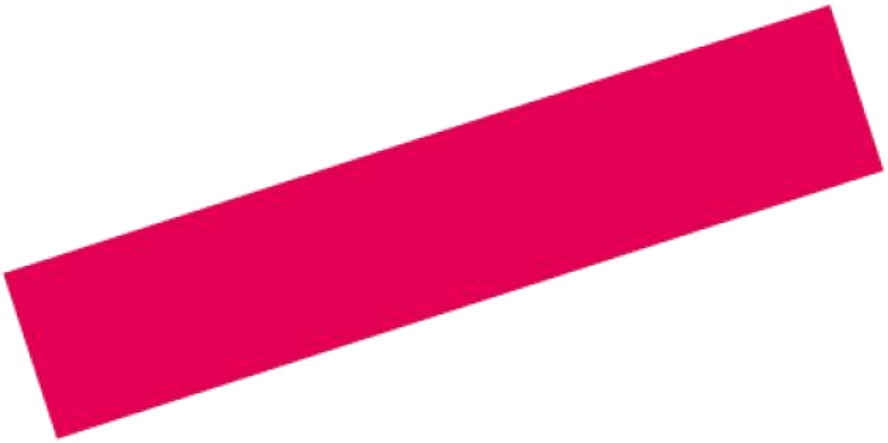

HAN_UNIVERSITY OF APPLIED SCIENCES

Data mining report

Introduction to data mining

Authors: Timon Thomassen & Robert-Jan Roest

Teacher: Mr. W. Ten Hove

Code: DATDRD06

Index

Business understanding:

The objective of this dataset is to predict the likelihood of a person developing a heart disease based on various factors. The dataset comprises individuals who either have or do not have heart disease. The purpose of this project is to create a machine learning model that, when provided with new data, can predict whether an individual is at risk of developing heart disease. This would be important in order to facilitate early intervention. It's important to note that this project is not associated with any specific company; instead, it stems from our general interest in tackling this challenge.

Data understanding:

The dataset

We chose this dataset because it offers a wide range of variables, making it an interesting study for both of us. The dataset is moderately sized, containing 1,027 rows and 14 variables, including the target variable. The target variable takes on two values: 1, indicating the presence of heart disease, and 0, indicating the absence of heart disease.

The factors influencing the target variable (1 or 0) include: age, sex, cp (chest pain type), trestbps (resting blood pressure), chol (cholesterol level), fbs (fasting blood sugar), restecg (resting electrocardiographic results), thalach (maximum heart rate achieved), exang (exercise-induced angina), oldpeak (ST depression induced by exercise relative to rest), slope (slope of the peak exercise ST segment), ca (number of major vessels colored by fluoroscopy), and thal (thalassemia). Some of these variables are binary (yes/no), which may not be suitable for visualizing correlations in scatterplots. To determine the relevant variables, we initially conducted a regression analysis. Variables that exhibited values other than 0 were retained for further analysis, and boxplots were employed for this purpose.

OLS Regression Results

Dep. Variable:	target	R-squared:	0.509
Model:	OLS	Adj. R-squared:	0.503
Method:	Least Squares	F-statistic:	87.65
Date:	Thu, 26 Oct 2023	Prob (F-statistic):	2.04e-147
Time:	12:48:14	Log-Likelihood:	-379.62
No. Observations:	1027	AIC:	785.2
Df Residuals:	1014	BIC:	849.4
Df Model:	12		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	0.7000	0.151	4.648	0.000	0.405	0.996
age	-0.0009	0.001	-0.615	0.539	-0.004	0.002
sex	-0.2036	0.026	-7.945	0.000	-0.254	-0.153
cp	0.1154	0.012	9.566	0.000	0.092	0.139
trestbps	-0.0016	0.001	-2.445	0.015	-0.003	-0.000
chol	-0.0003	0.000	-1.423	0.155	-0.001	0.000
restecg	0.0544	0.021	2.555	0.011	0.013	0.096
thalach	0.0032	0.001	5.388	0.000	0.002	0.004
exang	-0.1363	0.028	-4.945	0.000	-0.190	-0.082
oldpeak	-0.0547	0.012	-4.531	0.000	-0.078	-0.031
slope	0.0847	0.023	3.735	0.000	0.040	0.129
ca	-0.0936	0.012	-8.092	0.000	-0.116	-0.071
thal	-0.1135	0.019	-6.035	0.000	-0.150	-0.077

