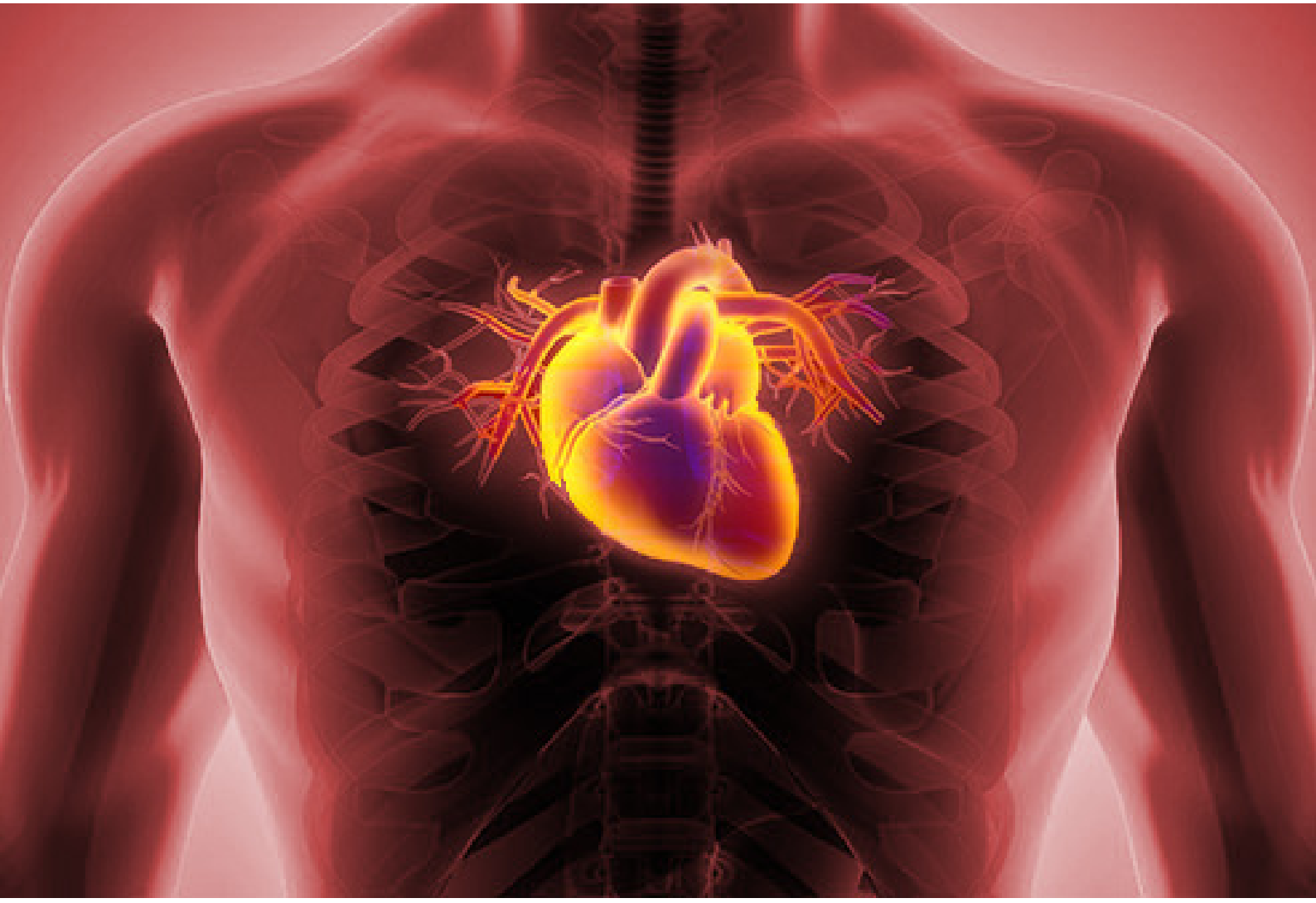
DTU Compute
Department of Applied Mathematics and Computer Science



Group 70 - Report 2
Name & Student number:
Krishna Harish Mohinani, s192182
Simon Winther Schor, s202817
Mikkel Stilling, s040038

Course 02450
Introduction to Machine Learning & Data Mining
Technical University of Denmark
20. April 2021

Preface

This report is the second of two reports in a study, conducted during the spring of 2021 as part of the course "02450 - Introduction to Machine Learning & Data Mining" offered at the Technical University of Denmark (DTU).
The objective of this report is to apply methods learned during the course on a data set, building on learnings attained from the first report to further analyse the data set.
All students have contributed to the project equally.

