Examensarbete

HÖGSKOLAN I SKÖVDE

A COMPARISON OF LIGHTGBM AND PERCEPTRON FOR CLASSIFYING THE CAUSE OF SALARY DIFFERENCES BETWEEN WORKGROUPS

Comparative study for classifying the reason for salary difference with different machine learning algorithms

Bachelor Degree Project in Information Technology
Basic level 30 ECTS
Spring 2021

Ibrahim Tekin

Handledare: Niclas Stål
Examinator: Juhee Bae

Summary

Machine learning is part of what is called AI. It is defined as the application of an algorithm to improve a result through learning.

In Sweden, the law requires large companies and organizations to revise their salaries every year to ensure there is no wage disparity between men and women. This could be used as an assisting tool if machine learning is applied to the analysis process.

By training two different models and test them against the same test dataset different metrics can be obtained and analyzed to see how they perform in comparison to each other. The results show a slightly improved performance by the perceptron and that there is room for further development.

This study is limited to a smaller dataset for training and testing. But in the future, more relevant features and larger datasets could be added for training the models and lead to a more accurate model.

Machine Learning

How to train on a task and evolve.

ML.Net

ML.Net is Microsoft's machine learning framework.

ANN

Artificial Neural Networks can be applied to machine learning. It resembles the human neural system.

Classification algorithms

Understanding and recognizing objects into a preset of categories.

Table of Contents

1 Introduction

Machine learning can be traced back to 1943. In 1958 the first artificial neural network algorithm was created by Frank Rosenblatt and was called perceptron to recognize patterns and shapes (Lefkowitz, 2019). Machine learning is a subpart of Artificial Intelligence and can be defined as the use of algorithms that improves the result through experience.

ML.Net is Microsoft's framework for working with machine learning. This enables the usage of machine learning within the .net sphere. Several different algorithms are supported within this framework. For this study, the two selected algorithms are LightGBM and the Perceptron.

LightGBM is an algorithm developed by Microsoft. The improved optimization could seem to be a promising choice of algorithm (Ke, Meng, Finley, 2017). The Averaged Perceptron is a further development of the original Perceptron algorithm based on an Artificial Neural Network structure. Some optimizations made to LightGBM are explained in this study. LightGBM has EFB (Exclusive Feature Binding) this reduces the features and the training time of GDBT and GOSS (Gradient One Side Sampling) that excludes small gradients and uses the rest to estimate the information gain.

In the Nordic countries today the appreciated wage difference between gender is estimated to be around 14% (Måwe I, 2019). The law in Sweden makes it mandatory for larger businesses and organizations to yearly review and document the results of their salary survey. This study aims to compare the two classification algorithms on predicting the reason between a female-dominated workgroup compared to an equal or equivalent male-dominated workgroup.

Two machine learning models are created using .net and ml.net. Two separate datasets are used. One dataset is used when training both models. The second dataset is used to test the models to obtain two different results. One validity threat to take into consideration is to separate the training and testing dataset to avoid overfitting. The obtained dataset is heavily imbalanced and is handled as such in this study. Unexplained predictions are classified as "NA / Others" and are seen as the default value because most of the samples come from this class. The obtained results are analyzed, and the metrics serve as the foundation for the study's conclusion.

Machine learning is interesting and applicable in this case because the idea is that the system learns and develops over time, allowing it to predict better. Applying conventional programming in the same respect is limited. Analyzing the cause of inequality between working groups is complex, even for consultants with experience.

Contribution

The results of this study show that there is a possibility to apply machine learning to analyze causes between different workgroups. For this study, the comparison is not done directly to male and female salaries but rather comparing female-dominated workgroups to equal or equivalent male-dominated workgroups. To the best of my knowledge, there is no machine learning used in the same sense as this study. With the results presented and with further research on creating a more general dataset, the application of this study can be applied in a broader aspect. The dataset used is heavily unbalanced and must be taken into account. All datasets used are collected from real surveys. When the trained model cannot find evidence for prediction it goes for the majority class which can be seen as the default value. The predictions made on the majority class “Na / Other” need further investigation. Given the imbalance of datasets, the most appropriate application is to be used as a support tool for analyzing the cause of inequality.

2 Background

Machine learning is part of what is called artificial intelligence. Machine learning is the use of algorithms that improves the result through experience. According to (Ståhl, Duarte, 2019) a learned program can propose new outputs given new inputs. This method of science is used in a variety of fields of everyday usage in the real world. It can be applied to image recognition, credit card fraud, or email filtering. Daily use of machine learning is constantly applied around us.

We can find the first case of a neural network in 1943. Neurophysiologist Warren McCulloch and mathematician Walter Pitts created the first neural network model using an electrical circuit. The world-famous Turing test was created in 1950 by Alan Turing. In 1952 Arthur Samuel created a checker software that learns as it ran. The first artificial neural network was designed by Frank Rosenblatt in 1958 called Perceptron to recognize patterns and shapes (Marr, B. 2020).

In the 1980s and 1990s, the interest in neural networks started to pick up again. Japan announced it would focus on more advanced neural networks and John Hopfield suggested creating a network similar to how neurons work. IBM computer Deep Blue managed in 1997 to beat the world chess champion (Marr, B. 2020).

Not until decades later have this field of data science grown as abundantly as in recent times. In the 21st century, many huge corporations research more heavily within this field. Some of these projects include ResNet, GoogleBrain, AlexNet, U-net, DeepFace, and many more (Marr, B. 2020).

Machine learning aims to allow computers to learn by themselves without the intervention of human assistance. The learning process starts with the observation of data to recognize patterns in data for improved decision-making of future usage.

The key point in machine learning is the entry of data to train the algorithms to find patterns and be able to make future predictions based on new data. For machine learning using the quality and quantity of the data can enable faster and more accurate results.

Machine learning is the data science that can be applied in several different fields and day-to-day usage. From finance, medicine, fraud detection to banking, and much more. The right kind of machine learning applied in a business could lead to advantages against competitors. In retail, it can be used to forecast demands, optimize prices and provide customer recommendations.

Smartphones loaded with Apple Siri, Amazon Alexa, or Google. In medical image analysis within the medical science field, the volume of imaging data has increased and potentially leading to more opportunities for human error in analyzing the data.

In recent years the ambition in the car manufacturing world to produce self-driving vehicles is very high. This is a very advanced and demanding task. To guide a car would demand continuously identifying objects in the environment and the usage of many forms of machine learning is required.

2.1 Machine learning methods

Within the field of machine learning, we have some different types of approaches for training the machine learning model. To better comprehend machine learning and the experiment of this study it is important to understand these methods. In this study, the dataset used contains both the input value and the output value making it a supervised learning method. Also, the value predicted by the trained model is a class value given some input.

2.1.1 Supervised learning

Supervised learning contains a set of data containing both the input and the output result. The machine learning model is trained on a set of labeled data and the labels can be used as the information that is to be determined. For example, if a machine learning model is tested to determine the breed of a dog, then all the pictures of dogs would be labeled with the dog breed.

This learning method requires less training data for the algorithm and makes it easier to predict a result because of the known actual labeled data. Labeled data can be more expensive to prepare and have a greater possibility of overfitting. With the help of the labels building the predicting model is easier for the learners (Ståhl, Duarte, 2019).

Classification

The difference between classification and regression problems is that the labels are discrete values. By discretize the values of the regression dataset, it can be transformed into a classification dataset. By discretize the values it means to group the continuous values into classes (Ståhl, Duarte, 2019).

Classification is to predict a class given some input data. By mapping a function (f) from input variables (x) to discrete output variables (y). It falls within the machine learning category of supervised learning.

In chapter 5.1 Datasets, a deeper explanation of the features and output value is explained. Making it obvious to the reader why supervised learning is used and the output value is of a classification type.

2.1.2 Unsupervised learning

Unsupervised learning requires the algorithms to find a structure in the data like grouping or clustering of data points. Unsupervised learning is more about identifying patterns and structures in data. This is possible because the algorithm extracts feature in the data in real-time to be labeled, sorted, and classified without human interference. This can be applied within the field of spam filter. By analyzing patterns and features in a huge number of emails the algorithm can recognize indications of spam in an email.

2.1.3 Semi-supervised

Semi-supervised means something in between supervised and unsupervised learning. Both labeled and unlabeled data are being used for training (Brown Lee 2019). For training a smaller dataset containing labeled data is used for feature extraction guidance. The trained model can then be applied to a larger unlabeled dataset for feature extraction. This is used in cases where different circumstances do not allow for a complete dataset containing labeled data.

2.1.4 Reinforcement

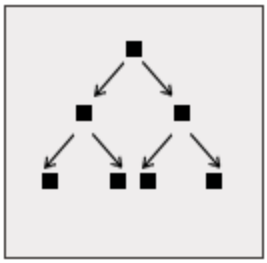
Reinforcement algorithms learn by interacting through actions and discovers errors and rewards. The most characteristic reinforcement method is trial and error and delayed reward. This learning algorithm is often applied to robotics, navigation, or gaming. It contains three primary components. The first component is the agent that is the decision-maker. The second component is the environment, which is everything that interacts with the agent. The third component is the actions, all the events done by the agent. Reinforcement learning aims to learn the best policy. This is done by the agent choosing the best action and result over a given amount of time that maximizes the expected reward. The agent can finish much faster by following a good policy.

2.2 Machine Learning algorithms

There are many different kinds of algorithms used in machine learning science. In this section, they are categorized and explained based on their similarity. Some algorithms of course could easily be grouped elsewhere. In this study, one of the compared algorithms is of the type Artificial neural network and is presented in depth in chapter 2.5 Perceptron. The other algorithm chosen for this study is the LightGBM which is also classified as a Decision tree algorithm and is further explained in chapter 2.5 LightGBM. In this chapter, a short presentation of the different types of algorithms is presented that is used in the experiment of this study.

