Danmarks Tekniske Universitet

(02450) Introduction to Machine Learning and Data Mining

Project 2

Carla Zdravkovic Hugod
s224209

August Emil Holm Jørgensen
s224166

Bertram Nyvold Larsen
s224194

November 16th 2023

Summary

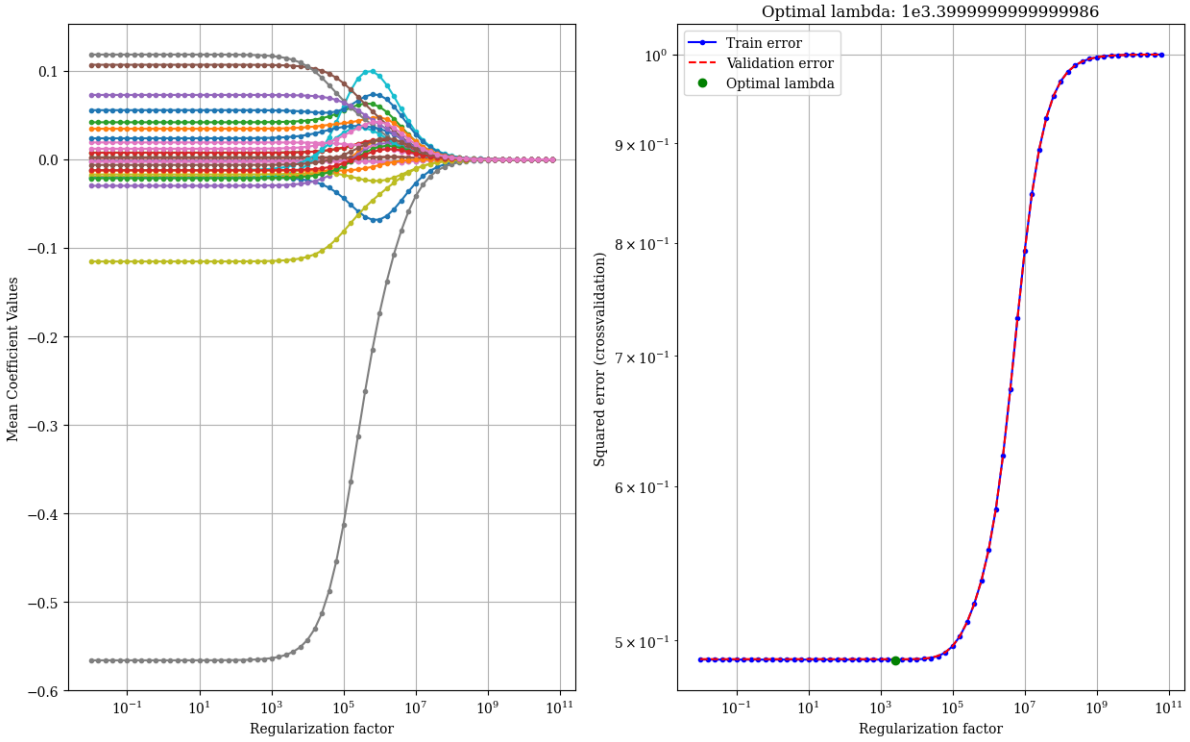
Student ID	Regression	Classification	Discussion	Exam questions
Bertram: s224194	30%	30%	40%	33.33%
Carla: s224209	30%	40%	30%	33.33%
August: s224166	40%	30%	30%	33.33%

1 Regression

1.1 Regression a)

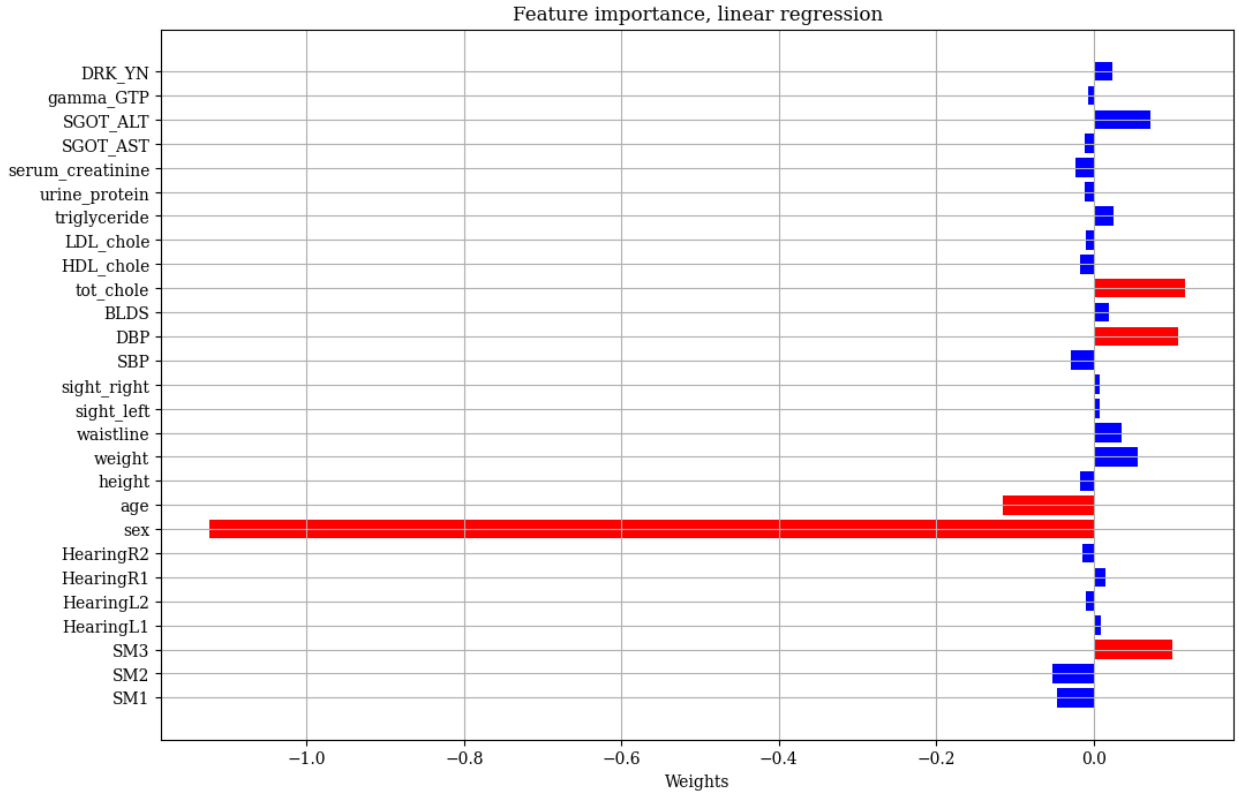
In the regression section, our goal is to predict the hemoglobin level, based on all the other attributes. We chose this, since hemoglobin seemed to have a higher correlation with the other attributes, (as can be seen in the appendix). Our aim is therefore to predict a model that can complete this task successfully. We have standardized all the data, such that is has mean 0 and standard deviation 1, and applied one-out-of-K-encoding on the nominal attributes. This is done on the feature, 'Smoking Status', since we don't aim to rank the various choices or levels of smoking. This goes for the hearing attribute as well, which can take two values; abnormal and normal.

