Early detection of medical deterioration of patients with diabetes by using machine learning

Andreas Hammerbeck

Master's thesis in Software Engineering at

Department of Computing, Mathematics and Physics,

Bergen University College

Department of Informatics,

University of Bergen

January 2020

Western Norway
University of
Applied Sciences

Acknowledgements

I want to express my greatest gratitude to my supervisors Rogardt Heldal at Western Norway University of Applied Sciences and Magnus Alvestad at Haukeland University Hospital. We have had many exciting discussions of both diabetes and machine learning, which has been a great motivation for me. I would give a special thanks to Rogardt for his understanding as I completed the thesis and started working as a CTO at the same time. Without his passionate participation, this accomplishment would not have been possible.

I would also thank Dr. Alexander Selvikvåg Lundervold, which has been a great help with the technical aspect of the thesis. Even though he was not my supervisor, his door was always open when I ran into troubles or had questions regarding machine learning.

Finally, I want to thank my family and friends, which has given me great support and encouragement throughout my years of study and the process of researching and writing this thesis.

Abstract

Diabetes is a growing healthcare problem in the world, which affects over 400 million adults. In collaboration with Haukeland University Hospital, we look at medical records from real patients diagnosed with diabetes. The study uses machine learning to predict if a given patient has a high risk of experiencing medical deterioration. Further, the thesis goes through the data cleaning necessary to provide such predictions. The first approach managed to identify 79% of high-risk patients. If the model classifies a patient to have a high risk of mortality, the model had an accuracy of 18%. In the second approach, we removed the last four weeks before mortality happens, and the model was able to identify 49% of the patients with high risk. If the model classifies a patient to have a high risk of mortality, the model had an accuracy of 12%.

Contents

List of Figures

1 Introduction

In 2016, there were 206 795 registered medical patients with Diabetes in Norway, and there were 481 amputations where diabetes was the main contributor[2]. World Health Organization(WHO) published a report[24], which stated that the number of adults with diabetes was 422 million in 2016. Furthermore, WHO noted that the number of deaths directly connected to diabetes was 1.5 million in 2012. If deaths related to high blood glucose levels are included, the number of deaths goes up by another 2.2 million people.

Patients with diabetes are expected to manage their decease in multiple ways. The patients need to track exercise, diet, low blood sugar episodes, insulin administration, and foot care. By detecting patients with a high risk of medical deterioration early, the Health Care Service can reach out to these patients and offer assistance. Offering assistance could lead to a decrease in the number of patients that experience medical deterioration, an increase in the quality of life for patients, and reduces the number of cases where diabetes leads to disability. Fewer required medical interventions will save the health care system resources. At the same time, the government will save money since they don't have to pay disability payments. As the number of patients with diabetes is too high for doctors to go through medical records to see if a patient is at the risk of experiencing medical deterioration, this project was completed to see if that process could be automated using machine learning.

A survey from 2012 made by Ponemon Institute found that 30% of all stored data belongs to the healthcare industry.[14] In November 2018, IDC predicted that the world contains 33 zettabytes of data, and it is not slowing down. Further, they predict that in 2025, it will contain 175 zettabytes. [27]. Data itself is not useful, as Ziad Obermeyer and Ezekiel J. Emanuel said: "To be useful, data must be analyzed, interpreted, and acted on." [20] One of the solution is Machine Learning(ML). While a Doctor, with years of training, may be able to look at 200 MR-scannings a day and predict whether or not a brain has a tumor. You can do the same with machine learning, but both training and prediction will take much less time. Machine learning may also be able to predict with better accuracy than doctors. In May 2019, a study[5] compared eight radiologists to a Google algorithm in making predictions on lung cancer screenings. The algorithm beat all radiologists and had an 11%

reduction in false positives, which is the main problem in lung cancer screening predictions, and was able to detect 5% more cancerous human beings than the radiologists.

In collaboration with Haukeland University Hospital, hence referenced Haukeland, we look at medical records for 20.000 patients diagnosed with diabetes. The data contains events from real patients at Haukeland, although some personal data is hashed to prevent it from being easily identifiable. The data required a lot of analysis and cleaning for a machine learning model to be able to make predictions regarding a patient's risk of experiencing medical deterioration.

1.1 Research questions

- **How well can medical record data be used in machine learning to predict medical deterioration of patients diagnosed with diabetes?**
  As mentioned above, a lot of personal factors and lifestyle choices affect diabetes. The data provided by Haukeland is purely medical record data, which lacks information about a patient's lifestyle and personal events.

- **By removing the latest events for a patient, how early can the risk of medical deterioration by identified?**
  Predicting that a patient will need an amputation or die today, gives no value as the healthcare system does not have time to react. For the thesis solution to be of value, it needs to be able to identify high-risk patients as early as possible.

1.2 Thesis Structure

The thesis starts with an overview of the problem description, before giving an in-depth explanation of what diabetes is and its consequences to society. Then the thesis goes through machine learning and techniques used to achieve our findings, which will be beneficial when reading the rest of the report. After the introductory chapters, the report goes through the thesis' research approach, which explains all steps taken in the research, followed by ethics in medical research. Further, an explanation of the structure and processing of the data is given. The next chapters

are about the experiments completed in the thesis, followed by an interpretation of the results and discussion. The thesis finishes with a conclusion and a description of future work.

2 Diabetes

This chapter gives a more in-depth explanation of diabetes and how its consequences affect society.

2.1 What is diabetes?

Diabetes is a chronic health condition that causes the body's insulin levels to be too low or that the body is resistant to insulin. Insulin is used to transfer sugar or glucose from the body's blood and into cells and muscles. If the glucose stays in the blood, it leads to too high glucose levels in the blood. If a person has too high glucose levels over time, it can lead to several health problems, such as eye diseases, heart diseases, or amputation[22]. There exist several medications to reduce blood glucose levels, such as insulin. It exists mainly three different types of diabetes:

Type 1 Diabetes

The body's immune system mistakes healthy cells for unhealthy ones, and attacks and destroys insulin-producing beta cells[25]. As the body destroys the cells, it leads to the body having problems producing insulin on its own. Why the body attacks these cells is not known, but researchers are working on finding out why. For example, TrialNet is a company dedicated to research and prevent diabetes type 1.

Type 2 Diabetes

People with diabetes type 2 have resistance to insulin; the body is still producing it, but can't use it efficiently. As the body has a problem using insulin, the body increases insulin production. Over time, the body can't produce enough insulin, and an increase in glucose in the bloodstream happens[23]. Furthermore, other contributors that increase the risk of developing type 2 diabetes is overweight, obesity, and physical inactivity.

Gestational Diabetes

Gestational diabetes is a type of diabetes women can get under pregnancy., caused by hormone changes in the body during pregnancy, lifestyle, and genes.

A common contributor which increases the risk of developing diabetes is genes. Genes affect type 2 and gestational diabetes more than type 1. Although, American Diabetes Association states that genes alone are not enough to develop diabetes[6]. Genetic mutations, diseases, damage to the pancreas, and medications can also cause diabetes[23].

2.2 Cost of diabetes

One study[31] estimated that in 2011, the total cost of diabetes was 516 million euros in Norway. The total cost of diabetes, 78 million euros, was connected to the "cost of in-patient hospital care of the most common comorbidities in diabetes." Furthermore, the indirect costs, which includes sick leave money, productivity loss, disability pension, and basic benefit and attendance, consisting of 107.6 million euro. By using costed numbers given by Haukeland, we can state that the mean costs for DRG213 - amputation for musculoskeletal system and connective tissue disorders is 167 869 Norwegian kroner(KR) for Haukeland. The mean cost for DRG113 - amputation for circ system except upper limb and toe is 150 670 KR. These numbers are provided by Siw Ottesen Iversen, Controller at Haukeland, and Anne-Mette Tang - Controller at Haukeland, through personal communication in January 2020. By reducing the cost of diabetes, the healthcare industry can increase budgets in other areas and focus on other diseases. Also, by reducing the number of operations needed for patients with diabetes, it can free up time for surgeons and other healthcare personnel. Lastly, by helping patients avoid medical deterioration, it can reduce suffering for individuals diagnosed with diabetes, which is a big reward in itself. Increasing public health can further bring down the cost regarding sick leave money, productivity loss, disability pension, and basic benefit and attendance.

2.3 Managing and risk reduction for diabetes

Some patients with diabetes experience medical deterioration because they do not follow the optimal/planned medication plan. Other reasons can be regarding the patient's lifestyle, such as lack of training or eating unhealthy. There are multiple ways to reduce the risk of experiencing medical deterioration. National Institution of Diabetes and Digestive and Kidney Diseases(NIDDK) have created an ABCs for managing diabetes[21]. These are :

- A for the A1C test, which is a test that shows the average blood glucose level over the last three months.
- B for blood pressure.
- C for cholesterol. There are two types of cholesterol, one "good" and one "bad." Having a high level of bad cholesterol increases the risk of heart attack or stroke.
- S for stop smoking. Both diabetes and smoking affect a person's blood. Stopping smoking can lower the risk of multiple diseases, improve cholesterol, blood pressure, and blood circulation.

Furthermore. NIDDK also states that following a diabetes meal plan help managing blood glucose, blood pressure, and cholesterol. Being more physically active, taking medicines, and regularly checking blood glucose levels will also help to manage the ABCs. Especially important for this project is that they also state that you should work with your health care team. They state that a patient should meet with the health care team at least once every two years or more often if a patient is having trouble managing their diabetes. This statement also supports the assumption that having consultation sessions can prevent medical deterioration.

The goal of the thesis is to be able to identify patients with a high risk of medical deterioration before it is too late. The healthcare system can reach out to patients and help them set up a medical plan to reduce future risk.

3 Machine Learning

This chapter contains basic concepts of ML, which will be useful when reading the rest of the thesis.

3.1 What is Machine Learning?

Machine learning is the science of programming a system/program that can learn from data. Instead of programmers or other involved persons to create rules for a system to follow, machine learning can detect patterns in the data by itself, which can be used to set up conditions and learn something about the data. A post-trained network should be able to make determinations or predictions about data the system has not seen before. [10]

3.2 Types of machine learning

In machine learning, there exist three different types of machine learning, which is supervised, unsupervised, and reinforcement learning. One of the first things we had to do was to define which category our problem would fit in.

Supervised learning

In supervised learning, all your data is labeled. For example, if creating a model that can predict if a patient has diabetes, the model would need data of patients with and without diabetes. The training data contains the answer. Supervised learning is used for classification/categorization or regression. An example of a supervised learning study is "Deep learning algorithm predicts diabetic retinopathy progression in individual patients"[4], where a neural network was created to predict diabetic retinopathy for patients with diabetes. This study used images with binary labels.

Unsupervised learning

Not all data sets are labeled, and sometimes researchers may not know the specific results they are seeking. In supervised learning, the model goes over the data to discover patterns. It is used for clustering and dimensional reduction of the data.

Reinforcement learning

In reinforcement learning, there exists an agent, which makes decisions in a created environment. The agent usually has a goal, such as to find a way out of a labyrinth. When the agent makes an action, it either receives a reward or a punishment. By repeating this step, the agent learns to maximize the rewards and make its way out of the labyrinth. This type of machine learning has been used in a lot of games, for example, in Dota 2[11] or Chess[29]. Without domain knowledge, the agent can learn the best moves in the games and achieve superhuman performance.