Table 1: Student contributions to the report

Section	**Authors**
1. Regression, part a	Simon/Krishna/Mikkel (30%/30%/40%)
2. Regression, part b	Simon/Krishna/Mikkel (40%/30%/30%)
3. Classification	Simon/Krishna/Mikkel (30%/40%/30%)
4. Discussion	Simon/Krishna/Mikkel (40%/30%/30%)
5. Exam problems for the project	Simon/Krishna/Mikkel (30%/40%/30%)

Contents

1. Regression, part a

The aim of this part of the report is to implement, perform and evaluate the results of a regularized Linear Regression model in which a variable will be predicted based on other variables from the data set of the CORIS Baseline study [1]. Further information on the data set and the CORIS study can be found in the preceding report to this one [2].

1.1

As described in the previous report, the *Adiposity* attribute has the highest correlation to the remaining attributes of the data set. For this reason, *Adiposity* was chosen as the prediction variable.

The hope was to train a model capable of predicting *Adiposity* from the other data set attributes, with low generalization error when evaluating the model. Further, the mean of *Adiposity* was found in the prior report to be 25.41, so when evaluating the model, the hope was also to accomplish an average close to this value when testing.

The feature transformation steps performed to the data set, also mentioned in the previous report, were the transformation of the *Famhist* attribute into binary values (0 for *Absent* and 1 for *Present*) and to remove the *row.names* column of the data set. Finally, the attributes were standardized by subtracting the mean and dividing by the standard deviation.

1.2

A regularization parameter λ was introduced to the model, for which the initial range of values were chosen to be in the range from 10^{-5} to 10^{8}. For each value of λ a K = 10 fold cross-validation was performed to estimate the generalization error, the results of which can be seen in figure 1.

While not obvious by visual inspection of figure 1 (b), by inspection of the numbers constituting the plot, the lowest test error was found at $\lambda = 10$. What this means, in terms of variance and bias, is that at values lower than $\lambda = 10$ the trained models vary unnecessarily much, while at values higher than $\lambda = 10$ too much bias is introduced, both of which affects the performance of the predicted model negatively.

Now, turning to figure 1 (a), notice that at low values of λ the variance of the model is high and the bias low, while at high values of λ the variance of the model is low and the bias is high (the weights of the attributes are small). It can be seen that at the optimal value of $\lambda = 10$, low bias in the model is introduced, but at the cost of variance not being particularly reduced.
As is apparent in both figure 1 (a) and (b), the range of values of λ were chosen in orders of magnitude of ten. This means that $\lambda = 10$ may not be the value with the lowest test error, but possibly some other value in the vicinity is. While not of particular importance to this section, this will be addressed in section 2.2.

With regards to prediction capability of the model, running it ten times with randomized test sets produced averages of *Adiposity* in the range [24.89 , 25.66], quite close to the target value of 25.41. Evaluating whether

the generalization error of the model is low is done by comparison to other models, which will also be addressed in the next part of this report.