Omnibus:	17.547	Durbin-Watson:	1.964
Prob(Omnibus):	0.000	Jarque-Bera (JB):	17.539
Skew:	-0.296	Prob(JB):	0.000155
Kurtosis:	2.758	Cond. No.	4.45e+03

Figure 1: Regression analysis

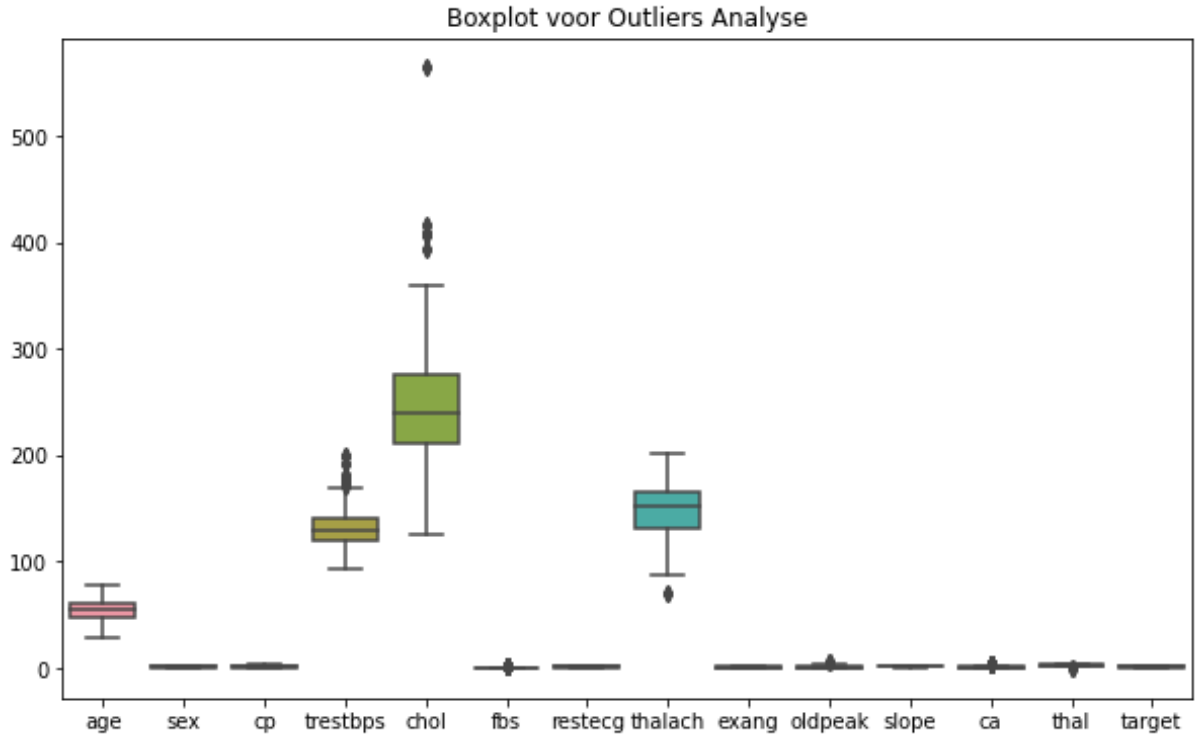

Figure 2: Boxplot for the variables

The regression analysis yielded the following outcomes in Python:

Standard Errors assume that the covariance matrix of the errors is correctly specified.