In our work, we use supervised learning since we marked the outcome of each patient, such as the severity of the outcome when an adverse event happens, such as eye operation and amputation. We could have used unsupervised learning as well to group patient; however, we decided to use supervised learning since we wanted to use machine learning to predict the outcome. We did not consider reinforcement in this work; however, in the discussion, we will discuss how reinforcement could have been used to improve the behavior of diabetes patients.

3.3 Overfitting and underfitting

The goal is that our model will be able to generalize, which means to give accurate predictions on a large quantity of data, instead of perfect predictions on a small subset of a data set. A model may predict perfectly on training data, but when it predicts on unseen data, it performs terribly. The model has not learned to generalize but learned the exact patterns of the training data. When a model performs well on training data, but not on unseen data, the model has overfitted. An overfitted model usually has a high error when predicting on validation data. Overfitting can be seen in the last graph below.

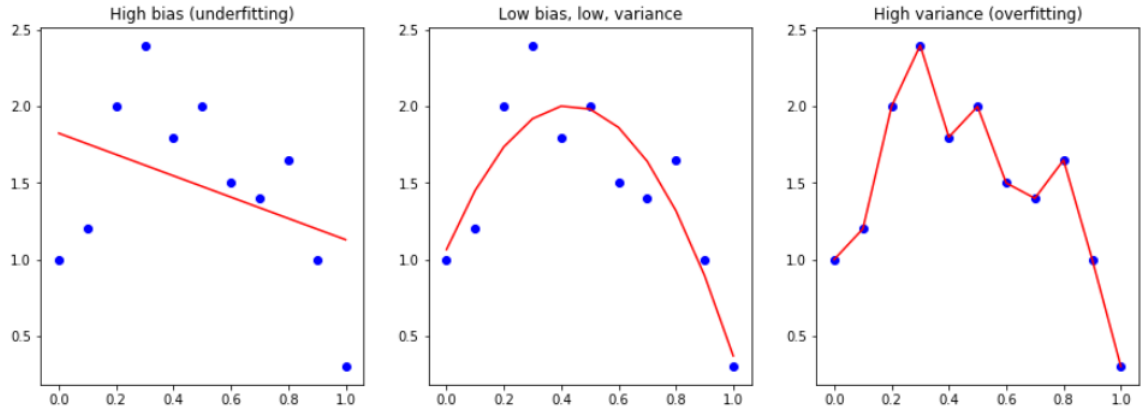

Figure 1: Bias variance tradeoff graph

Underfitting is when a model is unable to detect the patterns in the data, which means the model is not able to give precise predictions on any data. Underfitting usually has a high loss on both training and validation data. The first graph above shows underfitting.

In our model, we want something in between. A model that performs well on both training and the test data. This can be measured by low loss in both training and validation.

3.4 Feature Engineering

Before we could use the data from Haukeland hospital, we needed to do future engineering on it; make the data in a structure that is readable by the machine learning network. In addition, future engineering helps increase the performance of our model. Future modeling is done by removing or altering the data. These steps were a large part of the thesis, and the specific actions will be described in section 7.2. This step is where raw data is transformed into a new dataset, which is easier to interpret for a machine learning model. Feature engineering can be split into two different parts, called feature selection and feature extraction.

- Feature selection: The process of selecting features in the dataset to use during both training and predictions. The goal is to select features that contribute the most to the prediction of the data. Using bad features can negatively affect your model's precision as the model learns on irrelevant data. Feature engineering can give benefits such as increased accuracy, reduced overfitting,

and reduced training time.

- Feature Extraction: The process of creating new features based on existing features. The goal is that the new features are in a lower dimension than the original features, and still preserve data important for prediction. Known techniques to Feature extraction is principal component analysis(PCA), linear discriminant analysis(LDA), and more.

3.5 bias-variance trade off

In machine learning, a goal is to minimize both bias and variance, but lowering one usually increases the other. Bias in machine learning is the average between the predicted values and true values. Models that are too simple for the data have high bias and usually underfits.

When a model has high variance, it has "learned" all the noise in the data, and will predict well on the training data, but perform worse on the test data. The model will overfit. It would be unfortunate if we created a model that was able to identify all patients who had an amputation in our data, but when exposed to a new dataset, it was not able to identify anyone. If we can find the right balance between bias and variance, the model can avoid both overfitting and underfitting.

3.6 Cross validation

20.000 patients may sound like a large amount; however, few patients experienced medical deterioration. We need data for training, validation, and testing. To not further decrease the number of negative outcomes when training the model, we need to split the data in a clever way.

Cross-validation is a technique one can use when working with limited data, so we decided to use this. Cross-validation divides the data into K equal folds. One part is left out and used as a test set. The machine learning runs K-1 training iterations, where the part used as a validation set changes in each iteration. An example is shown in the figure below.

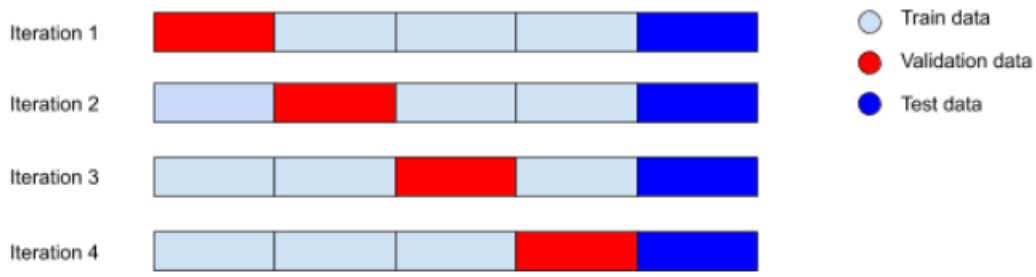

Figure 2: Cross validation

When we used cross-validation, we kept the ratio between the outcomes was kept in each fold, which is called a stratified split. That way, all folds still represent the real data, as the ratio between outcomes is equal in the train, validation, and test set.

3.7 Evaluation

After our model was created, we had to measure how well it was able to identify medical deterioration, and how accurate the predictions are. In machine learning, there exist multiple methods to evaluate a model. Before going through the different techniques used to evaluate our model, we need to understand some general terms for evaluating binary classification.

True Positives(TP) = Both the prediction and the true value is 1.
False Positives(FP) = The prediction is 1, while the true value is 0.
True Negatives(TN) = Both the prediction and the true value is 0.
False Negatives(FT) = The predicted value is 0, while the true value is 1.

A confusion matrix is often used to display a more visual representation of these values and can be seen below.

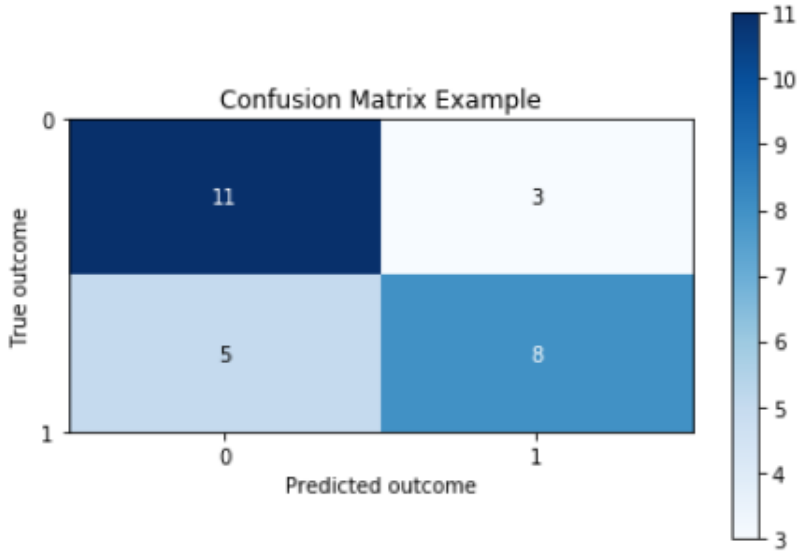

Figure 3: Example of confusion matrix

In the figure above, it visually shows that of 14 cases of class 0, a network predicted 11 correct and 3 wrong. For case 1, the network also predicted 8 of 13 instances correct for class 1. The data shown are randomly generated to explain a confusion matrix.

Accuracy

Accuracy is probably the evaluation method most people think about when speaking of metrics. The formula for accuracy is:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FT}$$

Using accuracy to evaluate a model is fine when there are about the same number of each labels. As our data is highly imbalanced, accuracy would not fit for evaluating our model. For example, if predicting if a person has diabetes or not, and the data have 2 out of 100 persons with diabetes, the model could predict that no one has diabetes. By making that prediction, the model achieves a 98% accuracy.

Precision and recall

Precision and recall are a method for calculating how many instances of a label a model is able to identify and how believable a prediction is. Both precision and recall need to be calculated for each label.

Precision calculates that when a model gives a prediction, what are the probability the model is correct. This means, after our model gives a prediction that a patient have a high risk for medical deterioration, we can state how strongly we can believe that prediction is correct.

$$Precision = \frac{TP}{TP + FP}$$

By using the numbers from the confusion matrix above precision is calculated as followed:

$$Precision = \frac{11}{11 + 5} = 0.69$$

When the model predicts outcome 0, it is a 69% chance that the model is correct.

Recall calculates the models ability to identify all relevant instances of a outcome. This means we can accurately tell how many high-risk patients our model is able to identify.

$$Recall = \frac{TP}{TP + FN}$$

By using the numbers from the confusion matrix above precision is calculated as followed:

$$Recall = \frac{11}{11 + 3} = 0.79$$

With a score of 0.79 in recall, it means the model is able to identify 79% of all cases of class 0.

ROC curve and AUC

Roc curve and AUC, also called AUROC, is often used to compare different tests and can be used to alter the threshold between classifications. We use it to compare both experiments in the thesis.

ROC curve or receiver operating characteristic curve is a graph where the true positive rate(TPR) and the false positive rate is plotted.

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

$$FPR = \frac{FP}{TN + FP}$$

These rates are plotted at different classification threshold.

AUC, or area under the curve, measures the area below the ROC curve, as seen in the figure below.

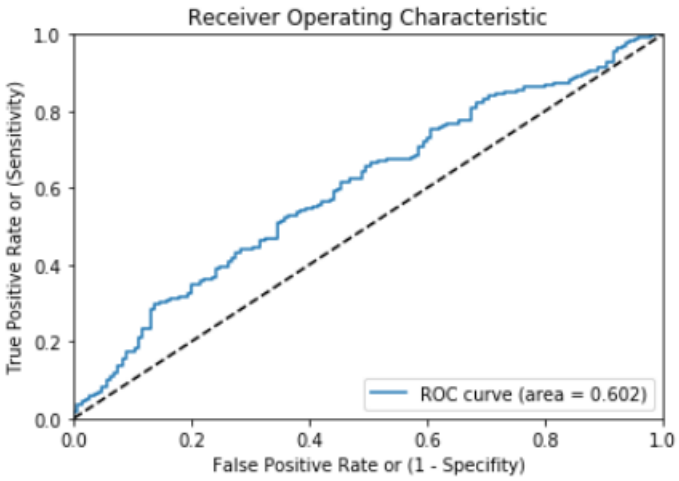

Figure 4: AUROC graph

It outputs a number between 0 and 1. A 0.5 score means the predictions are random, lower than 0.5 means the predictions are worse than random. So the closer to 1 the better. A rule of thumb for translating the AUC score into regular speech is as followed[28][32] :

- .90-1 = excellent (A)
- .80-.90 = good (B)
- .70-.80 = fair (C)
- .60-.70 = poor (D)
- .50-.60 = fail (F)

4 Neural Networks

The type of machine learning model created in this thesis is called a neural network(NN). The data we use is time-series based, and the main reason we chose to use neural networks is because of a particular subtype of neural networks called recurrent neural networks, which is created for processing time-series data. A neural network is a set of neurons, or also called nodes, separated into different layers. For a network to make predictions, data goes through all layers, and the final layer outputs a prediction. Both experiments completed in the thesis are done by creating neural networks.