2.2.1 Decision tree

Is made up of a model that was built by making decisions based on the data's actual values of attributes. Decision Tree algorithms are commonly used in machine learning, they are also often fast and accurate. They are applied for classification and regression problems (Brownlee, 2019). Some popular algorithms:

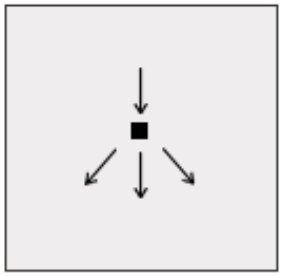


Decision Tree Algoritm
(Brownlee, 2019)

- M5
- Conditional Decision Tree
- Classification and Regression Tree (CART)

2.2.2 Artificial neural networks

Models based on the structure and functionality of biological neural networks. Regression and classification issues are common applications (Brownlee, 2019).



Artificial neural network
(Brownlee, 2019)

Some popular algorithms:

- Perceptron
- Multilayer Perceptron
- Back-propagation

2.2.3 Other kinds of algorithms

Deep learning and Modern Artificial Neural networks exploit abundant cheap computation. Uses multiple layers to extract a higher level of features from raw data. Bayesian algorithms apply Bayes Theorem to handle classification and regression problems. Bayes Theorem

describes the probability of an outcome based on the previous occurring outcome. Regression algorithms, a regression model is based on the relationship between variables. Iteratively, the model's predictions are refined using a measure of the error in the model's predictions (Brownlee, 2019).

2.3 Important ML Concepts

When working within the field of machine learning it is important to understand some basic concepts to avoid erroneous models. The concepts presented in this chapter explain some issues when working with machine learning. Some of these issues are discussed in chapter 3.4.1 validity threats and when analyzing the results in chapter 6. Figure 1 shows the different concepts of machine learning.

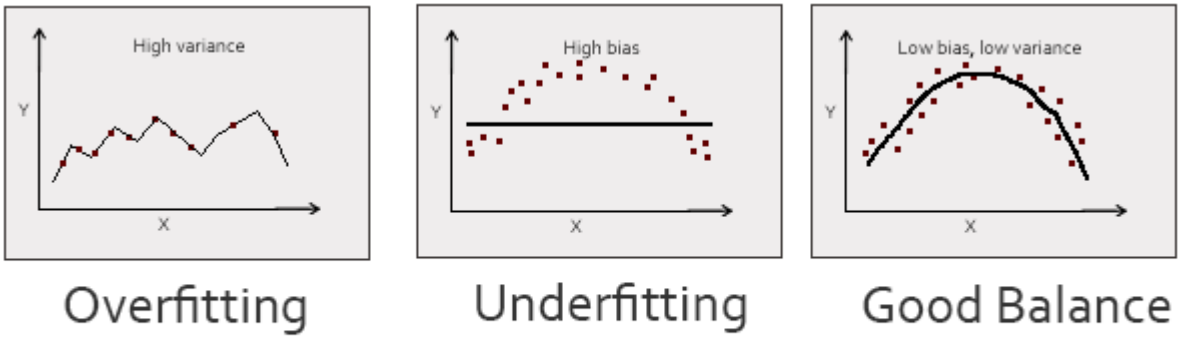


Figure 1 Diagram of over and underfitting (Singh, 2018).

When the model is learning it may suffer some learning problems. Overfitting and underfitting are the two most common ones. Both of these are intimately related to the variance and bias of the model (Ståhl, Duarte, 2019). Bias tells us how much structure of the dataset has not been learned. Variance tells us how much structure from the dataset the model has learned. The inability of the machine learning method to capture the true relationship is called bias.

Overfit occurs when a model's performance on the training set is substantially better than the performance of the test set. The main reason why the dataset is split into two sets, one for testing, and one for training are to have the ability to compare the trained model using the result of the test dataset to assess the model's capability for predictions (Klosterman, 2019).

The arising problem when a model is overfitting its training data set is that it has a high variance. One possible reason for this is variability in the training dataset. So, a model with high variance has likely been learned by a dataset containing noise. Noise meaning that a dataset contains high fluctuations or offsets from the true value. Noise distorts the true relationship between the feature and the response variable.

When a model does not fit the training data well, it is said to underfit. A strong bias is reported to exist in the model. The bias is the difference between our model's average prediction and the predicted value. When the training data is ignored and the model is oversimplified, the result is a high level of inaccuracy on both the training and test data (Sing, 2018). See figure 2 for a bullseye diagram of bias and variance.

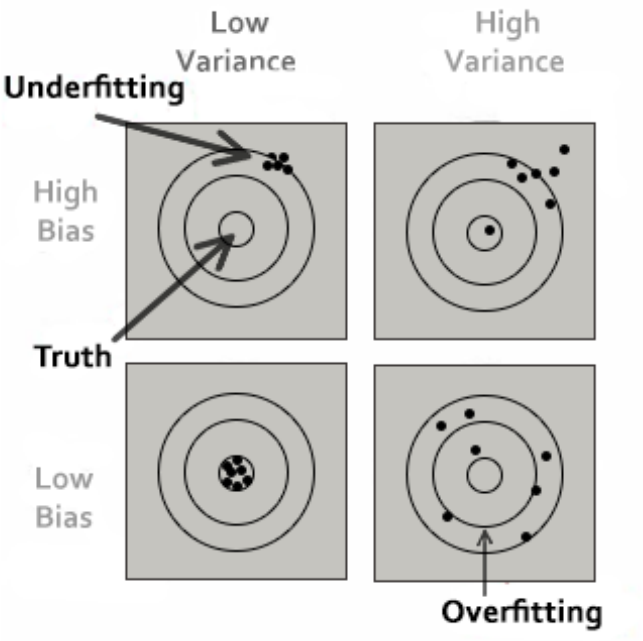


Figure 2 Bullseye diagram of Bias and Variance (Singh, 2018)

2.4 LightGBM

This linear regression algorithm is very popular and utilizes a Gradient-based One Side Sampling (GOSS) to filter out the data instances for finding a split value. It was created by researchers at Microsoft. LightGBM, unlike classic GBDT-based techniques like XGBoost and GBDT, grows the tree vertically, whereas other algorithms build trees horizontally, making LightGBM an excellent way for processing large-scale data and features (Sun, Liu, Sima, 2020). It uses a serial approach to combine decision trees. The approach combines decision trees so that each learner fits the residuals (negative gradients) from the preceding tree, resulting in an improved model. The outcomes of each phase are then combined to create a strong learner. The accuracy of GBDT has propelled it to the top of machine learning competitions. According to LightGBM optimizations, it outperforms some of the other

classical machine learning algorithms in both speed and accuracy. A decision tree splits observations based on feature values and by doing so the decision tree "learns". With an algorithm that is looking for the best split leading to the highest information gain

Entropy is the measurement of uncertainty or randomness. High entropy means a high level of disorder and a low level of purity. Information gain is the difference between entropy before and after the split. The higher the randomness of a variable the higher the entropy value. Find the best splits can be the most time-consuming part of the learning process of a decision tree. Pre-sorted and Histogram-based are two different implementations for split (Yildirim, 2020).

Because the histogram-based technique performed better than the pre-sorted approach, it was chosen as the starting point. To get the most information out of each feature, all data instances are analyzed to identify the best split. The complexity of the histogram-based technique is determined by the number of data instances and features. The GOSS and EFB are two strategies used to solve this problem and are discussed further in chapters 2.4.1 and 2.4.2. Figure 3 presents an overview of the gradient boosted decision tree.

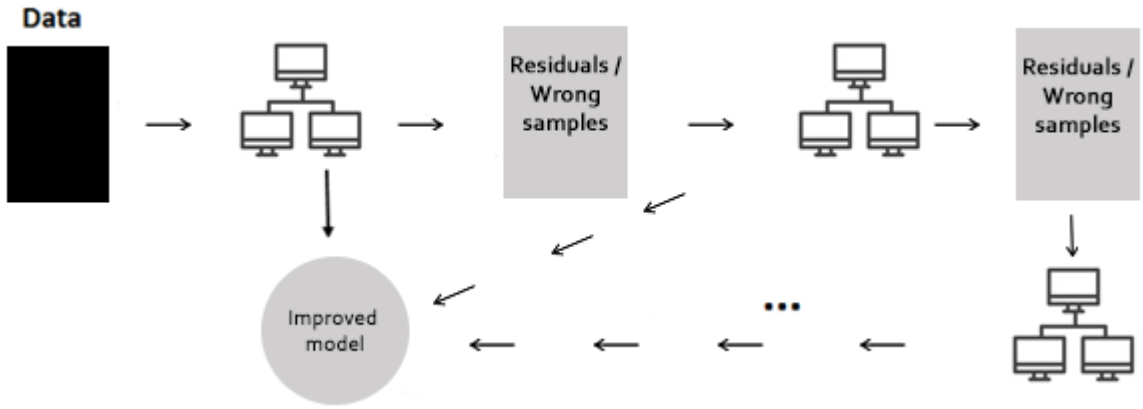


Figure 3 Gradient Boosted Decision Tree (Yildirim, 2020)

2.4.1 GOSS (Gradient One-Side Sampling)

To scan all data instances for best split is not the best or optimal way. If possible, to sample data based on information gains result in a much more effective algorithm. One possible alternative is sampling data based on their weights. This is not possible in GDBT because there is no sample weight (Yildirim, 2020).

By using gradients that allow for valuable insights into the information gain we could address this issue by applying GOSS that excludes small gradients and use the rest to estimate the information gain.

- Small gradient: the instance has been trained by the algorithm the upcoming errors associated with it are small.
- Large gradient: large errors are associated with this instance so it provides more information gain.

For the data distribution to remain we do not eliminate data with a small gradient. This could negatively affect the accuracy of the learned model. Luckily GOSS provides the possibility to take into account the data distribution and sample data based on gradients (Yildirim, 2020).

1. Sort data instances by the absolute value of their gradients.
2. Top ax100% instances selected.
3. A random sample of remaining instances of size bx100% is selected.
4. When information gain is calculated random sample of small gradients is multiplied by a constant equal (1-a)/b.

GOSS eventually achieves that the model's focus shifts to data instances that cause higher loss while having little effect on the data distribution.

2.4.2 EFB (Exclusive Feature Binding)

A dataset with possibly many features is likely to contain lots of zero values. Usually, the sparse features are mutually exclusive, which means it does not have simultaneously non-zero values.

To integrate these mutually exclusive features into a single feature, EFB uses a greedy algorithm. This reduces the dimensionality and reducing the training time of GDBT. The accuracy is not affected due to the complexity of creating feature histograms and is now proportional to the number of bundles, not to the number of features (Yildirim, 2020).