The condition number is large, 4.45e+03. This might indicate that there are strong multicollinearity or other numerical problems.

In response to this, ChatGPT recommended performing a VIF (Variance Inflation Factor) analysis. Our regression analysis indicated potential multicollinearity among the variables. To address this, we performed a VIF analysis. The results showed that the variables in our dataset do not exhibit significant multicollinearity, as the VIF values are generally low. This suggests that we can confidently include most of the variables in our regression model without significant concerns about multicollinearity.

The VIF analysis results are presented in the figure below.

	Variable	VIF
0	const	207.966728
1	age	1.428872
2	sex	1.155948
3	cp	1.293277
4	trestbps	1.167890
5	chol	1.146047
6	fbs	1.090063
7	restecg	1.064442
8	thalach	1.615182
9	exang	1.419047
10	oldpeak	1.708555
11	slope	1.643092
12	ca	1.197065
13	thal	1.137680

Figure 3: VIF analysis

VIF Analysis Outcome:

Data preparation

For data preparation, we opted to convert the CSV file into an Excel file. This choice was made primarily for our convenience, as Excel files are more straightforward to work with. Additionally, they facilitate data interpretation and analysis.

Furthermore, as part of our data preparation process, we conducted an outlier analysis. To accomplish this, we once again utilized boxplots. Our objective was to examine the presence of outliers within the dataset. Upon closer examination, we found that the outliers, while present, were well within acceptable limits. With the assistance of ChatGPT, we established that these outliers were under control and reasonable for our analysis.

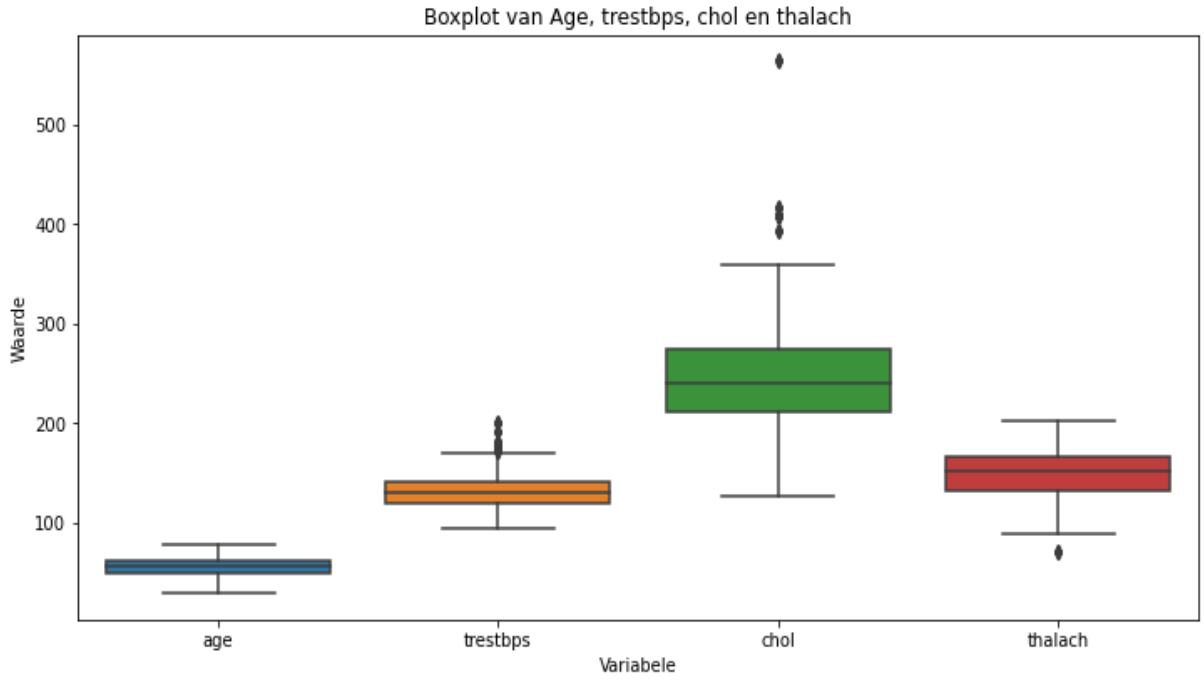

Figure 4: boxplot with variables showing the outliers.