A neural network starts with a input layer, which has as many neurons as there are features in the dataset. After the input layer, there are one or more hidden layers, which finds patterns in the data. The more hidden layers, the more complex problems the network can solve. The last layer is called the output layer, which outputs the prediction of the network.

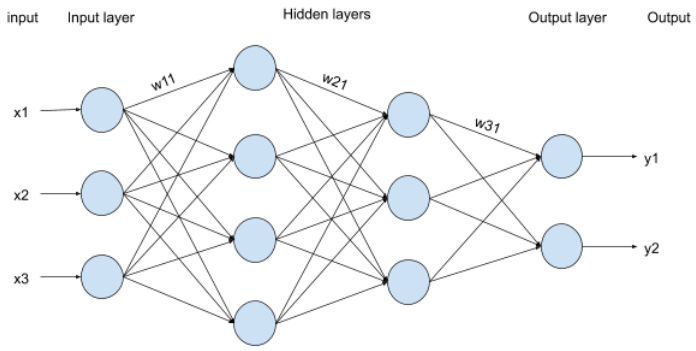

Figure 5: A simple neural network

As you can see in Fig 5, each layer is built up of several neurons, which contain a given weight. When a neuron gets input, it calculates the dot product between the input X and the weights W.

$$X = [x_1, x_2, ..., x_i]$$

$$W = [w_1, w_2, ..., w_i]$$

$$dotproduct = X * W = \sum_{i=1}^{n} x_i * w_i$$

The dot product is then sent to an activation function, which determines the output of the neuron

4.1 Backpropagation

A neural network starts with randomly initialized weights. After initialization, the training data is fed through the network, which is called feed-forward. The next step for the network is to make predictions on the data. After the predictions are made, the network uses a loss function to see how well it performs. Afterward, the network goes backward through the layers of the network and updates the weights with small steps to reduce the error from the loss function. Weights are updated by a learning algorithm called gradient descent is commonly used. The goal of gradient descent is to minimize the result of the cost function by finding a global minimum for the function. To decide how much to alter the weight, we use the learning rate, given as a hyper-parameter, to the model. Backpropagation is an iterative process, which happens multiple times in the training phase of a network.

4.2 Activation functions

Given the input to a node in a NN, the activation function calculates the output of the neuron. By using these non-linear functions, neural network is able to solve complex problems, which would not be possible in a linear space.

Sigmoid function

Sigmoid is a activation function that ranges from 0 to 1. It is often used to predict classes as the output can be directly translated to the probability the data is of a given class. Often the threshold is 0.5, which a network will predict the data is class 1 if the output is 0.5 or higher.

$$Sigmoid(x) = \frac{1}{1 + e^{-x}}$$

Tahn function

The tahn function is very similar to the sigmoid function, although it ranges from -1 to 1.

$$Tahn(y) = \frac{2}{1 + e^{-2y}} - 1$$

Relu function

Relu or Rectified Linear Unit is an activation function that removes negative values. It has proven to work well for very deep neural networks.

$$Relu(z) = max(z, 0)$$

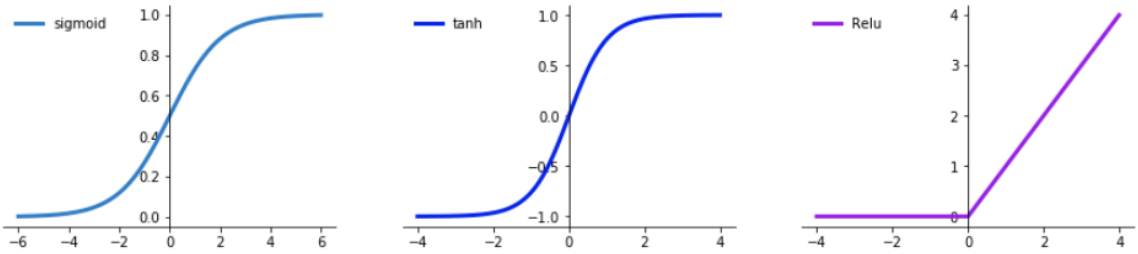

Figure 6: Activation functions

Above is three graphs that display the described activation functions.

4.3 Loss functions

A loss function is a performance metric of how well the model can make predictions on the input data. As machine learning data and problems may vary, there exist multiple different loss functions. The loss output from the loss function is used in the training of a machine learning model. The higher the loss output, the worse the model performs. In training, a model will try to optimize(minimize) the loss by adjusting weights in the model. There exist two main categories for calculating loss, regression, and classification.

Mean Square Error

An example of a loss function used in regression tasks. Calculates the mean error between the correct values and the predicted values. The function for mean square error is:

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{n}$$

n equals the number of samples, y is the correct value, and $\widehat{y}$ is the predicted value.

Binary Cross-Entropy

Binary cross-entropy are often used when solving a binary classification task. It can also be used for multi-label classification problems, if the different labels is independent of each other.

$$BCE = -\frac{1}{n}\sum_{i=1}^{n}(y * log(\widehat{y_i}) + (1-y) * log(1-\widehat{y_i}))$$

n equals the number of samples, y is the correct value, and $\widehat{y}$ is the predicted value.

Categorical Cross-Entropy

Categorical cross-entropy is used for classification tasks when there are more than two classes or in multi-labeling when the labels are dependent on each other.

$$CCE = -\sum_{i=1}^{m}\sum_{i=1}^{n}(y_ij * log(\widehat{y_ij}))$$

y is the correct value, and $\widehat{y}$ is the predicted value.

These 3 are just examples, and there exist multiple different loss functions. Selecting the right function will depend on the type of problem and the context. As our data has various outcomes that are treated independently of each other, binary cross-entropy is the loss function we chose for our models.

Choosing a learning rate

Setting the learning rate of a model mainly affects its ability to make predictions. A too-small LR could make it very time consuming to find the global minimum. Setting the learning rate of a model too high can cause it to overshoot the minimum, which makes the model diverge instead of converging, never finding the minimum. Using an adaptive learning rate is a technique where you start with one learning rate, then alter the learning rate in the training process. Usually, it is used to start with a higher learning rate, then decreasing it as you get closer to the minimum. As you get closer to the minimum, you reduce the learning rate to avoid overshooting it.

4.4 Dropout

To prevent overfitting, we used a technique called dropout. Dropout is a technique in machine learning where the model simply ignores some parts of the data. Dropout is implemented as a layer, which has a parameter for how much of the data should be ignored. A NN can contain multiple dropout layers.

4.5 Embedding

The data we have contains a lot of categorical values, and have to be processed before it is sent to a neural network. One way to do this is by using Word2Vec or embedding, which is a normal method when working with natural language problems. Words are translated into vectors, and a machine learning model can train and change the values in the vector. Cheng Guo and Felix Berkhahn [13] found that by using the embedding technique on categorical variables, it reduced memory usage, sped up the neural network, and, most importantly, it led to the categorical features to gain a richer representation. The categorical variables could be mapped into vectors where the vector values could represent properties. After we had used embedding, the id of a patient was represented as a vector. The id could represent additional information about the patient instead of just being an identifier for the patient.

4.6 Long Short Term Memory Networks

Our data is based on time-events, which is hard for a normal neural network to process. One patient has multiple events, which means that previous events need to persist in a network. A Recurrent neural network(RNN) contains loops that make it possible for the network to persist information[30]. By persisting the data, the network can use data from earlier events while processing data and is the reason it performs so well on time-series data, such as in this project. LSTM is a special type of RRN, which can persist data for a longer amount of time. To further understand a LSTM network, we need to look at a LSTM unit. Each unit has three gates and a cell state, as shown in the figure below.

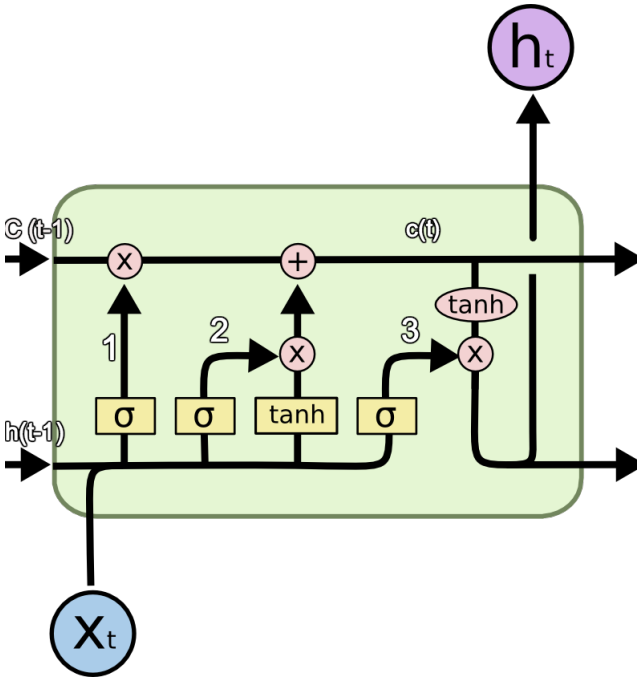

Figure 7: A LSTM unit

Source : http://colah.github.io/posts/2015-08-Understanding-LSTMs, used with permission from creator, Christopher.

Forget gate:

A forget gate(1 in the figure) looks at the previous data point and makes a decision of which parts of the previous data is relevant to keep in memory. Thus, only remembering the important data. The values sent out of the forget gate is between 0 and 1, where 1 means the data is important.

Input gate:

The input gate(2 in the figure) takes the hidden state(h(t-1)) and the current input and sends it into a sigmoid function. The output will be between 0 and 1, where 1 means that this data should be updated. At the same time, the same data goes through a tahn function, which outputs a number between -1 and 1 to help regularize the network. In the end, the output from the sigmoid function gets by the output from the tahn function. The tahn function help regularize the network, while the sigmoid function decides which of the data from the tahn function is important.

Cell state:

After the forget and input gates, the network calculates a cell state. First, the cell state gets pointwise multiplied with the output from the forget gate, which can remove data if the forget gate vector is close to 0. Then the network does pointwise addition to the input gates output data, which updates the cell state.

Output gate:

The output gate(3 in the figure) calculates the next hidden state. The output gates take the current input and the hidden state of the previous node. The output gate feeds the previous hidden state and the current input into a sigmoid function. At the same time, the network takes the new calculated cell state and feeds it to a tahn function. Lastly, the output from the tahn and sigmoid functions is multiplied to determine what information the new hidden state should be. The new hidden state and the new cell state is then sent to the next node or used as a prediction.

5 Research approach

There were several phases in this project. First, we need to get approval from Regional committees for medical and health research ethics(REC), then we need to set up the software system to be used. We had to clean the data, and finally, then the operation phase, where conducted the experiments and finally a post phase. We will go through all the stages here.

5.1 Regional committees for medical and health research ethics

Since this project uses sensitive data, the project had to be approved by Regional committees for medical and health research ethics(REC). REC is a group of committees which provides approval for medical and health research project, general and thematic research bio-banks and dispensation from professional secrecy requirement for other types of research. REC consist of seven regional committees, which consist of people of different backgrounds. The Ministry of Education appoints all committees and serve for four years.[9] We filed our REC application on 20.06.2018 and was approved on 24.09.2018. For another similar project with a time deadline, it is essential to start the REC process early.