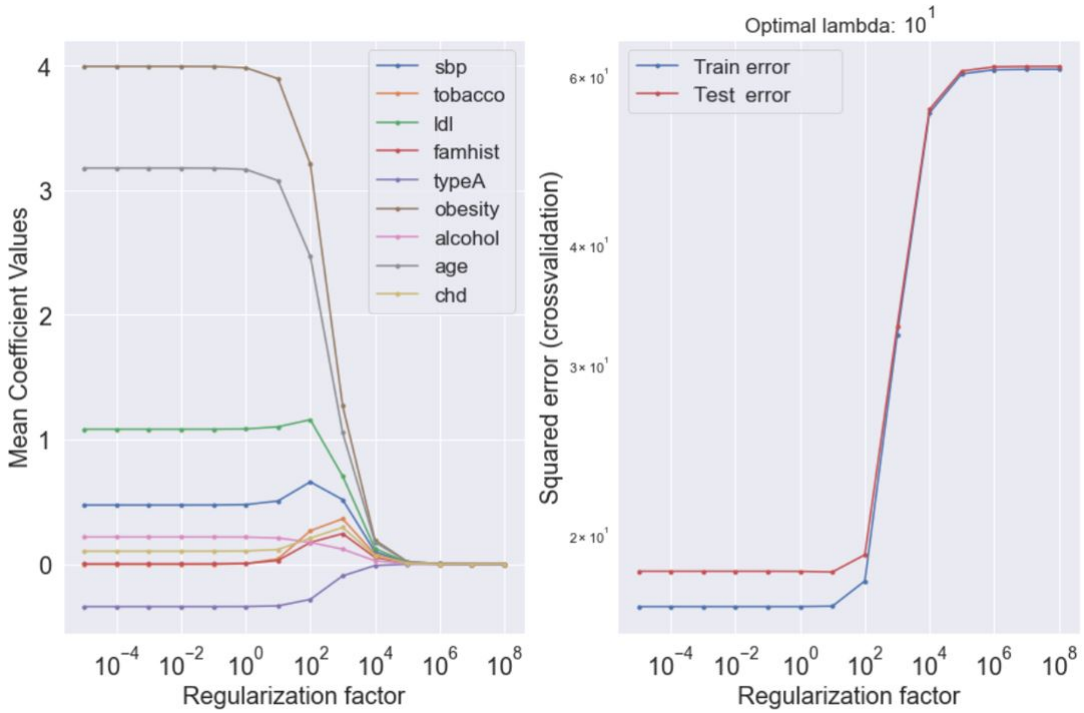


Figure 1: (a) Optimal weights of attributes as a function of λ, (b) test and training error as a function of λ.

1.3

Looking at figure 1 (a), one may also notice that *Obesity* and *Age* have the highest weights. This was looked into a bit further.

The regularized Linear Regression model was trained by minimizing the sum-of-squares error term, presented in the course book [3] section 14.1,

$$E_\lambda(\boldsymbol{w}, w_0) = \|\boldsymbol{y} - w_0\mathbf{1} - \hat{\boldsymbol{X}}\boldsymbol{w}\|^2 + \lambda\|\boldsymbol{w}\|^2$$

with the optimal weights w^*, assuming $\boldsymbol{X}$ and $\boldsymbol{y}$ are scalars,

$$w^* = \frac{Xy}{X^2 + \lambda}$$

The weights attained for each attribute can be seen in figure 2.

For any new data observation the prediction of its *Adiposity* will be predicted according to these weights. The results presented in the figure will look familiar to the person having read the prior report to this. Comparing these weights with the correlations of the attributes, presented in the previous report, one sees that the weights match the correlations quite well, both with regards to relative size as well as sign. Thus, the obtained results are in accordance with prior learnings and therefore quite sensible.

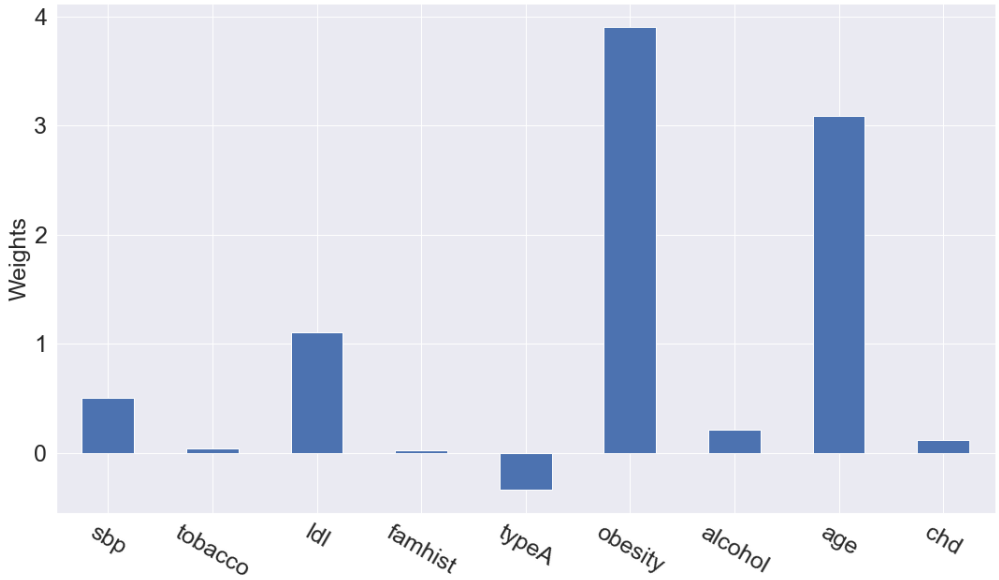


Figure 2: Weights of attributes for the optimal regularized Linear Regression model.

2. Regression, part b

The aim of this part of the report is to compare the Linear Regression model from section 1 with an artificial neural network (ANN) model. Additionally, in order to put the performance of these two models into perspective, a trivial model was made, to serve as baseline for the comparison.

2.1

For the ANN-model, a number of test runs were performed in order to determine a reasonable range of hidden units in the hidden layer. Since the test runs were quite time-consuming, for the 2-level cross-validation used to compare the models, $K_1 = K_2 = 5$ folds were chosen. The ANN test error results are shown in table 2. The table shows how much the test error is reduced when allowing the model to include more hidden units. The values max. h show the upper level of the range of possible h's starting from $h = 1$. max. h does not represent actual chosen values for h when executing the scripts.
Noticing that the test errors markedly decrease as the maximum possible number of hidden units goes from max. $h = 1$ to max. $h = 2$ to max. $h = 3$, from which it stabilizes. Hence, a range for the number of hidden units chosen for the further study, clearly, needed only be in a smaller range.

