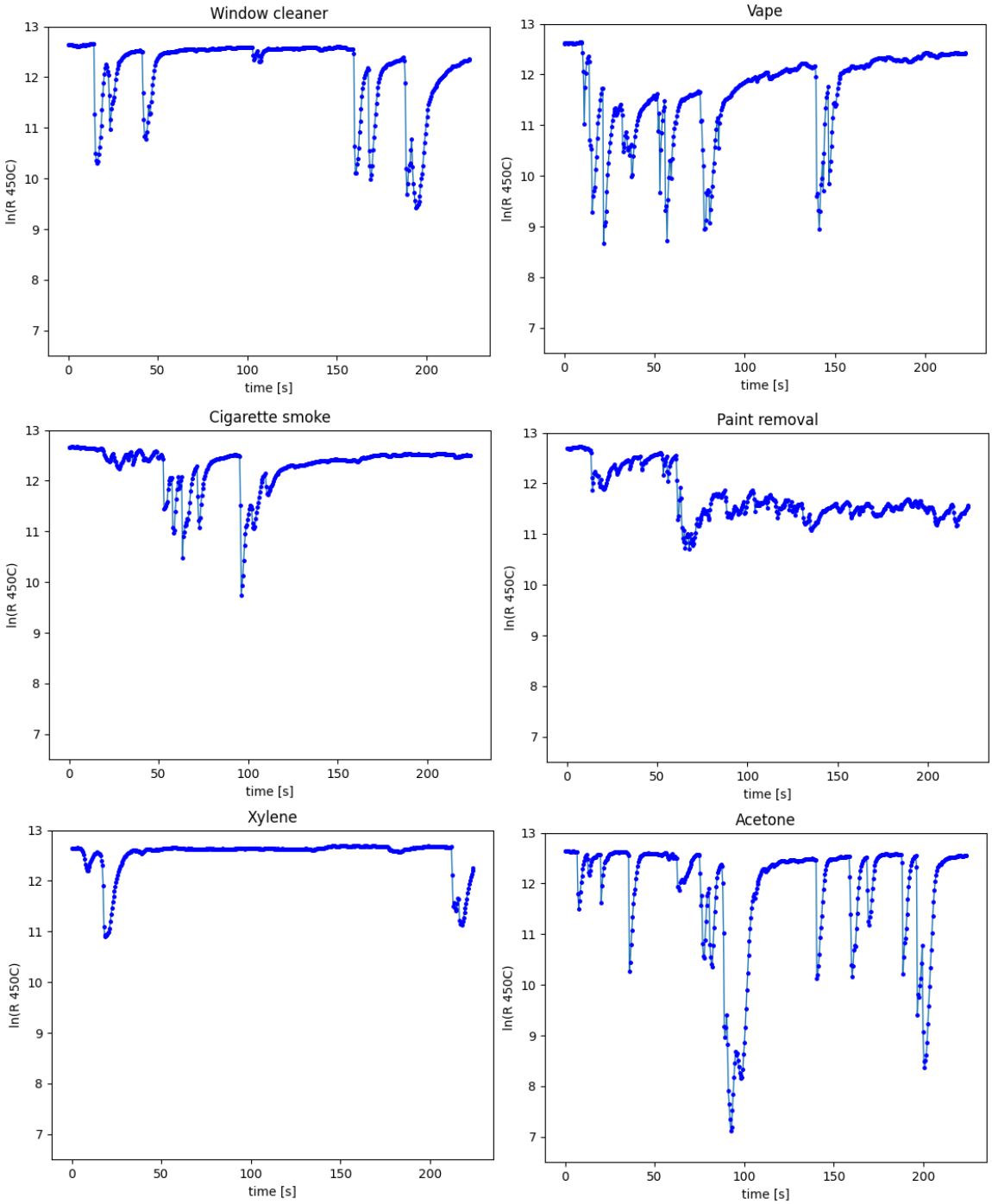
The sensor used in this thesis is most sensitive to acetone, which is a component of most if not all spray paint. Because of this sensitivity, other sources of acetone fumes might trigger false positives from the sensor. From our experiments, it is unlikely that this problem could be overcome through further developing our data analysis from the one sensor. Adding another sensor which is sensitive to some other component of spray paint, such as the propellant dimethyl ether, could be a possible solution to avoid these false positives. Using an additional sensor with a different sensitivity could also be used to improve the results in this thesis as well as help with unforeseen problems outside of what has been accounted for in this thesis.

Every one of the machine learning methods used in this thesis do pointwise classification. Adding time dependence could significantly improve results if done well. A good way to do this is using recurrent neural networks. Feature extraction using machine learning might also be good. Convolutional neural networks are especially well suited to this task. It is worth investigating if using recurrent and convolutional networks together improve results.

A Appendix - Resistance curves

In the following figures a selection of the directly measured resistance curves are shown. Note that only a small subset of the measured points is shown. For each substance 500 points are shown corresponding to 227 seconds. All the data was collected from the sensor placed in the center of the wall, marked B in Figure 17.




Window cleaner
Vape
Cigarette smoke
Paint removal
Xylene
Acetone
ln(R 450C)
time [s]

References

[1] Scikit-learn precision-recall. https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html.