The REC approval process required a final definition of all research questions before the project started, and the thesis could not change after REC's approval. As the thesis had to be approved by REC beforehand, the main research questions and purpose could not change after approval, which reduces the chance to introduce bias in the thesis. Publication bias is reduced as the thesis will be publicized regardless of the findings, and the thesis cant is altered to fit the findings better.

As there are strict routines when working on sensitive data, there were essential to have a secure development environment. A laptop that could be used to do data processing was ready on 14.11.2018. Lastly, the final computer to run machine learning algorithms was prepared to use on 15.07.2019. The computer had one GPU, meaning that only one machine learning process could be running simultaneously, making it a slow process. Using multiple GPU's would definitely make it possible to test more machine learning techniques and hyper-parameter tuning. The time-demanding processes of the REC application and the set up for the development

environment led to some delays for the thesis.

5.2 Data

After the project's approval and the setup of the development environment, it was time to analyze and process the data. In this step, we seek to gain a deeper understanding of the data. You have to fill or remove rows with missing data, followed by featured engineering. An overview of the data and the processing steps are written in detail in section 7.

5.3 Operation

This study was empirical in nature; we applied machine learning algorithms to the data that we cleaned, see section 5.2. The machine learning model learns from patterns regarding the patient previous medical history, which exists in the patient's medical record. In this step, validation played an important role; for each training iteration, we changed the data used to validate the model, which is described in section 3.6. Another set was left out when training the model, which is used to evaluate the model after training. The different evaluation methods are described in section 3.7. Each of the experiments will be outlined, and they follow this template:

1. A description of the models' architecture
2. An overview of hyper-parameters, batch size, loss function, and optimizer.
3. A combined overview of the results achieved in both experiments.

5.4 Post phase

One can view the overall study as a feasibility study: to find out how good machine learning is for predicting medical deterioration in patients diagnosed with diabetes. The evaluation methods used to make it possible to give an accurate statement about how well the model is predicting. The thesis can also be used to show that this approach works on other prediction or classification tasks regarding medical records.

6 Ethics

Bringing machine learning into the field of medicine has opened up a lot of possibilities. At the same time, it is important to think about negative consequences or difficult ethical questions regarding machine learning in medicine. Federico Cabitza, Raffaele Rasoini, and Gian Franco Gensini wrote an article[7] going over four unintended consequences of introducing machine learning into the health care system. The four unintended consequences they list are:

1. Physicians can over-rely on automation and technical applications in their field, which can reduce their skill in their field.

2. Relying on machine learning and automation can lead to an increased focus on the text, images, and other data. Context and other patient information can be hard or impossible to transcribe into data or may be missing from data files.

3. Intrinsic Uncertainty in Medicine

4. The black box problem in machine learning.

The first two are highly relevant to this project. Diabetes is a lifestyle disease, lifestyle data can be hard to translate into medical data, or the data is non-existing as well. The consequence of this is that the solution created in this project should only be used to identify patients with a high risk of experiencing medical deterioration, with the goal of contacting them and then use medical professionals in consulting sessions. Meaning that the skill of the professional is important, both to give advice and to detect information about the patient, which is not transcribed in the medical records. For the fourth consequence, the goal of this project is to find patients with high risk for negative consequences, so they can get a doctor's appointment to discover why and how to prevent it.

Another important factor when discussing ethics is a bias regarding gender, race, or religion. A study[16] was made to create a machine learning network to give a risk analysis of which prisoners should be released from prison. The network ended up giving black people a much higher risk of doing.

7 Data

This chapter describes the pre-processed data, the steps completed in feature engineering, and the balancing of the outcomes.

7.1 Pre-processed data structure

The data given by Haukeland contains data for around 20.000 patients. It is event-based where each event has information about one of the following:

- a patient arrived at a hospital department
- a patient left a hospital department
- a diagnosis was assigned
- a procedure was performed
- the patient did not turn up for a hospital appointment
- the patient rescheduled a hospital appointment
- the patient tested HbA1C, urine glucose or serum glucose (includes the result of the test)
- the patient changed their address
- the patient died

Below is a table of fake data that shows the structure of the original data, when it was received.

	patient_id	born	department	event_comment	event_name	event_value	order	gender	in_reference	test_value	event_time
0	9ba9d11e	1945	XHA-EYES	C123: eye-screening	diagnosis	ABC-EYES	0	1	None	NaN	2000-01-01:10:00:00
1	9ba9d11e	1945	Skin	K152: measurement	diagnosis	ABC-Skin	0	1	None	NaN	2000-01-01:11:00:00
2	9ba9d11e	1945	LAB	A122: glucose test	procedure	GT-LAB	80	1	Over referenced area	30.0	2000-01-01:12:00:00
3	9ba9d11e	1945	Emergency	Death	outcome	10	0	1	None	NaN	2000-02-01:10:00:00
4	a5bebecc	1982	XHA-EYES	C123: eye-screening	procedure	ABC-EYES	0	2	None	NaN	2000-02-03:10:00:00
5	a5bebecc	1982	LAB	A122: glucose test	procedure	GT-LAB	40	2	under referenced area	2.0	2000-02-03:10:00:00

Figure 8: Example of data

A lot of the titles in the table is self-explanatory, and will not be described. The first row, "patient_id" is a uniquely hashed identifier for each patient. "event_comments," explains the type of the given event, while "event_value," gives supplementary information. If multiple events for the same patient happened with the same timestamp, the row "order" is used to sort the events in the correct order. "test_value" contains a value of an HbA1C, urine glucose or serum glucose measurement. The field in_reference states if the test is in the standard reference area for the given test.

In the data, there were five different outcomes classified as 1, 2, 3, 4, and 10.

- outcome 1: Transpupilary laser treatment of retina, partial toe amputation and Partial amputation of a finger
- outcome 2: Intertarsal amputation
- outcome 3: Amputation of the femur and Transmetatarsal amputation
- outcome 4: leg amputation
- outcome 10: Death.

outcome	total	with other outcomes	1	2	3	4	10
outcome 1	305	47	305	4	20	20	18
outcome 2	7	5	4	7	2	3	1
outcome 3	146	68	20	2	146	38	29
outcome 4	133	62	20	3	38	133	24
outcome 10	1132	56	18	1	29	24	1132

Figure 9: Outcome Table

The table above shows the total occurrences for each outcome. It also shows how many with a given outcome also have one of the other outcomes. It would be reasonable to think that people with outcome 10 would also experience a large number of other outcomes, which lead to further investigation of the data.

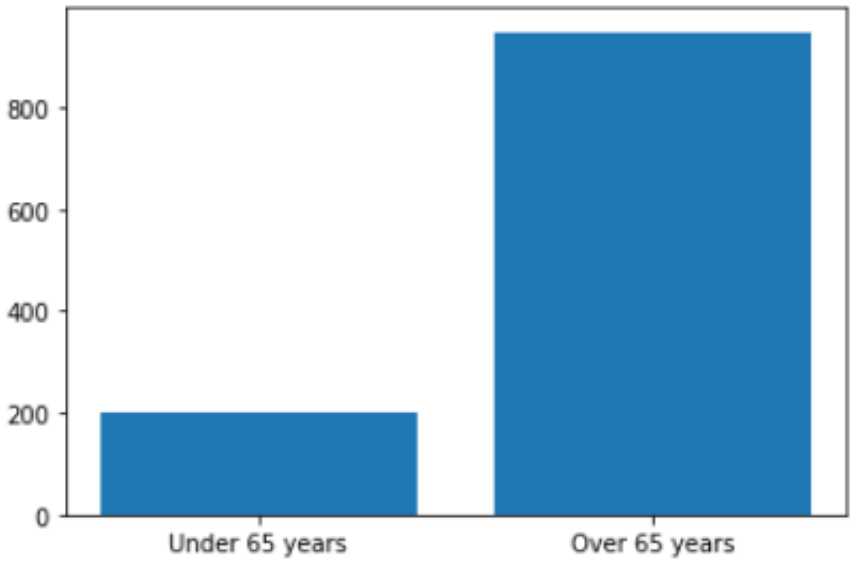

Figure 10: Outcome 10 based on age

As shown in the diagram above, most of the patients with outcome 10 is 65 years or older. This can make it harder to identify young patients with a high risk of mortality.

7.2 Data processing

Processing the data serves two purposes. First, a dataset must be in a format that can be sent into a neural network. Secondly, it can also improve the prediction capabilities of the network. The first step was to fill in for missing values in the dataset. Missing "date of birth" was filled out by copying from other events from the same patient. Gender went through the same step as the date of birth. Missing test value was replaced by 0, while in reference was filled with not measured. The "in reference" field included different spelling for the same values. The data was altered to contain one type of spelling that was used; otherwise, a neural network would look at them differently.

Feature selection

- Event_id was removed as it is unique for all events in the data, which makes it useless when learning a machine learning network to find patterns.
- Removed event_comments that occurred less than five times in the whole

dataset, which removed 14513 events. These events were removed for the same reason as the choice to remove event id from the data. Event comments are a medical code that explains a given event.

- The last step of removing events from the data was to remove patients with less than ten events, which removed 1207 patients, bringing it to a total of 19258 remaining patients.

After feature selection, the data would look like this:

	patient_id	born	department	event_comment	event_name	event_value	order	gender	in_reference	test_value	event_time
0	9ba9d11e	1945	XHA-EYES	C123: eye-screening	diagnosis	ABC-EYES	0	1	Not_measured	0	2000-01-01:10:00:00
2	9ba9d11e	1945	LAB	A122: glucose test	procedure	GT-LAB	80	1	Over referenced area	30	2000-01-01:12:00:00
3	9ba9d11e	1945	Emergency	Death	outcome	10	0	1	Not_measured	0	2000-02-01:10:00:00

Figure 11: Data after feature selection

The second row is removed as it contained an event that happened under five times, while the two last rows were deleted as the patient had too few events in total. The yellow is specifies the changes to the in_reference and test_value columns. In the test_value, Nan is replaced by 0, while the None values in the column in_reference are replaced by "Not measured."

Feature extraction

- Each event contained the date of birth but was replaced by age at the given time of the event.
- Date of the event was changed from being a single field to four different; Day, Week, Month, and Year.

After feature extraction, the data would look like this:

	patient_id	age	department	event_comment	event_name	event_value	order	gender	in_reference	test_value	year	month	day	week
0	9ba9d11e	55	XHA-EYES	C123: eye-screening	diagnosis	ABC-EYES	0	1	Not_measured	0	2000	1	1	1
2	9ba9d11e	55	LAB	A122: glucose test	procedure	GT-LAB	80	1	Over referenced area	30	2000	1	1	1
3	9ba9d11e	55	Emergency	Death	outcome	10	0	1	Not_measured	0	2000	1	1	1

Figure 12: data after feature extraction

As seen in the yellow columns, birth is replaced by the age at the given time of the event. Event_time is split into four different columns, which is; year, month, day,

week.

Before starting to train the neural network, variables had to be classified as categorical or continuous variables. This step was completed to select which variables should be sent through an embedding layer in the network. The following variables were classified as categorical: department, event comment, in reference, event name, event value, patient id, year, month, day.

7.2.1 Data balancing

Since most of the patients did not experience any of the cases, balancing had to be done. As explained in chapter 9, mortality was the only case predicted on in this study, meaning that only balancing for that outcome was completed. Before balancing, the training data contained 14501 patients who did not experience outcome 10 and 1132, who did, as shown below.