In table 2, one may notice that the test error is smaller for $h \in [1 : 4]$ than for $h \in [1 : 5]$. This is due to the randomized nature of the splits of the data set when running the tests, indicating that the performance of the model isn't significantly improved when going from max $h = 4$ to max $h = 5$.
Further, for the initial and additional test runs performed, when allowing the model the option of a 6th hidden unit, the lowest test error was always obtained at a number of hidden units less than 6. For these reasons, the range of $h \in [1 : 5]$ was chosen for the further study.

Table 2: Error-results of test-runs for upper level values of h.

max. h	E_{ANN}^{test}	change from $h = 1$
1	85.53	-
2	33.78	60.5 %
3	22.89	73.2 %
4	20.39	76.2 %
5	20.82	75.6 %
6	19.85	76.8 %

2.2

As mentioned, a simple model was also to be introduced to serve as a baseline. This simple model simply compared averages of the test sets to the known whole data set means.
From application of the cross validation, table 3 was constructed, logging the test errors E_i^{test}, regularization strengths λ for each fold in each of the three regression models, as well as the actual number of hidden units h applied in the ANN-model.

Inspecting the table, it becomes apparent that the ANN and the Linear Regression model has markedly lower test-errors compared to the Baseline. However, it is currently not obvious whether the difference between the two is statistically significant.

Table 3: Regression summary

Outer fold	ANN		Linear Regression		Baseline
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	3	23.80	12	19.26	72.52
2	5	17.96	12	14.09	56.24
3	5	19.33	13	15.42	68.47
4	3	21.78	13	24.51	62.21
5	5	21.20	13	17.76	44.21
Average	-	**20.82**	-	**18.20**	**60.75**

Additionally, the ANN-model has selected a combination of 3 and 5 hidden units for the individual folds during this particular run. In general, the values for h has fallen within the interval of 3 and 5 for all runs performed.

The calculated optimal regularisation strength λ_i^* fell between 12 and 13. This differs slightly from the value in 1.2. Given that the initial range of values of λ were in orders of magnitude of ten, λ_i^* was found to be 10. However for this test, the range was narrowed down iterative from run to run to focus on values from 10 to 15. This was done in order to increase the resolution of the values, giving a more precise measure of λ providing the lowest test errors.

2.3

A statistical test were carried out, comparing the three regression models, following the approach for correlated T-test for cross validation, outlined in method box 11.4.1 [3]. The results were gathered in table 4, which show the null-hypotheses, the confidence intervals, the P-values along with a conclusion according to the null-hypotheses for each comparison.

Table 4: Regression statistic comparison test

Pairwise Test	$P - value$	CI_{lower}	CI_{upper}	Conclusion
$E_{Baseline}^{test} = E_{Linear}^{test}$	$1.41 \cdot 10^{-3}$	27.75	58.04	H_0 is rejected
$E_{ANN}^{test} = E_{Linear}^{test}$	$9.36 \cdot 10^{-5}$	66.49	94.82	H_0 is rejected
$E_{ANN}^{test} = E_{Baseline}^{test}$	$9.65 \cdot 10^{-5}$	-44.44	-31.07	H_0 is rejected

Inspecting the table, It it becomes apparent that all the tests significantly differs from each other, evaluating with a 95% confidence interval, hence all null-hypotheses are rejected. As also suspected from inspecting table 3, the Baseline is significantly outperformed. Additionally, the Linear Regression model performs significantly better than the ANN model, making this model recommended for regression of *adiposity*, given the same means of resources. i.e. 10,000 iterations and 5 folds per level.

However it should be noted, that increasing the number of iterations and hidden units, would - while demanding more computations power - decrease the test error for the ANN model. A test-run of the regression models, in which 40.000 iterations were run. The strongest model in this context, turned out to be ANN (See tables 8 & 9 in the appendix).

As explained above in this chapter, the cross validation for the model selection is separate from the cross validation for the statistical tests. This explains the slight differences in the test E_i^{test} for model selection and

E_i^{test} for the statistical test. This measure is done to avoid performing a statistical test on the same data that is used to perform model selection. By computing a new random cross validation it is ensured that the cross validation split for the statistical setup II is independent from the cross validation used for model selection.

Additionally, it should be noted that the model used the most frequent occurring regularization parameters (i.e the most common λ and h). However, the amount of folds K applied for the cross validation for the statistical test follows the amount for the model selection, which in this case was $K = 5$.