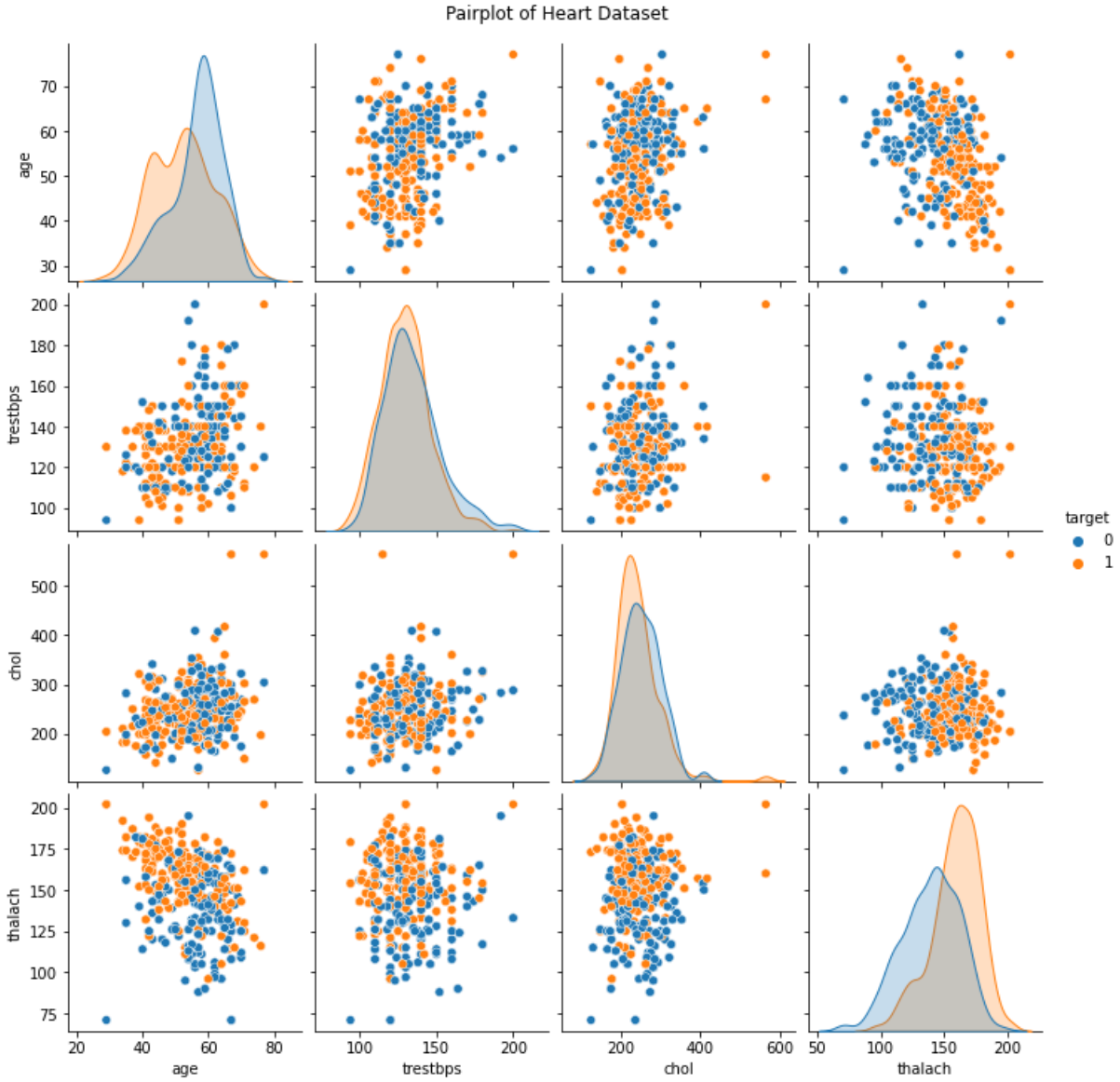

Figure 5: correlation between the variables

Despite the absence of distinct clusters or groups in the data, we observed a notable pattern where different variables tended to show a degree of grouping with one another. This observation suggests that certain variables often share common characteristics and may have some level of interdependence. As a result, this characteristic makes algorithms like K-Nearest Neighbors (KNN) particularly interesting. In KNN, the similarity between data points is determined based on their proximity, and since the different variables tend to be grouped together, this algorithm is well-suited to identify neighbors with similar characteristics.

In summary, our data preparation included the conversion of the dataset into an Excel format for enhanced readability and an analysis of outliers, which were found to be reasonable. The observed grouping of variables within the data suggests the potential effectiveness of KNN and similar algorithms for further analysis.

Modelling

For our modelling phase, we initially considered using a K-Nearest Neighbours (KNN) model because the data displayed distinct groups, even if they were closely related. We aimed to check how well the KNN model performed and if it could be a good fit for our predictive model. Later on, we explored other techniques, most of which we discovered through discussions with ChatGPT and some online research. The models we looked at include linear regression, Naïve Bayes, and a decision tree.

We used cross-validation to assess the performance of the KNN model and found that it demonstrated for similar analyses.

Cross validation

KNN model

Cross-validation provides a result between 0 and 1. In simple terms, a result of 1 is better than 0, and a result of 0.5 means the model is no better than random guessing.

We started by applying cross-validation to our KNN model, and here's what we found:

```
Cross-Validatie Scores: [0.76585366 0.74634146 0.76097561 0.71219512 0.75121951]
Gemiddelde CV Score: 0.7473170731707317