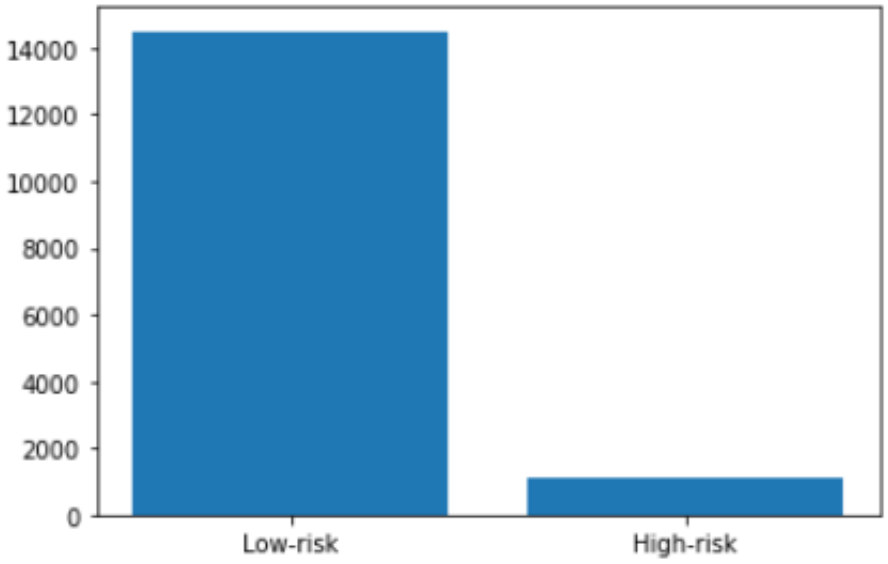

Figure 13: unbalanced data

Two techniques were used to counter the unbalance in the data. First, the patients with outcome 10 were copied four times. The copied patients were given a new unique patient id so that each copied patient is treated individually by the model. It is optimal to have an equal ratio between the outcomes, although if a few patients are copied too many times, a model may overfit on those cases. After this

balancing was completed, the number of patients with outcome 10 was now 4756. The new balance is shown in the graph below.

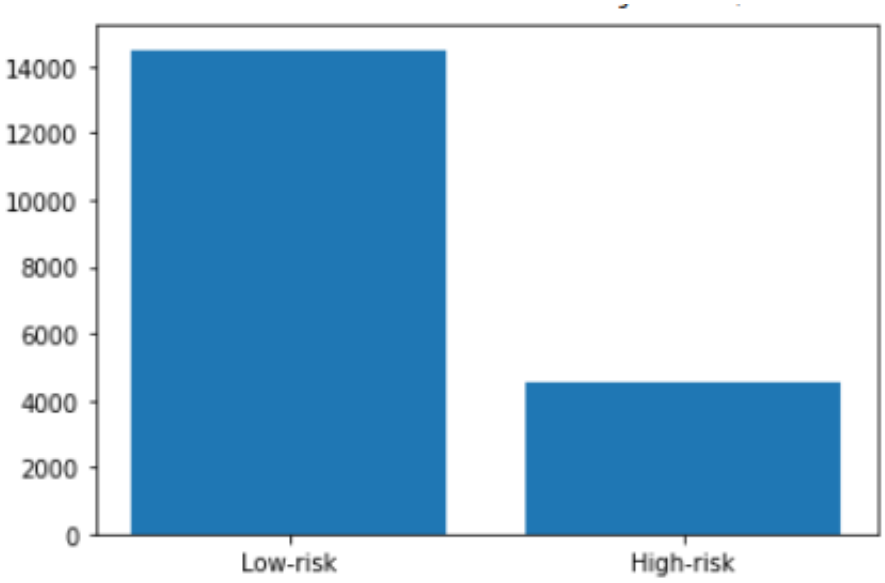

Figure 14: Balanced data

Lastly, a function from Keras, see chapter 8, called class_weight, was used when training the model. The function is used to let different classes affect the loss function differently. In our models, outcome 10 affects the loss more than not having outcome 10.

Another technique that could have been used to avoid overfitting of the copied patients is undersampling, which is when cases of the superior label are removed to balance the dataset.

8 Tools and implementation

This section gives a brief overview of the tools used in the project and the implementation of the solution. It is included in the study for others to be able to reproduce the study using the same approach, as the code written in the study is not public due to using sensitive data. The chapter can be skipped when reading the study.

The implementation of the experiments used in the project is written in the programming language Python. It also uses multiple supplementary libraries such as; Pandas, Numpy, FastAi, Scikit-learn, and Keras. The most used libraries are explained below.

TensorFlow

Tensorflow[1] is an open-source platform for machine learning. TensorFlow was developed by researchers from the Google Brain team within Google's Machine Intelligence Research organization. TensorFlow was not used directly but is used by Keras.

Keras

Keras[8] is a high-level API for writing Neural Networks which can run on top of TensorFlow. Keras focuses on user-friendliness, modularity, and easy extensibility. It simplified the process of creating a neural network, which made it possible to spend more time on data processing and the neural networks architecture and feature engineering in this project. All models created in the study are created using Keras. An example of how a neural network is implemented is shown in the second code snippet below.

Scikit-learn

Scikit-learn[26] is another library to create machine learning models. The thesis uses a small selection of functions used from the library, to avoid spending time on implementing functions which can easily be imported from scikit-learn. We used functions to create the confusion matrices and calculate the AUROC score, which can be seen in section 9.

FastAi

FastAi is yet another library to create machine learning models. The thesis uses one function from the library; add_datepart, which takes in a date and adds extra columns relevant to the date.

8.1 Implementation

Most of the implemented code can be found at `https://github.com/Andreass2/medical-deterioration-of-diabetes`. The structure of the git repository is split into the following files:

Utils.py

It contains helper functions to improve the readability of the code. This file includes functions for balancing the data, fill in missing values, and splitting the data into multiple datasets.

CleanData.ipynb

This is where most of the data cleaning and feature engineering is completed. A lot of the helper functions in Utils.py is used here.

PredictOutcome10_base.ipynb

This is the file where the first experiment is implemented. The file contains the implementation of the neural network used, and some structural changes to the data to be able to feed it to the network.

PredictOutcome10_with_time.ipynb

This is a similar file to PredictOutcome10_base.ipynb, but contains the implementation of experiment 2. This is also the file that removes events in a specified time before a mortality event.

A standard LSTM network requires each data-point to have the same number of rows. In our data, each patient had a different number of events. This was the most complex problem during the implementation of the experiments, and we had multiple discussions on how to solve this problem.

We created a solution using python's "None" value, which can be used to represent variable length. The input size for categorical values had to be (1, None 1), and the input size for the continuous variables had to be (1, None, 3). The None value represents the length of the medical records for a given patient. The last value represents the number of columns given to the layer. As all categorical variables were sent into a different layer, the input layer was fed one column at the time. There were three different continuous values, meaning the input layer was fed three columns.

To feed the data into the network during training, we had to use fit_generator from Keras, with a function we implemented to feed the data in the correct structure to the network. Below you can see the code for the training data generator we implemented.

```
def train_generator(indexes=[]):
    #step 1
    if(len(indexes) == 0):
        indexes = [i for i in range(len(X_train)-1)]
    shuffle(indexes)
    i = 0
    while True:
        if(i >= len(indexes)):
            print("reset train index")
            i = 0
        df = X_train[indexes[i]]
        length = len(df)
        inputs = []
        #step 2
        for cat in cat_names:
            inputs.append(np.array(df[cat]).reshape(1,length,1))
```

```
        #step 3
        inputs.append(np.array(np.array(
            df[['age','gender','test_value']])
            .reshape(1,length,3)))
        i = i+1
        #step 4
        yield inputs, y_train[indexes[i-1]]['10'].values
```

This function required four step.

1. The first thing that has to be done is to find the correct patient.
2. For each categorical value, the data is reshaped into (1, number of events, 1)
3. Then we add the continuous variables in the shape of 1, number of events, number of variables)
4. send the data to the network.

Below is simplified code(full code in GitHub) for the implementation of the second experiment.

```
inputs = []
layers = []
# Step 1
for cat in cat_names:
    i = Input(batch_shape=(1,None,1), name=cat)
    e = Embedding(vocab, embedding_size, input_shape=(1,None,1))(i)
    r = Reshape((-1,embedding_size))(e)
    inputs.append(i)
    layers.append(r)
#step 2
i = Input(batch_shape=(1,None,3,))
layers.append(i)
```

```
# Step 3
merged = Concatenate()(layers)
norm = BatchNormalization()(merged)
f = LSTM(hidden_size, dropout=0.5,recurrent_regularizer='l2',
                        return_sequences=True, stateful=False,
                        use_bias=True)(norm)
x = LSTM(hidden_size, dropout=0.5, stateful=False,
        recurrent_regularizer='l2',
        use_bias=True)(f)
out = Dense(1, activation='sigmoid')(x)
inputs.append(i)
# Step 4
model = Model(inputs=inputs, outputs=out)
```

As seen in the code above, this snippet can also be explained stepwise.

1. The first step is to create the input, embedding, and a reshape layer for each categorical value.

2. Creates the input layer for the continuous variables.

3. The input layers are concatenated so that a single layer can process all data at the same time. This is also the step where the LSTM layers are created.

4. Creates the model.

9 Experiments

This section goes through the two different approaches used in the project to predict the data set. It was decided to use multiple methods to compare different approaches. As the goal of the experiments varies, the data in each experiment also varies. Experiment 2 excludes 22 patients who were included in experiment 1.

Both experiences use the same neural network architecture, as shown in the figure below.