One of the difficulties with EFB is determining the best bundle. The problem was solved by Microsoft's researcher by converting the bundling problem to a graph coloring problem. The graph coloring problem adds edges between features that are not mutually exclusive and takes features as vertices.

Finding the optimal bundle is one of the challenges with EFB. Microsoft researchers designed the algorithm to convert the bundling problem to a graph coloring problem. The graph coloring problem takes the features as vertices and between features that are not mutually exclusive, it adds edges.

The algorithm takes it even one step further and allows bundling for rarely non-zero values simultaneously. That is to say, it bundles even almost mutually exclusive.

Another aspect to consider for the algorithm is the challenge of extracting the original feature from the bundle of features. Let's imagine we have a three-feature bundle. The ability to determine the value of these features using the bundled feature is required.

Bins for continuous values are created using a histogram-based technique. Exclusive values of features in a bundle are placed in different bins to address the issue of merging features. This can be done by adding offsets to the original feature values. (Yildirim, 2020).

According to (Ge, Gu, Chang, Cai, 2020) the number of features is reduced and the speed of the algorithm improved due to EFB optimization. Making LightGBM algorithm outperforming other machine learning algorithms in speed and accuracy.

2.5 Perceptron

This algorithm was first proposed by Frank Rosenblatt in 1943 and later refined by Minsky and Papert in 1969. It's the simplest form of a neural network. It makes predictions using a linear predictor function. To make a prediction, a combination of weights and a feature vector is used. The name perceptron comes from the basic unit's name of a neuron and in its most basic form of usage is found in the binary classification of data. According to (Alsmadi, Bin Omar, Noah, Almarashdah, 2009) the perceptron can be seen as the simplest kind of feed-forward neural network.

Multilayered Perceptron (MLP) is trained the same way as a single layer perceptron. It is a feedforwarded artificial neural network that can map input data to the appropriate output. Backpropagation is utilized as the learning process for neural network training. In MLP artificial neurons can utilize any arbitrary activation function (Saji, Balachandran, 2015).

The MLP is the most widely known multilayer perceptron. Because it requires a desired output to train/learn, it is known as a direct, or supervised network (Sethy, Panda, Behera, 2016).

In the present day, Perceptron is considered a very important algorithm within the science of artificial intelligence. Within the scope of the problems, it can be applied to machine learning it is considered reliable and fast for solving its problems (Goyal, 2020). Figure 4 presents an overview of a single-layer perceptron.

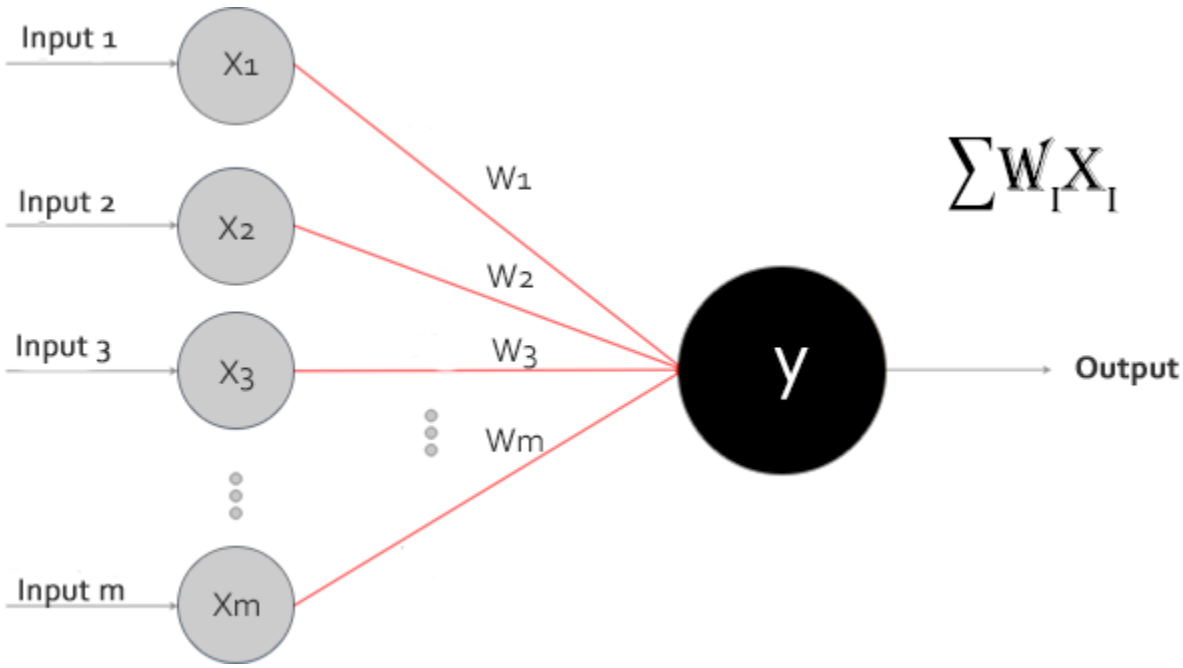


Figure 4 Modell overview of single-layer perceptron algorithm, $\sum W_i X_i$

2.5.1 Perceptron components

1. **Input**:
   are taken as features in the perceptron algorithm. A Bias is a special input. See figure 5.

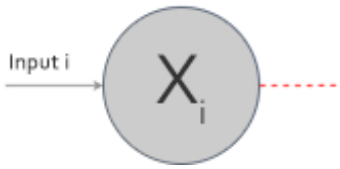


Figure 5 Perceptron Node

2. **Weights**:
   The values calculated during the training of the model are called weights. At the start, an initial value is given to the weights. At training, the value of the weights is updated at the occurrence of error. See figure 6.

Figure 6 Perceptron Weight

3. **Bias**:
   The special value bias allows for the classifier to move around the decision boundary from its original position. Its goal is to move each point a certain distance in a given direction. For training a model, bias perceptron can function faster and with higher quality.

4. **Activation/step function:**
   This is used to create non-linear neural network values. The conversion of value makes it easier to classify a data set. The step function can be used depending on the value required and should be continuous and differentiable.

5. **Weighted summation**:
   The corresponding weight values associated with the input value multiplied give us a sum of values called weighted summation. $\sum W_i X_i$ **for all i -> [1 to n].** See figure 7.

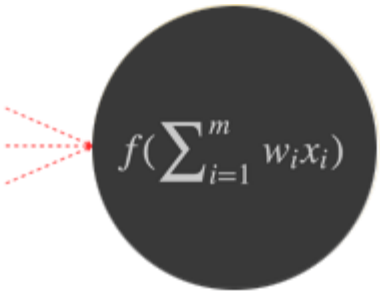


Figure 7 The equation for Weight summation. Where f (…) is the summation.

2.5.2 Perceptron steps of execution

Feed the input required to the model. Weights and inputs are multiplied, and the sum of the results is calculated. The bias value is applied and the output function is shifted. The type of activation required is determined by the value presented. The output value is the final step.

2.6 Summary of background

The background of this study aims to inform and educate the reader to better understand the different elements of machine learning. The background chapter aims to present an overview of important elements related to the use and understanding of machine learning. To understand future chapters, it is essential to understand the areas covered in the background chapters.

The dataset used to conduct the comparison is elementary for the reader needs to understand the meaning of supervised learning. The environment used for conducting this study is the ML.Net platform and the algorithms selected for comparison are the LightGBM and Perceptron.

When working within the science field of machine learning different kinds of errors can occur like bias, variance, over and underfitting. It is of great importance to clarify the cause and consequences of these erroneous scenarios to better understand the concepts and problems that lay ahead in the world of machine learning.

3 Problem

(Måwe I, 2019) Mentions that the wage differences in the Nordic countries are on average 14.3% according to Eurostat statistics from 2017. The gender-segregated labor market is one of the reasons for the wage gap. We can find sectors where women and men are segregated. But even in the so-called standard-weighted statistics with corrections, we can see gender-based wage differences.

The Swedish law for gender-salary segregation can be found in Diskrimineringslagen chapter 3 paragraph §8 and is stated as.

”Alla arbetsgivare ska årligen kartlägga och analysera bestämmelser och praxis samt löneskillnader mellan kvinnor och män. Syftet är att upptäcka, åtgärda och förhindra osakliga skillnader i lön och andra anställningsvillkor. Arbetsgivare med minst tio anställda ska skriftligen dokumentera arbetet.”

For Sweden to better implement the laws and measures to achieve equal pay between genders, a new authority was established on January 1st, 2018. With this law in place, it's mandatory for larger businesses and organizations to once a year document, map, report results, and analysis. The documentation also has to describe what measures are taken to counter salary gaps between genders.

The data used to train the model is based on grouping equal or equivalent kinds of workgroups. This means that we are splitting coworkers into groups depending on what they do. Based on these kinds of data the model trained is used to predict the reason for a pay gap between different groups. Every group with a higher percentage of 60% women is deemed as a female-dominated group and is compared to all male-dominated groups that are equal or equivalent. To analyze and conclude factual reasons for differences in equal or equivalent groups is not always an easy task. To provide a tool using machine learning that facilitates the field of analyzing equal or equivalent groups could lead to a better decision support system.

The two chosen algorithms to compare were LightGBM and the Perceptron. There are of course many other algorithms that could be used for this task, I have for this study limited it to two different algorithms. After doing some research the selection fell on LightGBM and the Perceptron algorithm. Both are found in the Microsoft ML.Net framework for machine learning. The reason for selecting LightGBM from Microsoft researchers was that it is a fairly modern classification algorithm. In short, according to (Ke, Meng, Finley, 2017) it is a Gradient Boosting Decision tree with high efficiency because of the implementation of *Gradient-Based One-Side Sampling* (GOSS) and *Exclusive Feature Bundling* (EFB) making it fairly interesting for the study.

The other chosen algorithm is the Perceptron. This is another type of classification algorithm based on an artificial neural network structure. This is also one of the earliest created algorithms within machine learning science. With time the perceptron algorithm has evolved to multilayer capabilities. The differences between the selected algorithms are the reason for their selection together with their separate capabilities.

Despite extensive research, no studies were found that resemble this study. Though some studies covered salary differences in another aspect. The main difference between the related

studies and this one is that this study focuses on differences between female and male-dominated workgroups rather than comparing male and female salaries.