Standaardafwijking van CV Scores: 0.018867394737379254
```

Figure 6: Cross validation for the KNN-model

Cross-Validation Scores: The scores ranged from 0.712 to 0.766.

Average CV Score: With a score of 0.747, the model performs better than random guessing but isn't extremely reliable. It's somewhere in between.

Standard Deviation of CV Scores: The model shows consistency, with a standard deviation of about 0.0189. A higher standard deviation would suggest more variation in performance.

Other modells

Moving beyond KNN, we decided to evaluate various models to see which ones worked best for our purpose. To make this decision, we used cross-validation on multiple models, and we planned to focus on the models that showed the best results, while leaving behind models with lower scores.

After some experimenting and research, we concluded that, in addition to KNN, linear regression, a Decision Tree and Naïve Bayes would be good choices.

Decision tree

We checked how the Decision Tree model performed using cross-validation:

```
Cross-Validation Scores: [1.         0.94084263 1.         1.         0.94139508]
Mean CV Score: 0.9764475429371944
Standard Deviation of CV Scores: 0.02884628001610084
```

Figure 7: Cross validation for the decision tree

Cross-Validation Scores: The Decision Tree model gave very high scores, ranging from 0.941 to 1.0.

Average CV Score: The model had an average CV score of 0.976, showing it consistently performed well on our dataset.

Standard Deviation of CV Scores: Although slightly higher than the KNN model, the standard deviation was still low, indicating consistent performance.

Naïve Bayes

We also tested how well Naïve Bayes performed through cross-validation:

```
Cross-Validation Scores: [0.87317073 0.82926829 0.84390244 0.7902439  0.77073171]
Mean CV Score: 0.8214634146341464
Standard Deviation of CV Scores: 0.03684138066048624
```

Figure 8: Cross validation for the Naive Bayes

Cross-Validation Scores: The scores ranged from 0.77 to 0.87, putting this model between KNN and the Decision Tree.