3. Classification

3.1

This section of the report will focus on the comparison of the performance of all three classifiers that were used to classify a binary attribute of the data set. The data set was primarily built to determine if a patient is either *chd* negative or positive. For comprehensive understanding, the attribute *chd* will be predicted based on the other nine attributes. It is crucial to mention that this is a binary classification problem because the attribute *chd* can only be 0 or 1.
To handle the data, attribute *row.names* was removed and the attribute values of *famhist* were transformed from present and absent to 1 and 0 respectively. To avoid issues from differences in scale and variation, the remaining nine attributes are standardized before the classification analysis.

3.2

The three models that will be analyzed and compared are Logistic Regression, K-nearest neighbor, and a Baseline model. In this section, two-level K-fold cross-validation is implemented to examine the models and the code from the regression part is more or less restructured for the classification section.

Logistic Regression is Linear Regression that is extended to classification by the use of probabilities. Since the output of the Linear Regression is a general continuous number the decision will be made by applying the logistic sigmoid transfer function. However, the complexity-controlling parameter will keep being λ and after some pretests, the lambda interval will be in the range of 10 to 35.

$$\lambda \in [10 : 35]$$

The KNN algorithm assumes that similar things exist in close proximity. The KNN classifier looks for K samples that are 'in the neighborhood' to the test sample. The decision is made by selecting the class that has more representation to the test sample. The distance between samples is measured with euclidean distance and that is why the input attributes have to be standardized for this model. Also, the complexity-controlling parameter is K which represents the number of the neighbors. The range selected for this parameter is from 1 to 30 based on some pretesting.

$$K_{neighbors} \in [1 : 30]$$

Finally, the Baseline model will compute the largest class on the training data and predict everything in the test data as belonging to that class. It is crucial to mention that the data set is not balanced i.e. 302 samples are *chd* negative and 160 samples are *chd* positive. In this case, the largest class is when *chd* is zero (negative) and is used to predict y on the test data.

3.3

Outer fold	KNN		Logistic Regression		Baseline
i	k_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	25	0.319	27	0.404	0.383
2	25	0.234	27	0.234	0.234
3	29	0.370	15	0.283	0.478
4	29	0.348	15	0.326	0.435
5	29	0.283	15	0.261	0.391
6	29	0.261	15	0.196	0.370
7	29	0.152	15	0.130	0.196
8	29	0.217	15	0.217	0.391
9	29	0.348	15	0.283	0.283
10	29	0.239	15	0.261	0.304
Average		**0.277**		**0.260**	**0.346**

Table 5: Two-level cross-validation table used to compare the three classification models.

Table 5 highlights the test error and the optimal complexity-controlling parameter for the three models. It is essential to notice the average errors for all the three models i.e. E_{KNN}^{test} is 0.277, $E_{Logistic}^{test}$ is 0.260 and $E_{Baseline}^{test}$ is 0.346.

Also, Logistic Regression performs better than KNN and the mean error of the Baseline model is greater than the previous models as expected because the model is not based on the other attributes. Recall, that 34.6% of the people in the sample are diagnosed with heart disease. Hence, it was anticipated that the Baseline model on average guess incorrect 34.6% of the time. As observed, the estimated test error changes within each outer fold and this is because of the relatively small data size. Consequently, the estimated test error is more sensitive to incorrect predictions. Finally, the complexity controlling parameter for KNN and Logistic Regression does not change much and is fairly consistent. The number of the neighbors is either 25 or 29 in the case of KNN and the lambda values for Logistic Regression is either 27 or 15.

3.4

As mentioned in the previous section, separate cross-validation is computed for statistical evaluation. The reason behind this is to ensure cross-validation split for the statistical evaluation is independent of the cross-validation used for model selection. Moreover, the most frequently occurring regularization parameters (lambda and number of neighbors) are used to also test the optimal parameter. In this case, the lambda (Logistic Regression) value used is 15 and the number of neighbors (KNN) used are 29.

Table 6 highlights the statistical evaluation of the three classification models. As observed, the Baseline model does not have the same generalization error as KNN and Logistic Regression, and hence its rejected. To fail to reject the null hypothesis the p-value should be higher than 0.05. When comparing Logistic Regression and KNN the resultant p-value is 0.6829 so we can fail to reject H_0 with high confidence. In actuality, the confidence interval is very small and close to zero, affirming the resemblance in performance.

Table 6: Classification statistic comparison test - Statistical evaluation of the three classification models using setup 2 (correlated t-test)

Pairwise Test	$P-value$	CI_{lower}	CI_{upper}	Conclusion
$E^{test}_{Baseline} = E^{test}_{Logistic}$	0.019	0.018	0.151	H_0 is rejected
$E^{test}_{Logistic} = E^{test}_{KNN}$	0.683	-0.046	0.067	H_0 is not rejected
$E^{test}_{Baseline} = E^{test}_{KNN}$	0.003	0.032	0.115	H_0 is rejected

3.5

Ultimately, the attribute *chd* is predicted using the Logistic Regression model with the optimal λ obtained previously in two-level cross-validation i.e. 15 or 27. After running the model, the weights show that the attributes *age* and *famhist* have more importance than the others to predict *chd*. This may be related to the fact that *chd* has no great correlation with the other attributes in contrast to *adiposity* (regression), which shows a great correlation with obesity so that its weight predominates over the other attributes.