3.1 Aim

This study aims to create two machine learning models based on different classification algorithms and compare them. Depending on the result evaluate if they are suitable for a possible support system in a real-life scenario.

3.2 The research question:

Which of these two algorithms, LightGBM and Perceptron would predict the most accurate explanation between a female-dominated workgroup compared to an equal or equivalent male-dominated workgroup?

The motivation for conducting this study is to compare two different classification algorithms to see if which one of these is a better suitable algorithm for implementation in the decision-making process. This study was conducted in cooperation with SysArb to evaluate a potential machine learning process to assist in analyzing equal or equivalent workgroups. With the possibility of implementing this machine learning process, SysArb could potentially improve the analysis process. Although SysArb is the beneficiary of this study many other corporations, organizations, and county councils could make use of this. It is also important to note that result of this study does not claim to provide a definite answer on which algorithm is more suitable for all possible scenarios.

3.3 Hypotheses

H_1: The LightGBM with its two optimization methods EFS and GOSS achieves better results in comparison to the Perceptron algorithm.

3.4 Research method

An experiment is set up to conduct this study. The key element to succeed in the experiment is to have a dataset that is relevant to the study. Together with the team at SysArb, we have extracted a dataset that is used for this assignment. The dataset is divided into two separate datasets. The first dataset is used to train the two different machine learning models. The next step in the process when the machine learning models are created is to test the models to see how they compare to each other. The second dataset is the test dataset and is used to test the two different models.

The quality of the dataset is the most fundamental part of the machine learning process. If the dataset contains relevant features, then the result improves. Even the amount of data is of importance. With a larger dataset, the probability increases for the trained model to predict more accurately, and the ability to improve increases.

The environment for this experiment is Microsoft's machine learning platform ML.Net. A console application is created that reads the training dataset and creates two different machine learning models for each of the selected machine learning algorithms.

The two different algorithms chosen are LightGBM and the Perceptron from the ML.Net platform.

3.4.1 Experiment

For this study when comparing two different machine learning algorithms, the most suitable evaluation method is to conduct a controlled experiment. The possibility's given to evaluate the study is more control of the environment. This ensures that the comparison made is equal and easily replicated for the future.

3.4.2 Alternative methods

Case study

This is not an option because the effect of this would mean giving up control over the environment. This is not the most suitable option for this study. It is difficult or sometimes not optimal for a real-life environment to ensure all necessary variables are required for a comparison. Sometimes the same environment cannot be offered on equal ground and could have an unfair advantage. For this study total control of the environment is a must to ensure equal conditions.

Survey

Surveys are conducted by questionnaires or by interviews of a known population. This is not suitable in a comparison study. The aim cannot be answered by this research method.

3.5 Evaluate

The models are tested against the second dataset to be evaluated. The prediction test result is evaluated using the metrics listed below.

Micro-Accuracy: is the fraction of instances predicted correctly. The closer to 1.00 the better. If a class imbalance is suspected that is to say we have one class with more instances than the rest then this could be a more useful metric.

Macro-Accuracy: the average accuracy at the class level. Each class is computed and the average of these computed classes is the macro-accuracy. It gives the same weight to each class no matter how many times the instance occurs in the dataset. The closer to 1.00 the better.

Confusion matrix: makes it possible to visualize the performance of an algorithm especially in the case of a supervised learning one. Here we can read the recall and precision values obtained by the metrics. See table 1 for an overview of the confusion matrix.

Actual / Predicted	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Table 1 Overview of a confusion matrix.

Recall: Is the true positive rate or sensitivity. **Precision**: is the positive predictive rate (PPV). See figure 8 for the definition of Recall and Precision.

$$\text{Precision} = \frac{tp}{tp + fp}$$
$$\text{Recall} = \frac{tp}{tp + fn}$$

Figure 8 Definition of Recall and Precision.

McNemar's test: McNemar's test is used to confirm statistical significance.

3.5.1 Validity threats

When experimenting, it is important to realize what kind of threats could affect the result in a non-positive way. It is important to take into regard these different elements so the result is as correct as possible.

Construct validity

When working with machine learning the collected dataset for training the model is the most essential part. A dataset with clean data and relevant data would make the learned model predict a better result. For the experiment, the data used can be collected from different customers of SysArb. The quality of the dataset is depending on the experience and competence of the customer to tag the reason why inequality exists between two different workgroups. This can result in a tag being wrongfully applied. This has been observed by the consults at SysArb.

To avoid using data tagged by customers and not be sure of the quality of the chosen data. For the experiment in this study, the obtained data has only been selected from customers where consultants from SysArb have been involved in the process of tagging differences between workgroups.

When filtering the dataset to clean it as much as possible it is necessary to separate it into two different datasets. One dataset is used for training the model. The other dataset is used to test the model. This has to be done to avoid using model predicting on identical data points and get a result that would make the model predicting too good.

Another issue to handle when training the model is to use a dataset that is not sufficient. This can lead to a machine learning model predicting with too few reference points. The result is an underachieving machine learning model.

Bias can occur if the obtained dataset differs from future datasets. That is to say, not all of the potential tags are represented. This could lead to the model being able to predict lesser-used tags.

The result gathered from the obtained dataset compiled for this study does not represent all the available data. To conduct this study, we have been selective in the data used to train the model. Data that were insufficient or unfavorable for this study were filtered out. So, this study does not apply to all cases of salary evaluation.

For the study, all constructs must be clearly defined. To avoid misunderstanding it is important to clearly define the metrics used to compare the models so the definition of being better is clearly understood. The metrics used in this study clearly define the performance of the two different models used. To avoid misunderstanding or misinterpretations multiple measurements are used as complements to each other. These different metrics give us different insights into the models to handle this threat.

External validity

If the conducted experiment differentiates from the conditions of the real world, then the obtained results do not reflect reality. The used environment in our study is equal in the experiment as in the real world. The dataset obtained for this study also reflects the data from the real world and the environment is the same as the real world.

This study was conducted by a student with no prior experience in machine learning. The lack of experience compared to an experienced researcher could be seen as a potential threat. With supervision from an experienced supervisor, this study has been carried out in consultation with the supervisor to ensure good design choices.

Internal validity

Choosing two models, there may be one that would perform better. When splitting up the training dataset and the testing dataset could have an impact on the models. When choosing the framework to use, other frameworks could have better algorithms and train better models.

Conclusion validity

The dataset used in this study is limited to a training dataset with around 500 posts and the testing dataset is limited to around 200 posts which could be a cause of *low statistical power* validity threat. The character of the dataset is heavily imbalanced which could affect the outcome by allowing a certain class to be better trained than the others.

4 Related research

After a lot of searching for earlier or related research, I came up with nothing that resembles exactly the research conducted in this paper. Machine learning in this experiment is about finding the reason why there are differences in salaries between two equal workgroups comparing two different machine learning algorithms for the task.

The closest I came to finding related research is one predicting salary classes of employees using different classifications algorithms. (Chakraborti, 2014) found that Decision tree classifiers and Bayesian networks under certain circumstances performed better in predicting the salary class. The reason for this lies in the basic structures of the features used in the dataset.

Another study within the field of gender discrimination using machine learning was conducted by (Alatrista-Salas, Esposito, Nunez-del-Prado, Valdivieso, 2017). This study uses three different clustering techniques to measure the degree of gender discrimination. The results show us that salary differences are bigger in England than in Spain. Even though this study is not examining gender inequality in the same sense that this study does, it shows the possibilities of applying machine learning within the scope of gender discrimination.

LightGBM is a decision tree algorithm that can be used for classification, regression, and ordering (Ke, 2017). According to (Sun, Liu, Sima, 2020) LightGBM outperformed SVM and RF in robustness. Making LightGBM effective in forecasting and managing a large number of data instances with large features. Making this a highly interesting algorithm to use when predicting a large volume of salary data.

The Perceptron is one of the oldest artificial neural network machine learning algorithms. With time it has developed and become more effective. According to (Lorencin, Ivan, 2020) the perceptron algorithm has been implemented in medical research over the years. With so many years that this algorithm has existed within the data science of machine learning and that it differs from LightGBM makes it more interesting to compare them against each other.

In the year 2017, the laws of gender-equal salaries were strengthened as a result the gap between gender salaries decreased. Instead of a mandatory salary survey every 3 years it was changed to every year. As described by (Strömsten, Gustavsson, 2019) they're still exists gender inequality today. The study focuses on the experience of the impact of the new law and to see if there are structures in society today that could potentially cause the new law not to achieve any result.

A problem with the study by (Strömsten, Gustavsson, 2019) is that several of the published references are made by the same author and thus cause a problem of objective sources. (SCB 2020c: 76) informs us that generally, women have lower salaries than men even in female-dominated professions.

5 Method

5.1 Dataset

For the completion of the experiment, the first thing that needed to be asserted is the dataset and its features. The data source for this study is real-world data from SysArb, which is used to create machine learning models. The dataset must contain enough samples and relevant features for machine learning to apply to it.

The dataset is split into two different datasets. The first is used for training machine learning models, and the second is used to validate the models' prediction capabilities.

The training dataset contains approximately 500 rows of data. While the test dataset contains approximately 200 rows. The features of the dataset are illustrated below as seen in table 3. The dataset used contains several features for predicting the reason for the difference between different workgroups, see table 2. The predicted labels used to determine the difference are the following both in Swedish and translated to English. In the used dataset the labels are tagged only in Swedish. The used dataset is limited to one consultant working at SysArb to ensure the quality of the tagged data due to the experience of the consultant.

This study is not about comparing age or sex when analyzing inequality between genders but focuses on analyzing gender-dominated workgroups. The size of the dataset is limited to ensure the quality of the data. The retrieved data is heavily imbalanced.

Na (övrigt) / Others
Erfarenhet / Experience
Kompetens / Competence
Utökat ansvar / Expanded responsibility
Utökat ansvar, Erfarenhet / Expanded responsibility, Experience
Alternativ arbetsmarknad / Alternative labor market
Erfarenhet, Alternativ Arbetsmarknad / Experience, Alternative labor market
Utökat ansvar, Alternativ arbetsmarknad / Expanded responsibility, Alternative labor market

Table 2 List of different causes of inequality between workgroups

Average Pay	Average Pay Diff	AID	Tag

Table 3 The features of the dataset.