[2] A. Abdullah, K. Kamarudin, S. M. M. Syed Zakaria, A. Adom, and Z. Juffry. Effect of environmental temperature and humidity on different metal oxide gas sensors at various gas concentration levels. *IOP Conference Series: Materials Science and Engineering*, 864:012152, 07 2020.

[3] H. Araki, M. Oura, and S. Omatu. An evaluation of an olfactory sensor and recognition system using metal-oxide semiconductor gas sensor with variable heater controller. *Artificial Life Robotics*, 23(1):73, 2018.

[4] M. Balaban, F. Korel, A. Odabasi, and G. Folkes. Transportability of data between electronic noses: mathematical methods. *Sensors and Actuators B: Chemical*, 71(3):203 – 211, 2000.

[5] N. Bârsan and U. Weimar. Fundamentals of metal oxide gas sensors. *14th International Meeting on Chemical Sensors*, pages 618–621, 2012.

[6] A. Dey. Semiconductor metal oxide gas sensors: A review. *Materials Science and Engineering: B*, 229:206–217, 2018.

[7] S. Di Carlo and M. Falasconi. *Drift Correction Methods for Gas Chemical Sensors in Artificial Olfaction Systems: Techniques and Challenges*. 01 2012.

[8] B. Firtat, C. Moldovan, C. Brasoveanu, G. Muscalu, M. Gartner, M. Zaharescu, P. Chesler, C. Hornoiu, S. Mihaiu, C. Vladut, I. Dascalu, V. Georgescu, and I. Stan. Miniaturised mox based sensors for pollutant and explosive gases detection. *Sensors and Actuators B: Chemical*, 249:647–655, 2017.

[9] J. Fonollosa, L. Fernández, A. Gutiérrez-Gálvez, R. Huerta, and S. Marco. Calibration transfer and drift counteraction in chemical sensor arrays using direct standardization. *Sensors and Actuators B: Chemical*, 236:1044 – 1053, 2016.

[10] A. Geron. *Hands-on Machine Learning with Scikit-learn, Keras and TensorFlow*, page 155. 2019.

[11] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[12] W. Guan, N. Tang, K. He, X. Hu, M. Li, and K. Li. Gas-sensing performances of metal oxide nanostructures for detecting dissolved gases: A mini review. *Frontiers in Chemistry*, 8:76, 2020.

[13] Y. Jia, G. Xiuzhen, D. Shukai, J. Pengfei, W. Lidan, P. Chao, and Z. Songlin. Electronic nose feature extraction methods: A review. *Sensors (14248220)*, 15(11):27804 – 27831, 2015.

[14] Y. Luo, W. Ye, X. Zhao, X. Pan, and Y. Cao. Classification of data from electronic nose using gradient tree boosting algorithm. *Sensors (14248220)*, 17(10):2376, 2017.

[15] J. Majlard. Klotternotan ökar - nära 300 000 kronor per dag. *Svenska Dagbladet*, 2018.

[16] M. Mizanur Rahman, C. Charoenlarpnopparut, and P. Suksompong. Classification and pattern recognition algorithms applied to e-nose. In *2015 2nd International Conference on Electrical Information and Communication Technologies (EICT)*, pages 44–48, 2015.

[17] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall. Activation functions: Comparison of trends in practice and research for deep learning, 2018.

[18] S. Omatu. Odor classification by neural networks. In *2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*, volume 01, pages 309–314, 2013.

[19] X. Pan, H. Zhang, W. Ye, A. Bermak, and X. Zhao. A fast and robust gas recognition algorithm based on hybrid convolutional and recurrent neural network. *IEEE Access*, 7:100954–100963, 2019.

[20] P. Ramachandran, B. Zoph, and Q. V. Le. Searching for activation functions, 2017.

[21] P. Sanmartín, F. Cappitelli, and R. Mitchell. Current methods of graffiti removal: A review. *Construction and Building Materials*, 71:363–374, 2014.

[22] P. N. Stuart Russel. *Artificial Intelligence A Modern Approach, Third Edition*.

[23] P. Wang, D. Wang, M. Zhang, Y. Zhu, Y. Xu, X. Ma, and X. Wang. Zno nanosheets/graphene oxide nanocomposites for highly effective acetone vapor detection. *Sensors and Actuators B: Chemical*, 230:477–484, 2016.

[24] B. Zhang, X. Zhou, C. Jiang, F. Qu, and M. Yang. Facile synthesis of mesoporous co3o4 nanofans as gas sensing materials for selective detection of xylene vapor. *Materials Letters*, 218:127–130, 2018.

[25] S. Zhang, C. Xie, D. Zeng, H. Li, Z. Bai, and S. Cai. A method of feature extraction from the desorption part of mox's response curves to gases. *IEEE Sensors Journal*, 8(11):1816–1823, 2008.

[26] S. Zhi-biao, Y. Tao, Z. Qun, L. Yang, and L. Yu-bin. Comparison of algorithms for an electronic nose in identifying liquors. *Journal of Bionic Engineering*, 5(3):253 – 257, 2008.

Master's Theses in Mathematical Sciences 2021:E14
ISSN 1404-6342

LUTFMS-3412-2021

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/