Average CV Score: With an average CV score of 0.82, the model performed better than random guessing but not as well as the Decision Tree.

Standard Deviation of CV Scores: While there was a slight increase in the standard deviation, it remained low, indicating consistent performance.

Linear Regression

Lastly, we looked at how Linear Regression performed:

```
Cross-Validation Scores: [0.57537092 0.5555885  0.5469967  0.42146654 0.36874223]
Mean CV Score: 0.4936329800744056
Standard Deviation of CV Scores: 0.08267158793823741
```

Figure 9: Cross validation for the Linear Regression

The results showed that Linear Regression performed worse than random guessing, with a mean score of 0.49. To put it in perspective, random guessing would give a score of 0.50.

We used cross-validation to assess the performance of the KNN model and found that it demonstrated reasonable accuracy, but not without limitations. The Decision Tree model, on the other hand, consistently performed well with high accuracy. Naïve Bayes, while falling in between KNN and the Decision Tree, also provided respectable results. However, Linear Regression performed poorly, to the extent that random guessing was more accurate. Therefore, we advise against using Linear Regression for similar analyses.

Testing

At the start of our journey into machine learning, we chose the algorithms we wanted to use and started building our AI models with guidance from ChatGPT. Initially, we struggled to understand the code provided, but with time and research, it became clearer.

One challenge we faced initially was the inability to input our own data into the code. We worked on modifying the code to make it more flexible. Our goal was to create a model that could predict whether someone might get sick based on our own data.

As we became more familiar with the code and the underlying concepts, we gained a better understanding of how it worked and how we could customize it. With this knowledge, we could use the model's predictions to make informed decisions.

This transformation gave us more control over the AI and allowed us to make data-driven choices with confidence, knowing that the model was reliable and accurate. This transformation not only expanded our control over the AI but also provided the foundation for data-driven decision-making. With confidence in the model's reliability, we were empowered to make choices and predictions based on the valuable insights it offered.

We aimed to avoid managing four separate models that required individual data inputs. Instead, our goal was to create a unified model capable of predicting outcomes based on the ranking it received during cross-validation. To accomplish this, we needed to amalgamate the four distinct models into a single, comprehensive model. You can find this consolidated model on GitHub under the title 'All 4, Fill in data.

Evaluation

K-Nearest Neighbors (KNN):

K-Nearest Neighbours is a simple and effective machine learning algorithm used for classification and regression tasks. It operates on the principle of similarity. Here's an explanation of your findings:

In our project, we used KNN to predict whether the person would or would not get a disease. To find the optimal number of neighbours (k) in KNN, we conducted a model evaluation by varying the value of k. We observed that the model's performance remained consistent after considering about 5 neighbours. This suggests that for our dataset and problem, a relatively small number of neighbours is sufficient to make accurate predictions.

KNN's performance can be surprising because it's a non-parametric model that doesn't make strong assumptions about the data distribution. It works well when the data points are not clearly separated into distinct groups, as it relies on the concept that similar data points tend to belong to the same class or category. This adaptability to various data distributions is one of KNN's strengths which showed strongly in our case. Since our data is grouped together the KNN showed to be a good model.

Decision Tree:

A Decision Tree is a powerful and interpretable machine learning algorithm that's widely used for both classification and regression. Here's a deeper explanation:

In our project, we found that the Decision Tree performed surprisingly well. Decision Trees are constructed by recursively splitting the data based on features, which allows them to make decisions or predictions by following a path of rules. The key aspects to understand are:

The Decision Tree algorithm can work effectively with minimal feature differences. It evaluates features to find the most discriminative ones and create decision rules based on them. This process is particularly useful when some variables have subtle or minimal differences that still have significant impacts on the outcome.

Decision Trees are interpretable, meaning you can visualize the tree structure and easily understand how decisions are made. This transparency makes them valuable for gaining insights into which features are most influential in our data.