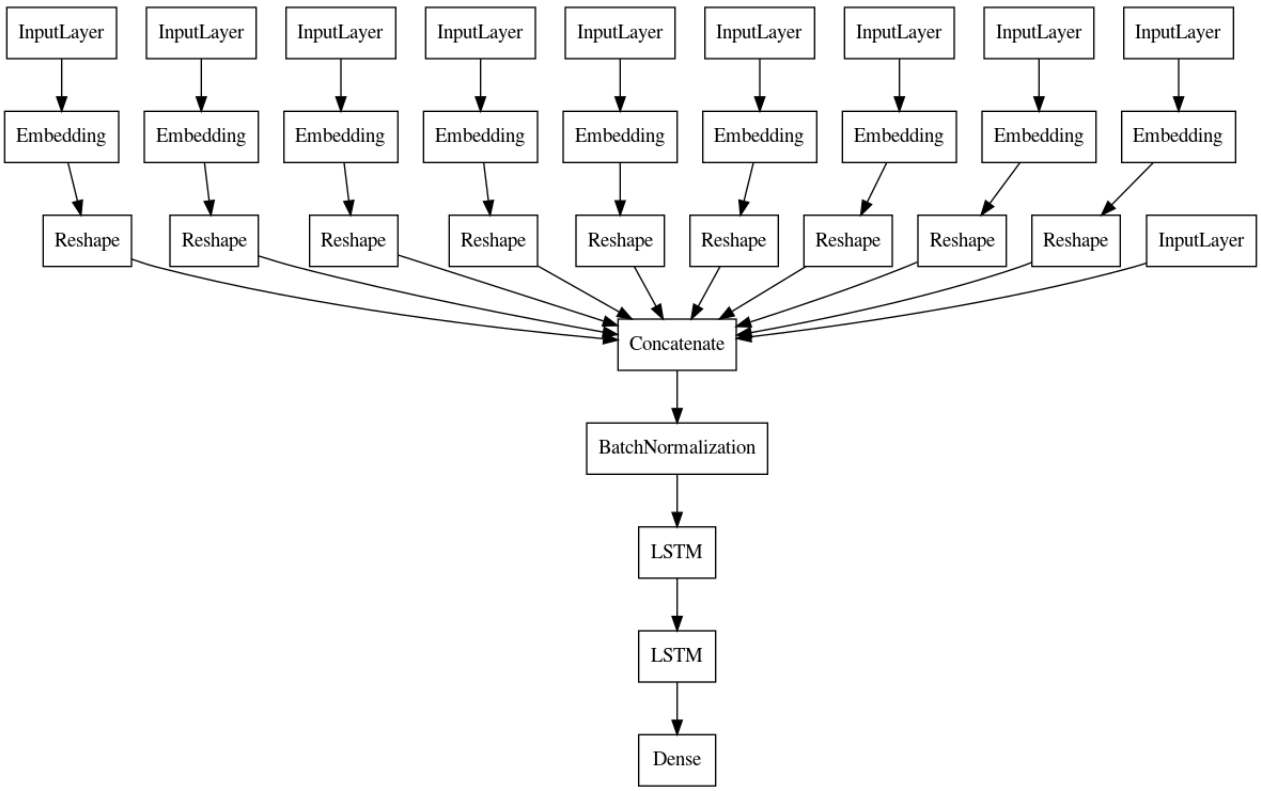

Figure 15: The neural network architecture

The network contains one input layer for each of the categorical values, which makes it possible to adjust the size of the embedding matrix individually. It is also a single input layer for all continuous variables, which includes age, gender, and test_value. After the embedding layers, the embedding matrix is sent to a reshape layer to represent the embedded data in the same structure as the continuous data. The concatenate layer combines the data from all layers. The batch normalization layer standardizes the input. Two LSTM layers are used, which both have the size of 40 neurons and the same parameters, which are:

- The first LSTM layer has return_sequences set to true, which is needed to keep the data in a format the next LSTM layer can process.
- Dropout is set to 0.5
- Recurrent_regularizer is set to use l2
- use_bias is set to True

The last layer is a Dense layer, which uses sigmoid as an activation function. The dense layer outputs a value between 0 and 1, which is the prediction.

9.1 Experiment 1

The first experiment's goal is to give a proof of concept that the dataset contains detailed enough data for predictions to give value. Mortality is the only outcome included, as it is the most common negative outcome. The network predicts a binary outcome of 0 (no risk of mortality) and 1 (risk of mortality).

Parameters

- **Learning rate:** 1e-4
- **Learning rate decay:** 1e-5
- **Optimizer:** Stochastic gradient descent, 0.8 momentum, nesterov set to true
- **Loss function:** binary crossentropy
- **batch size:** 512
- **class weight:** 0:1, 1:1.2

9.2 Experiment 2

This experiment extends Experiment 1. Like the previous experiment, the model only predicts mortality but excludes the last four weeks before mortality happened. The goal of this approach is to represent the problem more realistically. The network should be able to predict the risk of mortality as early as possible. If the network is only able to predict if a patient dies the same day, it does not add value. This

experiment excluded patients who only had events in the four weeks before mortality happened.

Parameters

- **Learning rate:** 1e-4
- **Learning rate decay:** 1e-5
- **Optimizer:** Stochastic gradient descent, 0.8 momentum, nesterov set to true
- **Loss function:** binary crossentropy
- **batch size:** 512
- **class weight:** 0:1, 1:1.2

Results

After implementation, the neural networks had to make predictions on data that had not been included in training or validation. To find the result, we use multiple metrics, as explained in section 3.7. We start with the confusion matrices for both experiments.

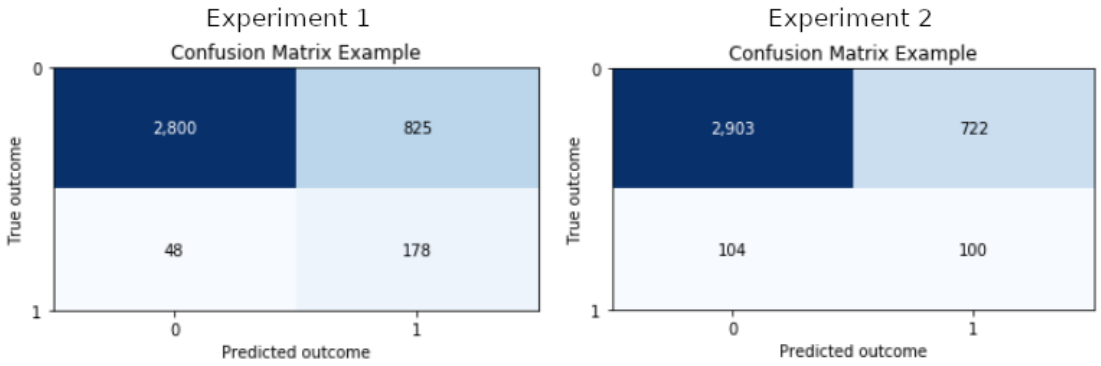

Figure 16: Confusion matrices

Above, it is shown that the model from experiment 1 predicted 2800 correctly and 825 wrong for class 0. While for class 1, it predicted 178 correctly and 48 wrong. The second model from experiment 2 predicted 2903 correctly and 722 wrong for class 0. While for class 1, it predicted 100 successfully and 104 wrong. The first experiment has more accurate predictions than the second.

Further, we need to look at precision and recall, which can tell how many instances of high-risk patients the models can identify and how reliable a given prediction is.

	Precision 0	Recall 0	Precision 1	Recall 1
Experiment 1	0.98	0.77	0.18	0.79
Experiment 2	0.97	0.80	0.12	0.49

Experiment 1

Out of all cases of class 0, the model can identify 77% of them. When the model predicts outcome 0, it is a 98% chance that the model is correct. As for class 1, the model can identify 79% of all cases. When the model predicts outcome 1, it is a 18% chance that the prediction is correct. The precision and recall for class 0 is higher than for class 1, this is expected as we have much more data about class 0.

Experiment 2

Out of all cases of class 0, the model can identify 80% of all cases. When the model predicts outcome 0, it is a 97% chance that the model is correct. As for class 1, the model can identify 49% of all cases. When the model predicts outcome 1, it is a 12% chance that the prediction is correct. Again predictions on class 0 is more accurate than class 1.

The last metric we used is AUROC.

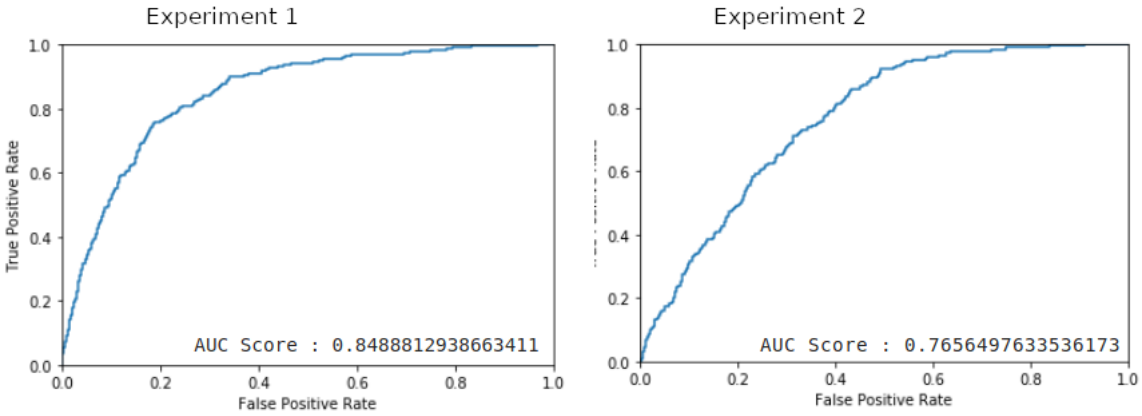

Figure 17: AUROC graphs

The first model got an auc score of 0.85, while the second model got a score of 0.76.

As there was an imbalance between age in patients with the mortality outcome, we decided to check how well the models perform for patients over and under 65 years.

	Under 65	Over 65
Experiment 1	35/39	143/187
Experiment 2	15/37	85/167

' For patients over the age of 65, model 1 had an accuracy of 76.5%, while model 2 had an accuracy of 50.9% When we look at patients under the age of 65. Model 1 predicted correct for 89.7% of those patients, while model 2 predicted correctly on 40.5%.

10 Interpretation of results

In this chapter, we take a closer look at the results. Patients without outcome 10 will be referred to as class 0. Patients with outcome 10 will be referred to as class 1. The first model achieved a better result, which was expected. We removed the last four weeks from all patients with class 1 in the second experiment, which means there is less detailed data for each patient. The main focus will be on the second experiment, as it predicts the risk of mortality at an earlier stage.

Recall would be the most crucial metric when interpreting the results. It explicitly tells how many instances of negative outcomes the networks can identify. The first experiment was able to identify 79% of all mortality cases, while the final result for experiment 2 is that the network was able to identify 49%.

Precision is lower than recall in both experiments. When the models predict a high-risk patient, model 1 has an 18% chance of being correct, while model 2 has a 12% chance of being right. This means Haukeland could potentially use time on the wrong patients.

The AUROC scores for both models are well above 0.5, meaning both models predict much more accurately than random. If we use the rule of thumb from section 3.7, the first model can be classified as a good result, while the second model can be classified as fair results.

The results can be used to answer research question 1 directly; "how well can medical record data be used in machine learning to predict medical deterioration of patients diagnosed with diabetes?". Given that the results are written in the percentage of how many negative outcomes the network can identify and how many high-risk predictions are believable predictions, it gives a clear answer. As experiment 1 can identify 79% of high-risk patients, it is clear machine learning can be a useful technique when working on medical records. In experiment 2, the network achieved a lower score but still manages to identify almost half of the high-risk patients.

The second research question: "By removing the latest events for a patient, how early can the risk of medical deterioration by identified?" can be answered by comparing the results from experiments 1 and 2. As expected, the result from the first experiment performs more accurately than the result of the second experiment.

By removing the last four months before mortality, recall is reduced from 79% to 49%, while precision is reduced from 18% to 12%. The decrease in the results from experiment 1 to experiment 2 is expected. The most critical events for predicting the risk for mortality will be closer in time to the mortality event. The AUROC score classifies the model's results as fair. This means that the network is still able to detect patterns at an earlier stage, at the cost of precision and recall.

11 Threat to validity

Validity explains whether or not a study is trustworthy and meaningful. When speaking of the threat to validity in research, there exist mainly two subcategories, internal and external. Internal validity concerns the structure of a study, how well it is conducted. It is often linked to how confident one can be with the findings when doing research. There are multiple threats to the internal validity in this thesis, for example, the assumptions made about the medical records. As there may exist outdated procedures in the dataset, it decreases the trustworthiness of the correlation between procedures and the outcomes for a patient. Measures were taken to improve internal validity. When splitting the dataset into a train, validation, and test set, patients were randomly selected for which set they should be in, although the initial ratio of the number of outcomes was kept in all sets.