4. Discussion

4.1

- In Regression, part a, the Linear Regression model was shown to predict averages of *Adiposity* close to the known actual average value.
- In Regression, part b, it was found that of the three models (Baseline, ANN & Linear Regression) the Linear Regression model had the lowest test error. This could be said conclusively, since statistic comparison tests showed that the test errors 95% confidence intervals were non-overlapping.
- In Classification, of the three models (Baseline, KNN, Logistic Regression) Logistic Regression had the lowest test error, however statistical tests (95% confidence intervals) failed to reject the null-hypotheses that the KNN- and Logistic Regression model to perform similar to each other.

4.2

To apply perspective to some of the findings from this study, the writers of this report at looked in to present literature to find similar studies, which were based on the data set used.

Computer scientists A.H. Gonsalves, F. Thabtah, R.M.A. Mohammad and G. Singh, presented in 2019 [4] their work on determining which of various possible classification models is more beneficial determining coronary heart disease using the same data set as used in this study.

In their study, Gonsalves et. al. carried out a 10 fold cross validation tests for three machine learning model approaches, namely Naïve Bayes , Support Vector Machine and Decision Trees. None of these methods were similar to the ones tested in the study done for this report, however it is still possible to compare overall results. In their study, the Naïve Bayes technique showed a test error E_{NB}^{test} of 0.284, Support Vector Machine $E_{SVM}^{test} = 0.290$ and Decision Trees $E_{DT}^{test} = 0.292$. These prediction performance measures fall within the same ranges as the ones assessed in this report, positioning above the Baseline and below the KNN- and the Logistic Regression techniques.

However, similarly to what was stated earlier in this report, the research article studied stresses the need to notice that the accuracy vs. error is not a sufficient performance measure due to the class imbalance in observations in the data at hand. This becomes very apparent with the Baseline model, as it neither applies any machine learning nor any other dynamic prediction techniques, and is still correct more than 65% of the time.

In this matter, ideally, it would be beneficial to include measures that either handle class imbalance by weighting the observations according to importance (i.e. to penalize) or equal out the observations with pre model data processing (to re-sample) or alternatively, introduce a new threshold parameter and change the performance measure to something which is invariant to class imbalance. When applying these measures, the goal of the study should generally be taken into account. in this case, it would be deemed worse to falsely predict negative cases, than to have false positives, as the former case would conclude in missed necessary treatment of actual chd-patients.

However, as stated in the project description, class imbalance management is outside the scope for this report,

which is limited to lectures 5 to 8. In conclusion, the presence of a higher occurrence of negative chd-cases in the data set would be that all models would gain accuracy, making it intuitively seem that the models are "better" at predicting the value of a given feature.

4.3

A topic worth briefly discussing is the potential changes in outcome related to changing the dependant variable of the regression analysis from *Adiposity* to *Obesity* in sections 1 & 2, as this feature showed such high correlation to *Adiposity* while also resembling intuitively very similar features to the common person. In case of *obesity*, the lambda values achieved were $\lambda = 10^0$ which is lower compared to *adiposity* regression where $\lambda = 10^1$. In terms of weights, *adiposity* is the attribute that mostly influences the prediction of *obesity*. *Obesity* regression was not the main aim of the team but still it was investigated because even *obesity* shows high correlation with other attributes.

5. Exam problems for the project

Question 1. Spring 2019 question 13

The correct answer is C:
To see this, start by observing that all four predictions have four positive observations (red crosses) and four negative observations (black circles). Thus, movement from point to point on the ROC curve will happen in steps of 1/4, either on the FPR-axis or the TPR-axis.
Now, start with a threshold value > 1 in which FPR $= 0$ and TPR $= 0$. As the threshold is lowered, one first observes a positive observation giving FPR $= 0$, TPR $= 1/4$, then a negative observation giving FPR $= 1/4$, TPR $= 1/4$, then another negative observation giving FPR $= 1/2$, TPR $= 1/4$. This rules out all options but C. One can then continue in this manner to confirm that option C indeed *does* correspond to the ROC curve.