Features of the dataset for model learning.
Average Pay: This field indicates the average salary for a workgroup.
Average Pay Difference: This field means the difference in average salary between the two comparative working groups
AID: Aid identification is a system for grouping tasks for municipalities and regions. It is intended to be able to analyze wage formation to provide a basis for certain planning.
Tag: is the predicted label that explains the reason for the difference between the two different workgroups.

5.2 ML.Net

The chosen platform for creating and running the application is ML.net because it is used by SysArb and by using the ML.Net platform the application can be constructed by using C#. The C# language is very familiar thus makes the process of creating the Machine learning application easier. The application is a console app that reads the two different datasets and creates two different models based on two different classification algorithms.

ML.Net is Microsoft's machine learning framework for developers. It was introduced by Microsoft in March 2018. With the ML.Net platform, you can train and build custom machine learning models. Included in the ML.Net platform are features like AutoML (automated machine learning), ML.Net CLI, and ML.Net model builder. This makes it easy for developers within the .net community to integrate machine learning into applications.

This framework can be used with other tools like Tensorflow, CNTK, and Accord.Net. At least 80 features and 40 machine learning models are supported in the ML.Net platform (Alexan, Alexan, Stefan, 2020).

ML.Net can be applied in many different scenarios such as sentiment analysis, price prediction, sales forecasting, product recommendation, image classification, object detection, and much more. The pipelines in ml.net due to their composition of multiple steps allows for a transformation of data input into ML models (Alexan, Alexan, Stefan, 2020).

5.2.1 High-level architecture of Ml.Net

1. **IDataView**: store loaded training data into memory
2. **Pipeline**: Create a map to the IDataView object to send values for model training.
3. **Fit**: initiates the training of the model.
4. **Save**: Saves the model.
5. **ITransformer**: Loads the model back to memory to run predictions.
6. **Evaluate**: Evaluates the model.

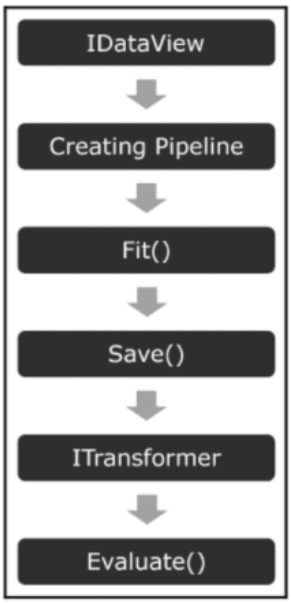


Figure 9 The high-level architecture of ML.Net

5.2.2 Extensibility of ML.Net

As a robust framework, ML.Net also provides extensibility to other externally trained model types. Like TensorFlow from Google. One of the more popular models of TensorFlow is the image classification model. ML.Net's adoption of the ONNX format allows for greater extensibility.

5.2.3 Setting up the environment

In visual studio create a .net core console application. To add machine learning capabilities to your application you can add this to your project. Right-click the project file. See the image below for an illustration. See figure 10.

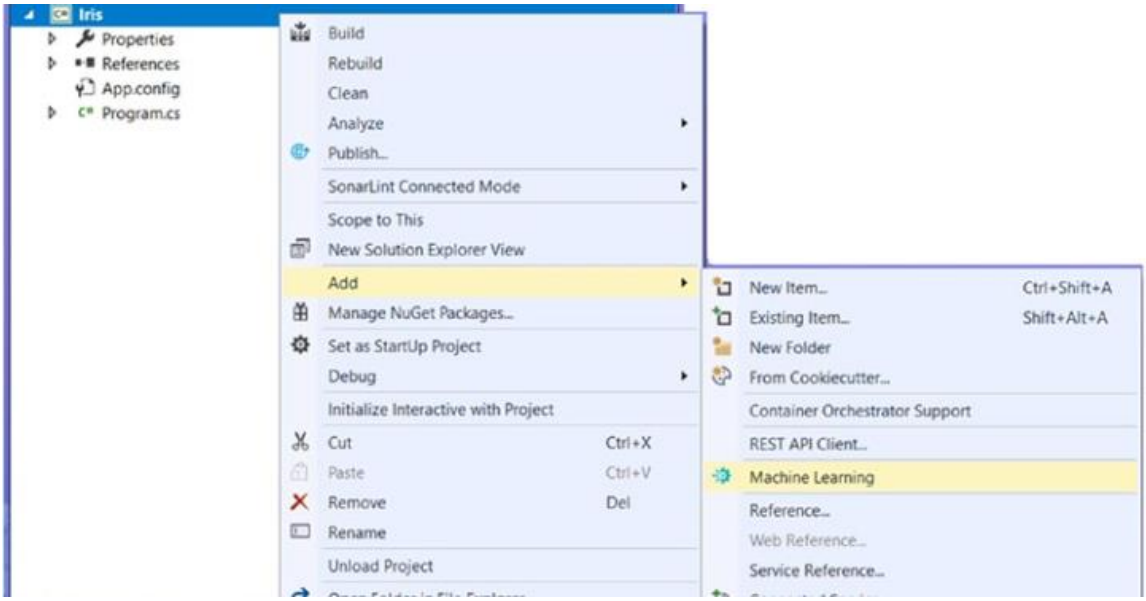


Figure 10 Adding machine learning to your project

Adding this to your project gives access to the model builder guide. See illustration below (see figure 11).

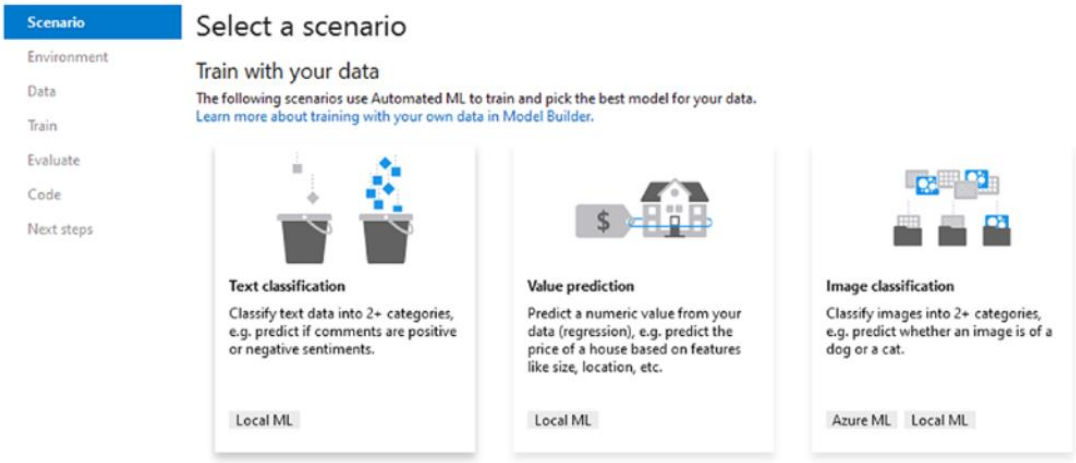


Figure 11 Model builder guide in visual studio.

Now just follow the guide by choosing the text prediction. This step-by-step guide takes your dataset, trains your model, and evaluates it. Before allowing it to train your model you have to set a timeframe for different kinds of algorithms. You can set it to 200 seconds. It does not allow you to choose which algorithms to use. It tries many different forms of classification algorithms and lists the one that is being better suited for you depending on the time it was allowed to train and your features in the dataset. When training is done you can add the solution to your project. Now when machine learning capabilities are added, modifications are done to the project so that the algorithms used are relevant for this study.

5.2.4 Modifications to the applications

A new class was added to the application named ModelBuilderTwo.cs. This new class is a duplicate of the class ModelBuilder.cs but adjusted to work for a separate classification algorithm. In the experiment, the first model builder was adjusted to work with LightGBM and the second one was adjusted to work with the Perceptron Algorithm. As shown in chapter 5.2.6 Training two different models, the code for using the respective algorithm is presented.

Make sure that the Microsoft.ML and Microsoft.ML.LightGBM packages are installed in your project for the experiment to be able to run.

5.2.5 Essential building blocks in ML.Net

The Context

Instantiating the MLContext is easy and In Ml.Net everything starts with the context object which is an encapsulated *MLContext type*. Different properties and capabilities are offered with this type to start a specific machine learning task. MLContext can be explained as the roof of the machine learning pipeline.

```
MLContext mlContext = new MLContext();
```

Data loaders

Offers many features to load data from multiple sources easily. This is important because data can come in different formats. Through mlContext.Data all loaders can be accessed. The IDataView component is specifically introduced for ML.Net and is the fundamental data pipeline type. There are the input and output Query Operators. IDataView enables integration with external machine learning frameworks and able to seamlessly integrate several data loading capabilities. Different ways to load data could be from binary files, text files, or databases.

For cases where data can be messy the usage of a filter could be very useful to clean data. This capability is provided by mlContext.Data.Filter.

Transformers

Transforming data before using it to train machine learning models is important. Data feed directly to the machine learning without processing like scaling, cleaning, or normalizing can cause the algorithm to be confused. The result can be off, or it is biased which makes the output useless and unacceptable. ML.Net offers several different transformers that can clean data from being messy. Clean data is data that is not missing values or out of range values.

Trainers

ML.Net offers different trainers depending on the needs of machine learning. They are added in the final step of the machine learning pipeline. A trainer is an algorithm that takes the data and provides a model which can be used to predict future values.

5.2.6 Training two different models

The first model is created with the LightGBM algorithm and the second model is created with the Perceptron algorithm. Both of these algorithms are supported within the ML.Net framework.