External validity is about how well the findings or outcome of a study can be applied to other cases; in other words, how well the outcome can generalize. Generalization is also a big topic in machine learning, as the goal of most machine learning models is to be able to make predictions on new unseen data. Threats to the external validity in this thesis may be:

- Selection bias: As mentioned in the introduction, there were over 200 000 registered patients with diabetes in Norway in 2016. The dataset in this project has around 20 000 patients. This means that it cant be certain that all demographic groups are included in this study. This can also affect internal validity.

- Outliers and noise: If the dataset contains a lot of noise or outliers, the neural network may "find" patterns in those, which will probably make the network generalize less. Common methods in machine learning are to remove outliers from the data and to use dropout, which makes the network generalize better.

- A patient may have other diagnoses and receive treatments from the healthcare system, making the network finding rare patterns that may limit generalization.

Multiple steps to improve external validity can be taken. Replicating the study with different data samples, and then using meta-analysis can improve both internal and

external validity. The thesis is written in a way that another study should be able to use the same approach on different data. In the following chapter, exclusion/inclusion criteria are clearly defined, so that others can process the data in the same way this thesis does. Most of the code is available, making it possible to replicate the project without problems. To see if the network is able to generalize and predict new unseen data, a test set of patients was excluded from training the neural network. The findings of the thesis are based on how well the network is able to make predictions on the test set, which directly tests the external validity.

We had to make multiple assumptions in this project due to the data and time limitations on the project. These assumptions will increase the threat of validity. The first assumption is that there are no changes in procedures and routines over time. For example, data from 1970 is processed the same way as data from 2015. The next assumption is that no other data is needed to predict medical deterioration. The patient's lifestyle is essential regarding medical deterioration for diabetes, but the data given from Haukeland does not contain such data, so it is assumed that this kind of data can be disregarded. In the thesis, we assume that consultant sessions with a patient will be able to prevent medical deterioration, but to be able to state this, testing has to be completed. It is relevant to know that these three assumptions are not realistic for the real world but only made for the project to be feasible. The three different types of diabetes are not processed differently, so it is assumed that to know which type of diabetes the patient has is enough.

The thesis uses real data in a structure used by the healthcare system today. This is exciting, as there was much doubt regarding the approval from REC. When creating fake data, the generated data may not represent real data in the correct way. In fake data, ratios between gender, age, labels, and event codes could have been wrong, making it easier or harder for a network to make predictions on the fake data. It would be easier to create data that would fit the thesis, introducing more bias in the findings.

12 Related work

MIMIC-III

MIMIC-III is an extensive database that includes data for 38,597 patients admitted to critical care units in USA[17]. The data is collected between 2001 and 2012 and has been completely anonymized. The data has a different structure as the data we received from Haukeland. If we started working with the MIMIC-III dataset, we would have to redo all data processing when working with our data. Instead, I have worked with this dataset in another project to prepare for this thesis, which can be found on `https://github.com/Andreass2/HospitalMortalityRate`.

As the data is open, it makes it possible to reproduce and improve studies completed on the dataset.

Predicting Mortality in Diabetic ICU Patients Using Machine Learning and Severity Indices

"Predicting Mortality in Diabetic ICU Patients Using Machine Learning and Severity Indices"[3] is a study where they used the MIMIC-III database to predict mortality for diabetic patients in the ICU. The study achieved an AUROC score of 0.787 by combining HbA1c, mean glucose during the stay, diagnoses upon admission, age, and type of admission. This study predicts mortality at a given hospital stay, while our goal is to predict as early as possible. The study is still important to us, as it states the importance of the data variables used when predicting mortality for diabetic patients.

Deep learning algorithm predicts diabetic retinopathy progression in individual patients

"Deep learning algorithm predicts diabetic retinopathy progression in individual patients,"[4] is a study in which they created a neural network to make predictions on diabetic retinopathy progression for individual patients. There are multiple differences between this study and ours;

- Treatment of retina is included in our data as outcome 1, although we would also be able to predict different outcomes at the same time.
- This study uses eye-screenings to make predictions, while we want to see if similar predictions can be completed while using medical records.

Predicting Diabetes Mellitus With Machine Learning Techniques

"Predicting Diabetes Mellitus With Machine Learning Techniques"[33], is a study that predicted if a patient had diabetes. They used a machine learning model called "random forest" and achieved an accuracy of 0.8084. They found that fasting blood sugar, age, and weight is important attributes when predicting diabetes. In the study, they state: "According to consulting relevant information, we know there are three indicators to determine the diabetes mellitus, which are fasting blood glucose, random blood glucose, and blood glucose tolerance."(Zou et al. 2015). This study shows the importance of blood glucose levels has in diabetes.

Type 2 diabetes risk forecasting from EMR data using machine learning

Mani et al. [19] used different machine learning techniques to forecast the risk of a patient developing diabetes. Their study combined demographical, clinical, and lab data in predictions and manage to achieve an AUROC score of 0.8 when assessing the risk of developing diabetes type 2 365 days before the diagnosis was given. The data they used had a significant focus on lab values and less on medical records. Almost all of their patients had measurements of glucose, which is highly linked to diabetes. The data available in our study includes three different test values and a lot of events and patients that did not have measurements.

Haukeland University Hospital

Haukeland University Hospital, in collaboration with the University of Bergen, has established Mohn Medical Imaging and Visualization Centre (MMIV), which has multiple publications where machine learning is included. Most of their work is about visualizing and processing medical images, and a complete list of their publications can be found at `https://mmiv.no/publications/` Our thesis is the first machine learning study using real medical records at Haukeland.

13 Discussion

This has been the first collaboration between Haukeland, Western Norway University of Applied Sciences, and the University of Bergen while working on sensitive data. We learned the importance of an early start regarding applying to REC to approve the project. It has the possibility to add significant delays to a similar project. While waiting for approval from REC, we would recommend setting up the development environment. The last recommendation would be to have multiple GPU's as it would speed up the practical part of the thesis significantly. We could have run several instances of training at the same time, meaning we could have tested a lot more techniques and hyper-parameters for the models. If we had multiple GPUs, it would have made it possible to work on experiments 1, 2, and 3 at the same time, meaning experiment 3 would have been included in the study. The plan for experiment 3 is explained in 15.2. As our predictions only include mortality, we cannot make statements about medical deterioration, but only in predicting the risk of mortality.

In discussion with Haukeland, we agreed that the most crucial part of the thesis was to be able to identify the high-risk patients, which is shown as the recall score. We also decided that the misclassification of low-risk patients is accepted, which is the reason we got a lower precision score. In the early discussions of the thesis, we discussed a solution for preventing Haukeland to spend, which will be explained in section 15.1.

The results could be improved if more data could be gathered, which could be done in multiple ways. The medical records do not include data from the patient's general practitioner, which could include measurements of blood glucose levels in the blood, depression, and more. Further, there exist devices that continuously measure a person's blood glucose levels. If that data could have been collected and included in the thesis, by also using this data, we could collect more data regarding patients. This would be part of the standardization of data. There is much data that does not exist in the medical records, which could be used to better predict medical deterioration, such as lifestyle, diet, or psychological information. For example, a patient with depression has a 37% increased risk of developing diabetes[18]. Much information that could improve prediction methods of patients is stored, but in

different systems. By having a universal standardization of data stored on different systems, one could create an API that has access to data from multiple systems. Then an application can get access to data from numerous sources with the same form.

Fast Healthcare Interoperability Resources(FHIR) is a standardization for medical data, which is fully integrated with mobile applications, cloud applications, and EHR-based data sharing and server communication in large institutional healthcare providers[15]. FHIR is collaborating with multiple of Norway's healthcare districts, which could mean that standardization for medical data is on the way.

Having shared standardized data can make it easier for medical specialists and IT-applications to get all the data it needs, in the same format. It can increase the quality and readability of the data, and make it easier to co-operate between different systems. By using the same standard, it would also decrease the amount of data processing needed to use machine learning.

The application to REC forced us to define the research questions before the study. This gives the possibility to check that the research questions are not changed afterward to fit the data, or research questions are added or removed. This is different from the practice in software engineering, in general, today since one does not need to provide any material such as research questions before the study. I discussed this issue with my supervisor, and he said that some communities, such as the ESEM/ISERN community, have discussed introducing the need of having pre-studies where the research questions will be stated before the real study will be done.

Another critical issue is open data, something that can benefit research a lot. This is discussed in many areas, and we can see more and more open data. Sometimes, companies keep data for themselves for business reasons. In our case, data is protected because of the privacy of people. However, it would be good to find a way to anonymous the data. However, this might not be very easy if only a few within a country have the same illness, such as Galactosemia, in which around 30 people in Norway is diagnosed with with[12].

Haukeland is in possession of eye screenings for patients with diabetes, which we do not currently have permission to use. A possibility would be to use this study as a guide to training another neural network to predict diabetic retinopathy using eye screenings. That network could be combined with our network to output a single prediction of the risk regarding diabetic retinopathy progression, which could be more accurate than any of the single networks.

After improving the accuracy of the model, it could be possible to create a new neural network using reinforcement learning. It could add new events into the data, then make predictions to see the new risk for the patient. If the new risk is lower, the agent receives a reward, or if the risk is higher, the agent will receive a punishment. That way, the network will be able to give recommendations of events and actions which could decrease the risk of medical deterioration.

14 Conclusion

We have shown that machine learning can be used to predict mortality for patients using medical records. The study required multiple stages of data-cleaning, filling in missing values, feature engineering, and balancing the data based on the outcome. Two experiments were conducted. In the first experiment, a network was created that was able to identify 79% of high-risk patients. In experiment 2, the network achieved a lower score but still manages to identify almost half of the high-risk patients. The precision score was lower than recall, but we agreed beforehand that the most crucial metric would be the recall metric.

There exist multiple ways to collect more data, which could make our predictions even more accurate. If the standardization of FHIR is completed, which is a standardization for collecting medical data, much more data will be available for replicating this work with more data. We strongly believe that with more data, medical deterioration of diabetes patients may be predictable at an earlier stage, avoid many amputations, eye operation, or even death.

15 Further work

Here we will explain how further work can improve our current implementations and how to add a new experiment.

15.1 Improving the completed experiments

The results achieved in the study is not accurate enough to implement the solution at the hospital. More data analysis and feature engineering are needed to improve the results. Some recommendations would be:

- The patients excluded in experiment 2, should be excluded in experiment 1 as well, which will lead to a better comparison between the models.
- Gender should be processed as a categorical variable by the neural network.
- More stratified splitting in the train, validation, and test set. When splitting the dataset into three different sets, the ratio between outcomes, gender, and age should be as equal as possible in each set.
- When balancing the dataset, it should also balance age and gender, not only be focused on outcomes.
- Implement Grid Search to improve the hyper-parameters of the model further. The training of the model took around 5 hours, so grid search would be time demanding and was down-prioritized in the study.
- Experiment 2 should be tested by removing different lengths of time.

Further, neural networks are data-dependent. If more data could be gathered and used to train the networks, it would most likely be able to affect the results significantly. The data used in the project contained data about 20.000 patients, while there are 206.795 registered medical patients with diabetes in Norway. Potentially, data from these patients could be included, enlarging the size of the data by 1000%. Also mentioned earlier in this thesis is to implement standardization of data used across hospitals and healthcare services. By utilizing a standardization, it would be easier to not only gather data from more patients but also get more detailed data about each patient.

If Haukeland implements this solution into their daily routines, a complete pipeline has to be created. The pipeline will select patients and make predictions on them. Then the high-risk patients will run through a system that checks if the patients are already in dialog with Haukeland or other healthcare services. This will reduce the resources used on the wrong patients.