Question 2. Spring 2019 question 15

The correct answer is C:
In total there are $N = 135$ observations, distributed among the four congestions levels as:

$$N(y=1) = 33+4+0 = 37 \qquad N(y=2) = 28+2+1 = 31$$

$$N(y=3) = 30+3+0 = 33 \qquad N(y=4) = 29+5+0 = 34$$

The parent impurity measure is:

$$I(r) = 1 - max\left(\frac{37}{135}, \frac{31}{135}, \frac{33}{135}, \frac{34}{135}\right) = \frac{98}{135}$$

For the two-way split of $x_7 = 2$ the number of observations and the impurity measures are:

$$N(v_1) = 1 \qquad I(v_1) = 1 - max\left(\frac{0}{1}, \frac{1}{1}, \frac{0}{1}, \frac{0}{1}\right) = 0$$

$$N(v_2) = 134 \qquad I(v_2) = 1 - max\left(\frac{37}{134}, \frac{30}{134}, \frac{33}{134}, \frac{34}{134}\right) = \frac{97}{134}$$

The impurity gain then is:

$$\Delta = \frac{98}{135} - \left(\frac{1}{135}\cdot 0 + \frac{134}{135}\cdot\frac{97}{134}\right) = \frac{1}{135} \approx 0.0074$$

Question 3. Spring 2019 question 18

The correct answer is A:
To see this, one must start by taking note of the number of hidden layers in the ANN and the sizes of them. Given that there is only one single hidden layer, one needs only consider sizes:

Input size: $i = 7$
Hidden layer size: $h = 10$
Output size: $o = 4$

The number of parameters is determined by the connections between layers and the biases in every layer, i.e.:

$$(i \cdot h + h \cdot o) + (h + o)$$

Inserting the values, one gets:

$$7 \cdot 10 + 10 \cdot 4 + (10 + 4) = 124$$

Question 4. Spring 2019 question 20

The correct answer is D:
Option A can be ruled out, since ***A:*** $b_1 \geq -0.16$ followed by ***C:*** $b_2 \geq 0.01$ will result in a horizontal line going from $(b_1, b_2) = (-0.16, 0.01)$ and to the right in the classification boundary figure, which is not what is seen.
Option B can be ruled out, since ***A:*** $b_1 \geq -0.76$ followed by ***B:*** $b_1 \geq -0.16$ makes ***B*** redundant, since it is not to the left of ***A*** in the classification boundary figure.
Option C can be ruled out, since ***A:*** $b_2 \geq 0.03$ will result in a horizontal line all through the classification boundary figure.
Left is option D and one can simply follow the propositions of ***A***, ***B***, ***C*** and ***D*** to confirm that indeed these result in a classification boundary figure as shown.

Question 5. Spring 2019 question 22

The correct answer is C:
To see this, one can initially focus on the logistic regression. The number of outer folds is $K_1 = 5$, the number of λ values is $L_\lambda = 5$ and the number of inner folds is $K_2 = 4$.
The time taken for cross validation of one outer fold is

$$T_{cv} = L_\lambda \cdot K_2 \cdot (T_{train} + T_{test}) = 5 \cdot 4 \cdot (8ms + 1ms) = 180ms$$

To estimate performance one must test and train one final model. This takes

$$T_{val} = T_{train} + T_{test} = 8\ ms + 1\ ms = 9\ ms$$

This routine needs to be performed for each outer fold, thus the total time elapsed for logistic regression is

$$T_{log.rec} = K_1 \cdot (T_{cv} + T_{val}) = 5 \cdot (180\ ms + 9\ ms) = 945\ ms$$

Similarly for ANN one gets

$$T_{cv} = L_{n_h} \cdot K_2 \cdot (T_{train} + T_{test}) = 5 \cdot 4 \cdot (20\ ms + 5\ ms) = 500\ ms$$

$$T_{val} = T_{train} + T_{rest} = 20\ ms + 5\ ms = 25\ ms$$

$$T_{ANN} = K_1 \cdot (T_{cv} + T_{val}) = 5 \cdot (500\ ms + 25\ ms) = 2625\ ms$$

The total time to compose the table therefore is

$$T_{Total} = T_{log.rec} + T_{ANN} = 945\ ms + 2625\ ms = 3570\ ms$$

Question 6. Spring 2019 question 26

The correct answer is B:
The aim is to determine which of the four observations $\boldsymbol{b} = [b_1, b_2]$ has the highest probability P for y = 4. The method for this is straight forward. Using option B as example:
First, calculate $\hat{y}_k$ for $k = 1, ..., 3$ and calculate the exponential of this:

$$\hat{y}_1 = \begin{bmatrix} 1 \\ b_1 \\ b_2 \end{bmatrix}^\top \boldsymbol{w}_1 = \begin{bmatrix} 1 \\ -0.6 \\ -1.6 \end{bmatrix}^\top \begin{bmatrix} 1.2 \\ -2.1 \\ 3.2 \end{bmatrix} = -2.66 \qquad \Rightarrow \qquad e^{\hat{y}_1} = 0.0699$$