LightGBM

In C# to create the machine learning model we use the following code.

```
var trainer = mlContext.MulticlassClassification.Trainers.LightGbm(labelColumnName:
@"Tagg", featureColumnName: "Features")
.Append(mlContext.Transforms.Conversion.MapKeyToValue("PredictedLabel",
"PredictedLabel"));
```

Perceptron

For the creation of the machine learning model, we use the following C# code.

```
var trainer =
mlContext.MulticlassClassification.Trainers.OneVersusAll(mlContext.BinaryClassificatio
n.Trainers.AveragedPerceptron(labelColumnName: @"Tagg", numberOfIterations: 10,
featureColumnName: "Features"), labelColumnName: @"Tagg")
.Append(mlContext.Transforms.Conversion.MapKeyToValue("PredictedLabel",
"PredictedLabel"));
```

5.3 Limitations

For this study, there are limitations in the form of the selection of algorithms to be compared. The comparison of algorithms was limited to two from the ML.Net framework because of time limitations for this study. Some features could be added to the dataset with more time and research. Due to privacy issues, all features were not relevant to use for this study to offer total transparency of the data used. The construction of the dataset used in this study limits the volume of datasets available from SysArb to certain customers with matching data structures. The small dataset used in this study is also limited to one experienced consultant working with tagging data to exclude as much erroneous data as possible in the training dataset.

6 Result

6.1 Presentation

The main research question of this study is to compare the two different algorithms for mapping the reason of salary differences between two workgroups that are equal or equivalent. The two algorithms are examined according to the following metrics, Micro-Accuracy, Macro-Accuracy, and Confusion matrix. A single prediction is simply a test run of the trained models and has no bearing on the outcome of the two compared algorithms.

6.1.1 Single Predictions result

Several single predictions were executed to test the probability of the trained machine learning model based on the two different classification algorithms. The result of one of the test case predictions is presented below. The single predictions are just a simple comparison of how the two different machine learning models would predict based on the same random values. These values are not relevant when the comparison is made later in the study to evaluate the machine learning models.

Case 1 Perceptron (see figure 12)

Average salary: 26685 KR, **AID:** 521015, and **Average Salary Difference:** -808 KR.

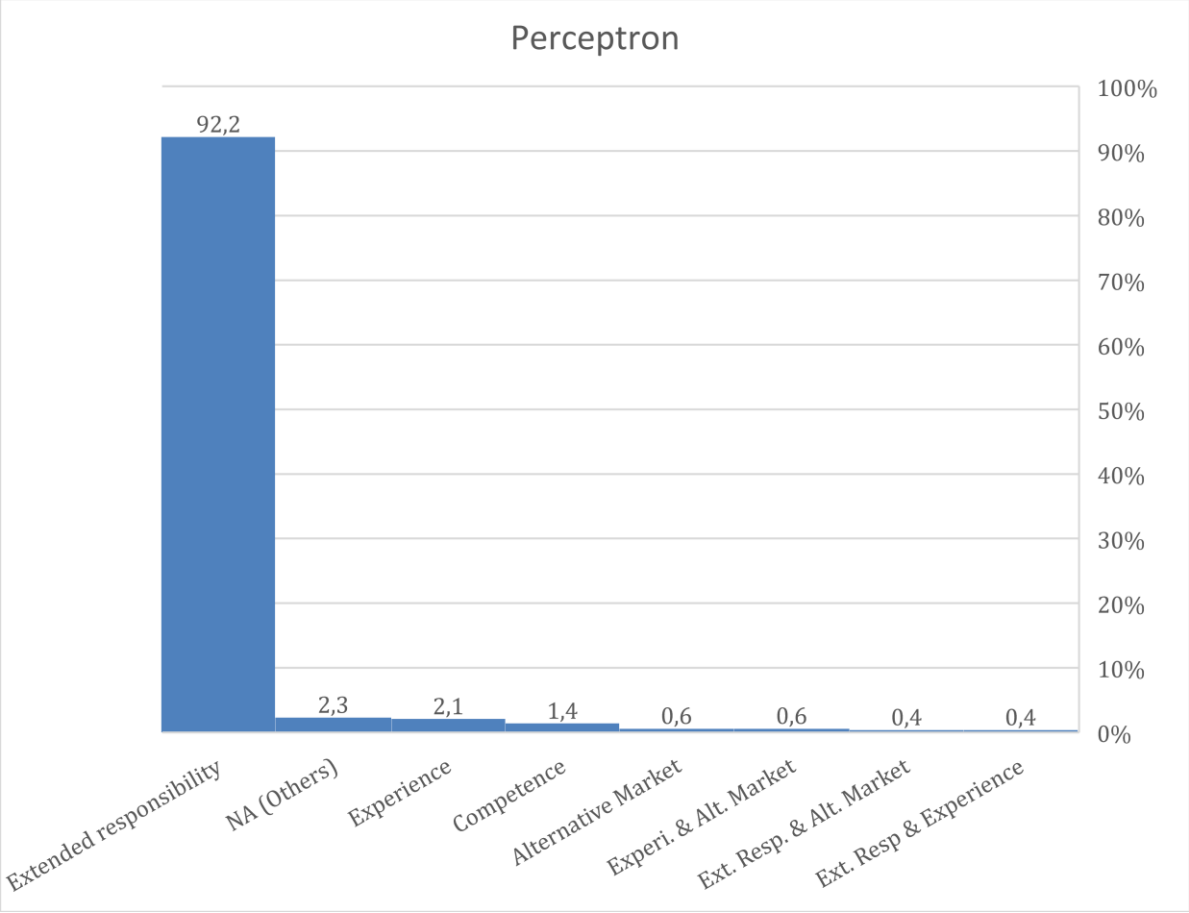


Figure 12 Perceptron case 1 single prediction. Based on the input values we can see that the predicted outcome for the reason of the salary differences is "Extended responsibility". This model was predicted with a 92.2% certainty. In second place is "NA" which means undefined reasons with 2,3% certainty.

Case 2 LightGBM (see figure 13)

Average salary: 26685 KR, **AID:** 521015, and **Average Salary Difference:** -808 KR.

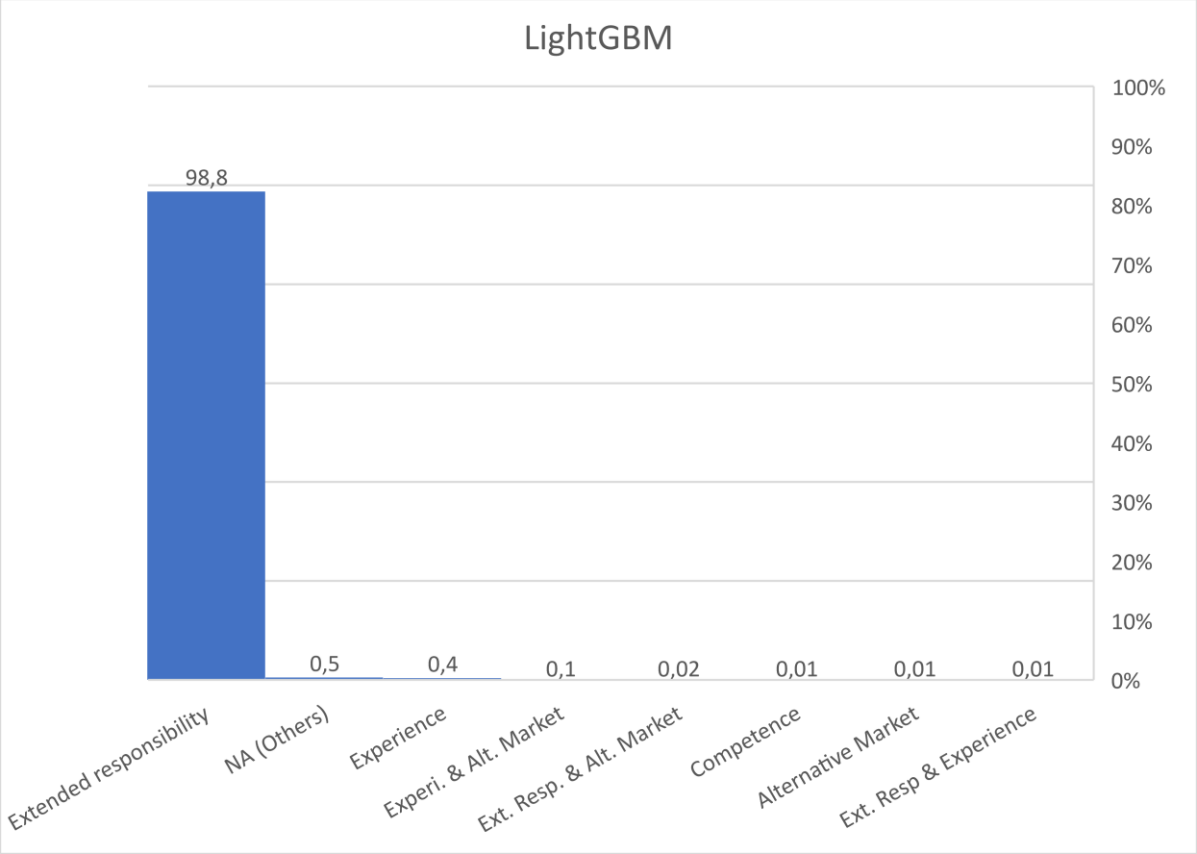


Figure 13 LightGBM case 1 single prediction. Based on the input values we can see that the predicted outcome for the reason of the salary differences is "Extended responsibility". This model was predicted with a 98.8% certainty. In second place we find "NA" which means undefined reasons come with 0,5% certainty.

6.1.2 Micro Accuracy result

Micro Accuracy is the precision of the aggregated contributions of all classes measured. High scores can be achieved with micro accuracy even if the model performs poorly on a rare class because the model gives common classes greater weight. The advantage of this metric is that the value obtained does not take into account classes with very low representation affecting the overall metric heavily if the model's more common classes have better performance. This could be of interest because we have a class imbalanced dataset. The closer the score is to one, the better. If the problem involves more than two classes that can be true, the micro accuracy score may be of relevance. Micro accuracy tells us how often does a workgroup gets classified to the right cause of the difference.

LightGBM:

The achieved values for these metrics when evaluated for Micro Accuracy is: 0,4277

Perceptron:

The achieved values for these metrics when evaluated for Micro Accuracy is: 0,4162

6.1.3 Macro Accuracy

The average accuracy at the class level is referred to as Macro-average accuracy. Each class's accuracy is calculated, and the macro-accuracy is the average sum of these accuracies. Essentially, each class contributes the same amount to the accuracy metric. Equal weight is given to the large and minority classes. No matter how many instances from a class exist in the dataset the macro-average metric applies the same weight to all classes. With macro accuracy value obtained the closer to one, the better. The dataset used in this study is heavily imbalanced and this makes the macro metric less important. In the case of single-label multi-class predictions, this metric is of interest.