15.2 Finish the last experiment

The original goal of our research was to give predictions on all outcomes, not only mortality. However, since it was hard to obtain accurate predictions due to lack of data, we focus in this work on mortality since its the most common outcome, and the idea was to show that machine learning can be used. Here we will discuss how this can be potentially solved if the work will be continued.

First of all, the dataset has seven occurrences of outcome 2, which is too few for a machine learning network to be able to give generalized predictions. Outcome 2 should be left out, and the model should predict the outcomes 1, 3, 4, and 10.

Before training a new model, more data processing has to be completed. A patient can have multiple outcomes, which can be transferred into multiple instances. The timelines below display a patient's original data(timeline1) and replaces it with three new data instances(timeline 2, 3, and 5), where each can be used to predict different outcomes.

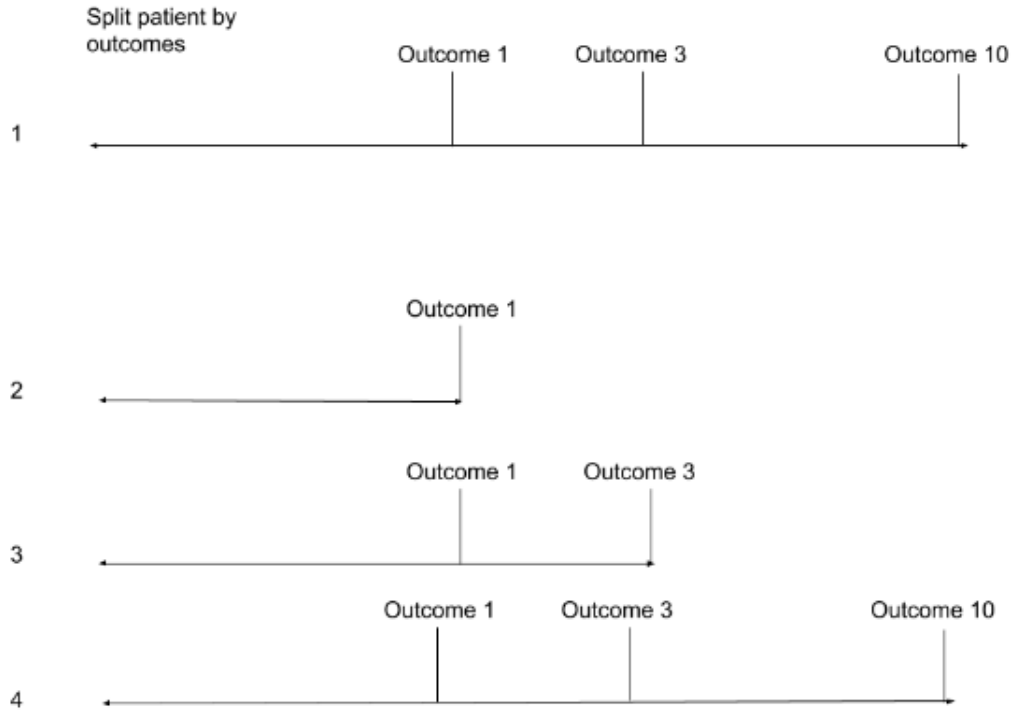

Figure 18: Splitting a patient into multiple instances

Timeline 1 in the graph above represents a timeline for a patient in the dataset. The patient has three different outcomes, which means we can split the data into three "different" patients. Timeline 2 contains events up until the event the patient experiences outcome 1, which can be used to predict that outcome. This process should be repeated for each outcome for each patient.

If a patient experiences the same outcome multiple times, it can be used to generate more data, as shown in the graph below.

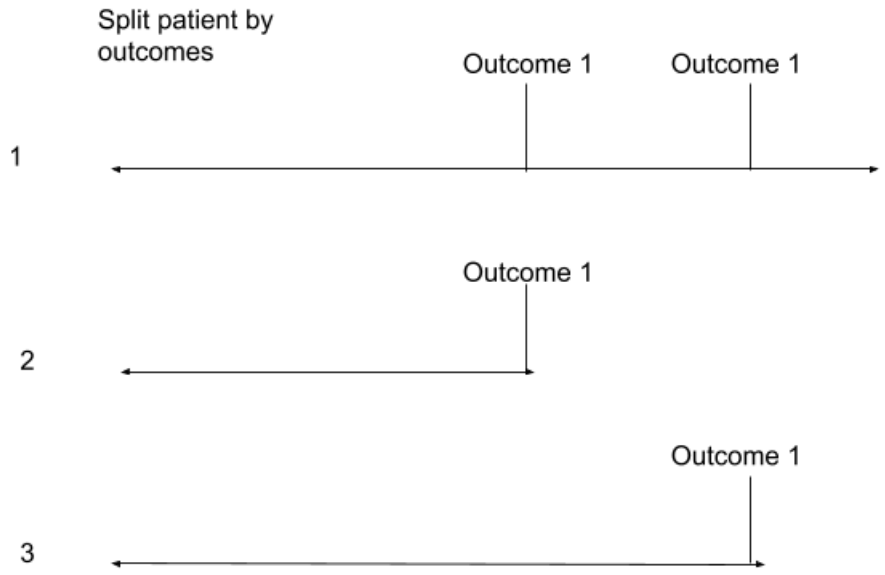

Figure 19: Creating more data from one patient

By doing these two steps, more data is generated, which can improve the result of a neural network. Another benefit is that the newly generated data only contains patients with a negative outcome, helping to balance the dataset.

The last step will be to either create a neural network able to predict all outcomes with multi-labeling the outcomes or create one neural network for each outcome, then combining the results afterward.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Helse Norge Amputasjon. Amputasjoner blant pasienter med diabetes. `https://helsenorge.no/kvalitet-seksjon/Sider/Kvalitetsindikatorer-rapporter.aspx?kiid=Amputasjoner_blant_diabetespasienter`, 2018. [Online; accessed 06-February-2018].

[3] R. S. Anand, P. Stey, S. Jain, D. R. Biron, H. Bhatt, K. Monteiro, E. Feller, M. L. Ranney, I. N. Sarkar, and E. S. Chen. Predicting mortality in diabetic icu patients using machine learning and severity indices. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2017:310–319, May 2018. 29888089[pmid].

[4] F. Arcadu, F. Benmansour, A. Maunz, J. Willis, Z. Haskova, and M. Prunotto. Deep learning algorithm predicts diabetic retinopathy progression in individual patients. *npj Digital Medicine*, 2(1):92, 2019.

[5] D. Ardila, A. P. Kiraly, S. Bharadwaj, B. Choi, J. J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado, D. P. Naidich, and S. Shetty. End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. *Nature Medicine*, 06 2019.

[6] American Diabetes Association. Learn the Genetics of Diabetes. `https://www.diabetes.org/diabetes/genetics-diabetes`. [Online; accessed 27-November-2019].

[7] F. Cabitza, R. Rasoini, and G. F. Gensini. Unintended Consequences of Machine Learning in Medicine. *JAMA*, 318(6):517–518, 08 2017.

[8] F. çois Chollet et al. Keras. https://keras.io, 2015.

[9] Regional committees for medical and health research ethics. REK - Regionale komiteer for medisinsk og helsefaglig forskningsetikk. https://helseforskning.etikkom.no. [Accessed 19-September-2019].

[10] M. Copeland. What's the Difference Between Artificial Intelligence, Machine Learning, and Deep Learning? https://arxiv.org/abs/1703.07771, 2016. [Online; accessed 16-February-2018].

[11] J. M. Font and T. Mahlmann. Dota 2 bot competition. *IEEE Transactions on Games*, 11(3):285–289, Sep. 2019.

[12] Senter for sjeldne diagnoser. Galaktosemi. https://sjeldnediagnoser.no/home/sjeldnediagnoser/Galaktosemi/8672, 2017. [Online; accessed 15-January-2020].

[13] C. Guo and F. Berkhahn. Entity embeddings of categorical variables. *CoRR*, abs/1604.06737, 2016.

[14] Building Better Healthcare. Comment: Health networks - delivering the future of healthcare. https://www.buildingbetterhealthcare.co.uk/technical/article_page/Comment_Health_networks__delivering_the_future_of_healthcare/94931. [Online; accessed 02-January-2020].

[15] HL7. Summary - FHIR v4.0. https://www.hl7.org/fhir/summary.html. [Accessed 19-September-2019].

[16] S. Mattu J. Angwin, J. Larson and ProPublica L.Kirchner. Machine bias. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, 2016. [Accessed 13-January-2020].

[17] A. E. W. Johnson, T. J. Pollard, L. Shen, L. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3(1):160035, 2016.

[18] M. J. Knol, J. W. R. Twisk, A. T. F. Beekman, R. J. Heine, F. J. Snoek, and F. Pouwer. Depression as a risk factor for the onset of type 2 diabetes mellitus. a meta-analysis. *Diabetologia*, 49(5):837, Mar 2006.

[19] S. Mani, Y. Chen, T. Elasy, W. Clayton, and J. Denny. Type 2 diabetes risk forecasting from emr data using machine learning. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2012:606–615, 2012. 23304333[pmid].

[20] Z. Obermeyer and E. J. Emanuel. Predicting the Future — Big Data, Machine Learning, and Clinical Medicine. `https://www.nejm.org/doi/full/10.1056/NEJMp1606181`, 2016. [Online; accessed 08-August-2019].

[21] National Institue of Diabetes, Digestive, and Kidney Diseases. Managing Diabetes. `https://www.niddk.nih.gov/health-information/diabetes/overview/managing-diabetes`, 2018. [Online; accessed 27-November-2019].

[22] National Institue of Diabetes, Digestive, and Kidney Diseases. Preventing Diabetes Problems. `https://www.niddk.nih.gov/health-information/diabetes/overview/preventing-problems`, 2018. [Online; accessed 06-February-2018].

[23] National Institue of Diabetes, Digestive, and Kidney Diseases. Symptoms & Causes of Diabetes. `https://www.niddk.nih.gov/health-information/diabetes/overview/symptoms-causes`, 2018. [Online; accessed 27-November-2019].

[24] World Health Organization. Global report on diabetes. Technical report, World Health Organization, 2016.

[25] C. O. Osborn. Type 1 and type 2 diabetes: What's the difference? `https://www.healthline.com/health/difference-between-type-1-and-type-2-diabetes`, 2017. [Online; accessed 27-November-2019].

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[27] D. Reinsel, J. Gantz, and J. Rydning. The Digitization of the World from Edge to Core. `https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf`, 2018. [Online; accessed 08-August-2019].

[28] The Darwin Web Server. The area under an ROC Curve. `http://gim.unmc.edu/dxtests/roc3.htm`, 2016. [Online; accessed 18-Desember-2019].

[29] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T.and Lillicrap, K. Simonyan, and D. Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[30] Ralf C. Staudemeyer and Eric Rothstein Morris. Understanding lstm – a tutorial into long short-term memory recurrent neural networks, 2019.

[31] M. Sørensen, F. Arneberg, T.M. Line, and T.J. Berg. Cost of diabetes in norway 2011. *Diabetes Research and Clinical Practice*, 122:124 – 132, 2016.

[32] Acute Care Testing. ROC curves - what are they and how are they used? `https://acutecaretesting.org/en/articles/roc-curves-what-are-they-and-how-are-they-used`. [Online; accessed 18-Desember-2019].

[33] Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, and H. Tang. Predicting diabetes mellitus with machine learning techniques. *Frontiers in genetics*, 9:515–515, Nov 2018. 30459809[pmid].