$$\hat{y}_2 = \begin{bmatrix} 1 \\ b_1 \\ b_2 \end{bmatrix}^\top \boldsymbol{w}_2 = \begin{bmatrix} 1 \\ -0.6 \\ -1.6 \end{bmatrix}^\top \begin{bmatrix} 1.2 \\ -1.7 \\ 2.9 \end{bmatrix} = -2.42 \qquad \Rightarrow \qquad e^{\hat{y}_2} = 0.0889$$

$$\hat{y}_3 = \begin{bmatrix} 1 \\ b_1 \\ b_2 \end{bmatrix}^\top \boldsymbol{w}_3 = \begin{bmatrix} 1 \\ -0.6 \\ -1.6 \end{bmatrix}^\top \begin{bmatrix} 1.3 \\ -1.1 \\ 2.2 \end{bmatrix} = -1.56 \qquad \Rightarrow \qquad e^{\hat{y}_3} = 0.2101$$

These values are then simply put into the softmax transformation to obtain the per-class probabilities P of $y = 1, ..., 4$. For instance

$$P(y = 2|\hat{\boldsymbol{y}}) = \frac{e^{\hat{y}_2}}{1 + e^{\hat{y}_1} + e^{\hat{y}_2} + e^{\hat{y}_3}} = \frac{0.09}{1 + 0.07 + 0.09 + 0.21} = 0.06$$

The procedure is repeated for each observation giving the results shown in table 7 below, from which one can then see, that B is the correct answer.

Table 7: Per-class probabilities for the four observations with the most probable in bold

	A	B	C	D
P(y=1)	**0.78**	0.05	**0.63**	**0.68**
P(y=2)	0.20	0.06	0.33	0.29
P(y=3)	0.02	0.15	0.04	0.03
P(y=4)	0.00	**0.73**	0.00	0.00

References

[1] J. E. Roussouw et al. "Coronary risk factor screening in three rural communities". In: *South African Medical Journal 64: 430–436* (1983).

[2] Krishna Harish Mohinani et al. *Introduction to Machine Learning & Data Mining - Group 70 - Report 1.* 2021.

[3] Tue Herlau et al. *Introduction to Machine Learning and Data Mining.* 2021.

[4] Amanda H. Gonsalves et al. "Prediction of Coronary Heart Disease Using Machine Learning: An Experimental Analysis". In: *Proceedings of the 2019 3rd International Conference on Deep Learning Technologies.* ICDLT 2019. Xiamen, China: Association for Computing Machinery, 2019, pp. 51–56. ISBN: 9781450371605. DOI: 10.1145/3342999.3343015. URL: https://www.researchgate.net/profile/Rami-Mohammad/publication/335094208_Prediction_of_Coronary_Heart_Disease_using_Machine_Learning_An_Experimental_Analysis/links/5e3fb042458515072d8aa2ec/Prediction-of-Coronary-Heart-Disease-using-Machine-Learning-An-Experimental-Analysis.pdf.

[5] D. S. Freedman et al. *The body adiposity index (hip circumference ÷ height$^{1.5}$) is not a more accurate measure of adiposity than is BMI, waist circumference, or hip circumference.* https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3477292/
Visited on 02-03-2021. 2012.

[6] Rayman W. Bortner. *A short rating scale as a potential measure of pattern A behavior.* https:/https://www.sciencedirect.com/sdfe/pdf/download/eid/1-s2.0-0021968169900617/first-page-pdf
Visited on 08-03-2021. 1969.

Appendix

Table 8: Regression summary (40,000 iterations)

Outer fold	ANN		Linear regression		Baseline
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	1	17.18	12	19.26	72.52
2	1	16.15	12	14.09	56.24
3	1	14.29	13	15.42	68.47
4	1	12.83	13	24.52	62.21
5	1	18.21	13	17.76	44.21
6	1	12.00	13	13.72	76.53
7	1	32.56	13	36.77	65.10
8	1	11.87	13	12.06	59.54
9	1	15.23	13	15.84	39.09
10	1	23.41	13	19.80	49.40
Average	-	**17.37**	-	**18.20**	**60.75**

Table 9: Regression statistic comparison test(40,000 iterations)

Pairwise Test	$P - value$	CI_{lower}	CI_{upper}	Conclusion
$E_{base}^{test} = E_{LinReg}^{test}$	$3.51 \cdot 10^{-6}$	33.27	52.33	H_0 is rejected
$E_{ANN}^{test} = E_{LinReg}^{test}$	0.20	-3.31	0.80	H_0 is not rejected
$E_{ANN}^{test} = E_{base}^{test}$	$1.43 \cdot 10^{-6}$	35.25	53.19	H_0 is rejected