LightGBM:
The achieved values for these metrics when evaluated for Macro Accuracy is: 0,7555

Perceptron:
The achieved values for these metrics when evaluated for Macro Accuracy is: 0,8597

6.1.4 Confusion matrix

See table 4 for the list of classes compared in the confusion matrix. Figure 14 shows the obtained metrics for LightGBM and figure 15 shows the obtained metrics for Perceptron. Confusion-matrix is a suitable tool for describing the performance of a multilabel classification algorithm in the event of imbalanced classes. The Confusion Matrix is a performance metric for a classification system with two or more output classes. (Nabi, 2018). The recall is also referred to as sensitivity and is used to measure the percentage of the correctly identified positives. With the recall value, we get an explanation of the ability of the classifier to find all positives. The definition of precision is true positives divided by the sum of true positives and false positives (Koehrsen, 2018).

0	Na (Others)
1	Experience
2	Competence
3	Extended responsibility
4	Extended responsibility, Experience
5	Alternative labor market
6	Experience, Alternative labor market
7	Extended responsibility, Alternative labor market

Table 4 List of the different tags for the confusion matrix

Light GBM

	0	1	2	3	4	5	6	7	Precision
0	**57**	12	10	11	7	52	2	0	**0,3775**
1	0	**1**	0	0	0	0	0	0	**1**
2	1	0	**2**	0	0	0	0	0	**0,6667**
3	1	0	0	**4**	0	0	0	0	**0,8**
4	1	0	0	0	**2**	0	0	0	**0,6667**
5	2	0	0	0	0	**7**	0	0	**0,7778**
6	0	0	0	0	0	0	**1**	0	**1**
7	0	0	0	0	0	0	0	**0**	**0**
Recall	**0,9194**	**0,0769**	**0,1667**	**0,2667**	**0,2222**	**0,1186**	**0,3333**	**0**	

Figure 14 Result for confusion matrix LightGBM.

Perceptron

	0	1	2	3	4	5	6	7	Precision
0	53	14	10	11	9	52	2	0	**0,351**
1	0	1	0	0	0	0	0	0	**1**
2	0	0	3	0	0	0	0	0	**1**
3	0	0	0	5	0	0	0	0	**1**
4	0	0	0	0	3	0	0	0	**1**
5	2	1	0	0	0	6	0	0	**0,6667**
6	0	0	0	0	0	0	1	0	**1**
7	0	0	0	0	0	0	0	0	**0**
Recall	**0,9636**	**0,0625**	**0,2308**	**0,3125**	**0,25**	**0,1034**	**0,3333**	**0**	

Figure 15 Result for confusion matrix Perceptron.

6.1.5 McNemar's test result

McNemar's test is used to see if there is a statistically significant difference in proportions between two sets of data, see table 5 for an overview of a McNemar's 2x2 contingency table.

	Test 2 positive	Test 2 negative
Test 1 positive	*a*	*b*
Test 1 negative	*c*	*d*

Table 5 2x2 contingency table

	Perceptron positive	Perceptron false
LightGBM positive	146	175
LightGBM negative	171	182

Table 6 McNemar's contingency table

A McNemar's test was used to determine whether there is a significant difference between the two different classifiers. McNemar's test compares the predictive accuracy of the two models using the variables TP, FP, TN, and FN. Resulting in a p-value, see table 6 for the contingency table used to calculate the p-value.

The result from McNemar's test gave a p score of 0,85 which is higher than 0,05 showing that the models are not statistically significant. Meaning that no model is significantly better than the other.

6.2 Analysis

The confusion matrix, micro, and macro accuracy are selected to evaluate the obtained metrics. The F1 Score was excluded in favor of the confusion matrix in this study. With the confusion matrix, we get a bigger overall picture. The ability to visually analyze if classes are correctly and incorrectly classified based on the confusion matrices in figures 14 and 15. Unlike the F1 score which only calculates the mean value of precision and recall. The results of the confusion matrix are the most important in evaluating the obtained metrics.

6.2.1 Single prediction - analysis

Both models predicted the same cause for the salary difference with a fairly high percentage. This is just e test case prediction with the trained models and a comparison of the obtained result. It doesn't tell us anything about the accuracy of the models based on this test run.

In this simple test with some randomly chosen values, both classification algorithms predicted a high degree of certainty. Both algorithms had reached the same conclusion and picked the class “Utökat ansvar / Extended responsibilities”. With LightGBM resulting in 98.8% probability for the first pick. The other algorithm the Perceptron chose the same class with a prediction of 92.2% probability. The second pick “NA” class was also the same second pick for both algorithms. LightGBM predicted 0,5% probability and the Perceptron showed a predicted probability of 2,3%.

This shows that both algorithms show a fairly strong probable prediction for the first choice in the single test prediction case. When comparing the performance of the two machine learning models, the single prediction case has no real impact and is excluded. We can only conclude from the single prediction results that both models chose the same option with a high probability.

6.2.2 Micro-Accuracy - analysis

If we look at the values of the dataset used for testing the model, we can see that we have one class that is superior in representation compared to other existing classes. This shows that we have an imbalanced dataset. So, the result of this metric could be of particular interest for understanding the quality of the model’s multiclass classification task. Micro accuracy overweight’s small quantity labels, so a less frequent cause of difference counts as much as a cause of difference with a high frequency.

By examining figure 15, the confusion matrix for the Perceptron algorithm, it can be seen that the label “NA / Others” (Row 0) has 151 records in the test dataset, and the label “Alternative labor market” (Row 5) has 9 records in the test dataset. This means that the label “Alternative labor market” is overweighted and be counted as equal to the “NA / Others” label. The result of these metrics shows that the LightGBM algorithm achieved a result of 0,4277 and the Perceptron achieved a result of 0,4162. The closer to 1 the better the achievement.

The micro accuracy metrics favor LightGBM by a very small margin. However, the difference in outcomes between the two algorithms is trivial. The performance of both algorithms is quite low when you look at the metrics obtained.

6.2.3 Macro-Accuracy - analysis

Looking at the obtained metrics for average accuracy at the class level, we can see that LightGBM achieved an average of 0.7555 and Perceptron achieved an average of 0.8597. The results indicate that the Perceptron performed better. It is known that the dataset used is heavily imbalanced and this makes the result of this metric less relevant. The macro metric does not take into account class imbalance as the micro accuracy does. In a case where the trained models are used only for single label prediction, the obtained metrics from macro accuracy could be useful.

6.2.4 Confusion Matrix - analysis

When reviewing the Confusion matrix (see figure 14 - LightGBM and figure 15 - Perceptron) for both classification algorithms the results imply that the precision is higher for the Perceptron. Three precision scores are higher for the Perceptron, three are equal and two are in favor of LightGBM. The results here prove that the Perceptron achieves a small advantage in scores in regards to the LightGBM when comparing scores for precision.

When the recall values for both confusion matrices are compared, the perceptron algorithm achieves a higher overall score. However, the differences are sometimes marginal for some values. The label “NA/Others” shows a relatively high value for both compared models. LightGBM scored 0,92 and the perceptron scored 0,96. These obtained values are much higher than the rest of the obtained recall values for both confusion matrices. The skewed distribution of values for the recall is because the dataset is very imbalanced. The recall value tells us how many classifications of all classifications made on a certain label are correctly classified.

Further visually analyzing the distribution of labels in the confusion matrices, shows a high frequency of the label “NA / Other “. The prediction score for this frequently occurred label got 0,38 for LightGBM and 0,35 for Perceptron. These low scores indicate that this label obtained lots of incorrect classifications. This could be a result of underfitting when the model has not obtained enough training and not learned enough patterns from the training data. In this case, the size of the dataset is limited to only around 500 posts and can be seen as a small dataset. The precision score informs us how many classifications of a certain label were correctly predicted.

6.3 Analysis of the result

When comparing different metrics, the results can vary a lot because of the few data points. This can be seen when comparing the results of the single prediction in chapter 6.1.1 and later the same class achieves a lower prediction in chapter 6.1.4. In chapter 6.1.5 we can see the result of McNemar´s test showing a p-value of 0.85 which is higher than 0.05. This implies that it cannot be rejected that the models are equally as good, no model is significantly better than the other at classifying the samples. Comparing all of the different results, the overall scores indicate that the Perceptron classification algorithm slightly exceeded compared to the LightGBM classification algorithm.

For the Micro-Accuracy we can interpret the result in favor of LightGBM with a very small difference. The measured value for Macro accuracy was higher for Perceptron than LightGBM. Micro accuracy is a better method for measuring accuracy in this study due to the unbalanced dataset. This does not mean that a higher score is expected for Micro Accuracy over Macro Accuracy. It only means that we get a fairer picture of how the models perform. The obtained measures from micro accuracy give us the average performance of each class independently of the class frequency. The relatively low score obtained from the micro accuracy score implies that the models have not been trained well enough and this could be the result of the low dataset used to train the models.

Macro Accuracy does not take unbalanced data in regards and predicts without regard to the class size.

The confusion matrix created for both algorithms gives us the possibility to visually analyze the result. Each output label could also be separately evaluated and this makes the confusion matrix useful in this study. When comparing the recall and precision values for both matrices we can see that the Perceptron algorithm achieves higher scores for both recall and precision. We can also see that the dataset contains one tag with a higher frequency than the other tags. This makes our dataset imbalanced. This class also shows an overrepresentation of wrong classifications. This could be a good thing meaning that there is some need for human analysis for this class. For this reason, the imbalanced dataset makes the micro accuracy a bit more relevant to use for examining the metrics obtained. In this case, the difference is so small between the metrics obtained by the two algorithms.

For minority classes, the aspiration is to achieve a higher precision. With high precision, the impact is that when a small minority class is predicted the probability of that class having a correct prediction is high. With the class “NA / Other” the higher recall value is more desirable. This means that the ability to capture “NA / Other” predictions in the right class is more desired and if other classes are predicted it does not matter because predictions made on “Na / Other” should be manually revised by an experienced consultant.