The high accuracy of 97% suggests that the Decision Tree algorithm in our project was able to capture the underlying patterns and relationships in the data, leading to precise predictions.

Naïve Bayes:

Naïve Bayes is a probabilistic machine learning algorithm commonly used for text classification, spam detection, and more. Here's an extended explanation of our findings:

In our project, we expressed some hope for Naïve Bayes and found it to be a decent performer with an accuracy of 82%. However, it didn't meet our highest expectations. Here are some insights into Naïve Bayes:

Naïve Bayes is based on Bayes' theorem and makes the "naïve" assumption that features are conditionally independent given the class. This simplifies the calculations but might not hold true for all datasets. It's particularly effective in text data where word independence can be a reasonable assumption.

An accuracy of 82% is a respectable score. It indicates that Naïve Bayes is a viable choice for classification tasks but might not be the best fit for our specific dataset. The performance of machine learning algorithms can vary depending on the nature and characteristics of the data.

The surprise we mentioned might be due to a mismatch between the independence assumption of Naïve Bayes and the actual data. It's always a good practice to experiment with different algorithms and evaluate their performance to choose the one that best fits our specific problem.

Linear Regression

Linear regression is a traditional statistical method used for modelling the relationship between a dependent variable and one or more independent variables. However, in our project, it performed poorly, to the extent that random guessing was more accurate. As a result, we advise against using it for future analyses.

In summary, each machine learning algorithm has its strengths and weaknesses, and their performance can be influenced by the nature of the data and the specific problem being addressed. While KNN, the Decision Tree, and Naïve Bayes showed promise in our project, Linear Regression proved to be inadequate for our dataset. The findings emphasize the importance of selecting the right model for the specific problem at hand, as different algorithms may yield significantly different results.

Personal reflection (Robert-Jan Roest)

For me, this course within this minor has been an enrichment of my knowledge. Before I started, the whole concept of AI and data modelling was something quite distant to me.

With the help of the classes, our teacher, and ChatGPT, we made significant progress. Thanks to my background, I already had some knowledge of data visualization, but I had never approached it this way with the assistance of Python. It was a valuable addition to my toolkit.

Collaborating with Timon went smoothly; living close to each other made working together more convenient. We had a well-distributed workload, and a great working relationship naturally formed while creating this document.

Despite the various hiccups I encountered along the way, I found it to be an enjoyable and educational task. The way we posed questions to ChatGPT had a significant impact on the outcomes. Additionally, Python's sensitivity led to some frustration here and there. However, as we progressed, everything fell into place, and now we have a well-performing predictive model that provides four results simultaneously. This gives me a strong motivation for the future, where I would like to do more with Python. This course lays the foundation for my career as a BI specialist.

Personal reflection (Timon Thomassen)

At the beginning of this minor, I had no prior experience with coding or using Python as a programming language. My exposure to VBA in Excel was limited. However, throughout this minor, and particularly in this course, I dedicated a significant amount of time to learning and practicing Python.

Furthermore, the introduction to data mining was particularly fascinating. Learning how to find and import public datasets into Python has proven to be incredibly valuable, not only in the context of this course but also for my future career.

In addition to Python, I acquired a wealth of knowledge about the utilization of ChatGPT. Prior to this minor, I had never really explored the available AI tools. The substantial attention given to the use of AI tools during the course has significantly increased my utilization of them. I now use these tools for a variety of tasks, not just those related to my academic endeavors. As a result of my practice with ChatGPT, I've also identified the most effective prompts to use.

Looking ahead to the future, I have established a foundational understanding of Python, which I am genuinely pleased about. I do not intend to specialize in data analysis, as my role will primarily involve bridging the gap between data analysts and management. Thanks to the knowledge I've acquired, I am now better equipped to facilitate these conversations. I can recognize the concepts that a data scientist discusses and understand the implications of specific choices.