7 Discussion

7.1 General discussion

The process of analyzing the data for different workgroups is time-consuming and could be affected by the experience and knowledge of the responsible consultant. There are many explanations for differences between workgroups and to come to a conclusion can depend on several different factors. Not all differences that currently exist between equal workgroups are based on gender. For example, free markets can lead to higher salaries. If a comparison between nurses and network technicians is to be done and both workgroups are equal. Nurses which is female-dominated and network technician is male-dominated. But if we take a look at the salaries for both groups then probably a difference in salaries would occur and that a network technician obtains a higher salary. One reason for this is that they have a competitive number of employers in the public and private sectors. While the nurse is limited to the same public employer in all of Sweden. The network technician operates in a more open market that competes with higher salaries.

This study has shown that the Perceptron performed better than LightGBM with a small margin when reviewing the metrics obtained by the confusion matrices, micro, and macro accuracy. Thus, showing that the hypothesis did not come true, that a more modern developed and optimized algorithm would achieve a higher score. The artificial neural network algorithm Perceptron achieved a higher score in the confusion matrix. With the recall and precision values in favor of the Perceptron, the LightGBM showed a slightly higher score with the micro accuracy metrics. The fact that the test dataset is imbalanced makes the micro accuracy a more relevant metric to take into consideration in regards to macro accuracy. McNemar's test showed that no model was significantly better than the other.

The overall performance of the algorithms in this study shows that there is room for improvement. The precision score in the confusion matrix was low in some cases. With more research and development this could be applied as a support system. The fields left for improvement are to test other algorithms and add more relevant features to the training dataset.

With time larger datasets can be obtained to train the models to achieve higher prediction scores. Based on this small dataset the Perceptron algorithm achieved some good results. With some research, the dataset could be generalized to include more data and given the possibility to be applied to other datasets. This study is limited to the obtained dataset and cannot be applied to other datasets.

The result is also limited to the features obtained from this dataset. Some restrictions to protect sensitive data are made to keep the data as transparent as possible. In this study, only two were compared. There are many more classifications algorithm that could be included in future comparisons.

7.2 Compare to previous work

It is known that we have a labor market with inequality between genders in Sweden according to (Måwe I, 2019). The efforts made by the Swedish government against inequality were to propose a law that makes it mandatory to annually report and document the result.

With the assistance of a trained machine learning model, the quality of the analysis process could enhance. The support of a trained model could lead to faster and more accurate analyses or even detect human-made errors. The research and development could also continue to be applied in other aspects than just to find reasons for differences between different equal or equivalent workgroups. More experience and development within this field of machine learning may result in it being applied to other scenarios like locating unequal salaries between two equal employees.

The study conducted by (Alatrista-Salas, Esposito, Nunez-del-Prado, Valdivieso, 2017) applied machine learning to compare and evaluate salary differences between different countries. This shows the different scenarios that machine learning can be applied to.

No other study was found that studies gender inequality in the same sense as this study making it a bit difficult to compare.

7.2.1 Dataset

The dataset supplied by SysArb was limited to one experienced consultant. To avoid noise in the dataset a decision was taken to only use quality-assured data. The reason for the decision is that the ambition is to use as qualitative tagged data as possible and to avoid datasets created by someone with insufficient experience.

The selection of features was also limited to one type of structured dataset. This means that features available only apply to the dataset chosen to work with for this study. The dataset obtained has proven to be very imbalanced. This is because of the difficulty of being able to tag data. The obtained dataset reflects reality. One class "NA / Other" appears much more in the collected dataset. Hence the unbalanced dataset. Developing the classes for classification is an ongoing work at SysArb to better reflect reality. One method to balance the dataset is to remove overrepresented classes, this is not a good option because of the limited size of the obtained datasets.

Imbalanced dataset poses a challenge for a machine learning algorithm because they are designed with an assumption of equal distribution of a class in a dataset. This can result in the model's poor prediction ability, especially for minority classes.

The imbalanced dataset is tolerable for this study and can be seen as a positive feature. Predictions that cannot be explained are classified as NA (Others). This can then be seen as the default value. When NA predictions are made this could be a sign for further investigation by the user to verify its correctness.

7.2.2 Ethical aspects

The data supplied for this study is owned by a commercial business. Certain measures have been taken to not disclose information that could harm their business while allowing as much transparency as possible for academic purposes. To conduct this study preparations were made to minimize the risk of producing erroneous results. This is done to avoid presenting misguided performance metrics for the ability to help assist in future research or work.

7.2.3 Socially beneficial aspects

With this study, it has been shown that machine learning can be applied to help assist in the work of salary surveys. The process of salary surveys can be directly associated with an organization for gender equality. In that sense, the result of this project can be applied in the work to strive for a more gender-equal society through achieving more equal workplaces.

7.2.4 Genus

In Sweden, a burning issue is that gender inequality still exists today, with women's lower salaries compared with men´s. The male and female labor force is almost par in Swedish society. Expansion of the service sector and family policies have greatly supported women's employment. But women and men face different conditions in the labor market.

Within the field of equality between men and women, we can see that exists a lot of research by sociologists and social psychologists. In the present day, several different theories exist as to why we have differences in salaries between genders. One of those is the devaluation theory. The essence of this theory is that the value of women's work is not equally valued as the work by men.

7.3 Future work

This study was completed by only a smaller dataset to prove that it is possible to apply machine learning within this field in the future. For further development of a better achieving model, more relevant features could be added to the dataset that helps to improve the capabilities for the algorithms to train a higher-performing machine learning model. Another aspect for improvement is the use of a higher quantity of relevant data to train and test the learned model. With a larger dataset for training a model the higher probability of the model to predict more accurate results.

For this experiment, some basic features were elected and with time and research, this can be extended.

In this study, the two selected algorithms compared are LightGBM and the Perceptron algorithm. For future work, other algorithms could be compared to see if they would have an improved performance than the two selected for this study.

In Sweden, we currently have legislation that compels larger companies to annually revise their salaries and be able to justify differences that arise between genders. Research and development in this area will be compelling for companies like SysArb who specialize in this field. Of course, this could also be applied to evaluate salaries in other aspects than just unequal salaries between female-dominated workgroups compared to male-dominated workgroups.

Bibliography

AI in *Radiology*. *History of machine learning* URL: https://www.doc.ic.ac.uk/~jce317/history-machine-learning.html

Alatrista-Salas, H., Esposito, B., Nunez-del-Prado, M., & Valdivieso, M. (2017). *Measuring the gender discrimination: A machine learning approach*. In 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI) (pp. 1-6). IEEE.

Alexan, A., Alexan, A., & Ştefan, O. (2020). *Machine learning activity detection using ML. Net*. In 2020 IEEE 26th International Symposium for Design and Technology in Electronic Packaging (SIITME) (pp. 188-191). IEEE.

Brownlee, J. (2019). *A tour of machine learning Algorithms*. URL: https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/

Chakraborti, S. (2014). *A Comparative Study of Performances of Various Classification Algorithms for Predicting Salary Classes of Employees*. (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2), 2014, 1964-1972

Chandra. A. (2018). *Perceptron Learning Algorithm: A Graphical Explanation of Why It Works*. URL: https://towardsdatascience.com/perceptron-learning-algorithm-d5db0deab975

Goyal. K. (2020). *Perceptron Learning Algorithm: How It Works*. URL: https://www.upgrad.com/blog/perceptron-learning-algorithm-how-it-works/

IBM Cloud Education, (2020). Machine learning. URL: https://www.ibm.com/cloud/learn/machine-learning

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017*). LightGBM: A highly efficient gradient boosting decision tree*. Advances in Neural Information Processing Systems, 30, 3146–3154.

Khalil Alsmadi, M., Omar, K. B., Noah, S. A., & Almarashdah, I. (2009, March). *Performance comparison of multi-layer perceptron (Back Propagation, Delta Rule and Perceptron) algorithms in neural networks*. In 2009 IEEE International Advance Computing Conference (pp. 296-299). IEEE.

Klosterman. S. (2019). *Overfitting, underfitting, and the bias-variance tradeoff* URL: https://towardsdatascience.com/overfitting-underfitting-and-the-bias-variance-tradeoff-83b42fb11efb

Koehrsen, W. (2018). *Beyond Accuracy: Precision and Recall* URL: https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c

Lorencin, I., Anđelić, N., Španjol, J., & Car, Z. (2020). *Using multi-layer perceptron with Laplacian edge detector for bladder cancer diagnosis*. Artificial Intelligence in Medicine, 102, 101746.

Marr, B. (2020). *A Short History of Machine Learning -- Every Manager Should Read* URL: https://bernardmarr.com/default.asp?contentID=1216

Måwe I. (2019). *Likalön I Norden – Lagar och politiska strategier*

Nabi, J. (2018). *Machine Learning — Multiclass Classification with Imbalanced Dataset* URL: https://towardsdatascience.com/machine-learning-multiclass-classification-with-imbalanced-data-set-29f6a177c1a

Preciado. A. (2021). *Applying a clustering algorithm to feature contribution*. URL: https://towardsdatascience.com/tagged/lightgbm/

Saji, S. A., & Balachandran, K. (2015). *Performance analysis of training algorithms of multilayer perceptrons in diabetes prediction*. In 2015 International Conference on Advances in Computer Engineering and Applications (pp. 201-206). IEEE.

SAS insights. *Machine learning, what it is and why it matters* URL: https://www.sas.com/en_id/insights/analytics/machine-learning.html

Sethy, P. K., Panda, L., & Behera, S. K. (2016). *Ann based image restoration in approach of multilayer perceptron*. In 2016 International Conference on Inventive Computation Technologies (ICICT) (Vol. 2, pp. 1-4). IEEE.

Singh. S. (2018). *Understanding the Bias-Variance Tradeoff* URL: https://towardsdatascience.com/understanding-the-bias-variance-tradeoff-165e6942b229

Sun, X., Liu, M., & Sima, Z. (2020). *A novel cryptocurrency price trend forecasting model based on LightGBM*. Finance Research Letters, 32, 101084.

Yanling, Z., Bimin, D., & Zhanrong, W. (2002). *Analysis and study of perceptron to solve XOR problem*. The 2nd International Workshop on Autonomous Decentralized System, 2002., 2002, pp. 168-173

Yildirim. S. (2020). *Understanding the LightGBM*. URL: https://towardsdatascience.com/understanding-the-lightgbm-772ca08aabfa