The regularization parameter λ is introduced, and the generalization error is estimated for $\lambda \in [10^{-20}, 10^{18}]$. For each value of lambda we use 10-fold cross validation. This is seen in figure (1) below, for $\lambda \in [10^{-2}, 10^{11}]$. The purpose of using regularization is to substantially reduce the variance in our models without introducing too much bias. The optimal λ is therefore the regularization parameter that fulfills this purpose. $\lambda = 10^{3.4}$ seems to have the lowest generalization error, however most small values of lambda corresponds to a small error, but namely also high variance. (for a more zoomed out plot, look in appendix, 4)



Figur 1: Left: weights as a function of lambda. Right: Squared error as a function of lambda

The output, y, of the model with the lowest generalization error is computed by using the weighted sum of features along with the intercept term. The intercept term and weights are determined in the training of the model on the given data, hence the prediction is then made upon the value of the weighted sum. As λ increases, the values of the weights approaches 0. A positive weight, means that an increase in the value of the corresponding attribute, would result in predicting on a higher level of hemoglobin in the blood (for this case). The same goes for negative weights, they decrease the prediction. The absolute size of the weight indicates the relative importance of that attribute when the model attempts predict the value of the given attribute. Below is a bar plot, illustrating the feature importance.



Figur 2: Weights of the best linear regression model with 5 largest absolute weights in red

We see that the most important features are *sex*, *age*, *SM3* (currently smoking), *tot_chole* (total cholesterol level) and *DBP* (diabolic blood pressure). Amongst these, *sex* is the most important feature. This is indicated by a large negative weight. The *sex* attribute is 1 for female and 0 for male. This means that if *sex* is high (female), the models are more likely to predict a lower level of hemoglobin in the blood. This is also the case for age; older patient's would have lower levels of hemoglobin. On the other hand, the larger positive weights, *SM3, DBP and tot_chole*, indicates, that large values of these attributes, would result in a larger probability of the prediction of the hemoglobin level. Specifically for the smoking status *SM3: currently smoking*, larger values, would result in more hemoglobin the blood. This is consistent with current research on the topic[1]. Conclusively, we observe that a typical patient, predicted to have a high level of hemoglobin exhibits the following physiological symptoms; younger men, current smokers with high blood pressure and cholesterol level. On the other hand, a typical patient predicted to have a low hemoglobin level, posses the following traits; older females, non-smokers with low blood pressure and cholesterol level.

1.2 Regression b)

In implementing a two-level cross-validation approach for a linear regression model, we introduce the regularization parameter, λ. This parameter is designed to fine-tune the linear regression, controlling the model's complexity and preventing overfitting. Evaluating test errors derived from models trained with the optimal parameter, selected within the inner loop, allows us to identify parameters where the model demonstrates robustness to variations in data. Firstly, we split the data into 10 training and test sets. For each iteration we find the optimal value of λ, using 10 fold cross validation on the test sets. In the inner loop, the test set is then split into a validation set. We then train the model using the linear regression model on the full outer fold training set. Lastly, we compute the generalization error for each fold.

In this section, we will compare three models. The linear regression model from earlier, an artificial neural

[1] `https://journals.lww.com/aomr/fulltext/2020/32020/effects_of_smoking_on_hemoglobin_and_erythrocytes.5.aspx`

network (ANN) and a baseline model. As per the project description, the baseline model is defined as a linear regression model with no features, which means that its guess, is the mean of the target attribute in the training data. The ANN is build with 1 hidden layer, with hyperbolic tangent as the activation function.

The mean squared error is used as the error measure:

$$MSE = \frac{1}{N^{test}} \sum_{i=1}^{N^{test}} (y_i - \hat{y}_i)^2$$

Below is a table of the performance of the three models, amongst with the regularization parameter $\lambda_i^* \in [10^{-5}, 10^5]$ and number of hidden units $h_i^* \in [2, 6]$.

Outer fold	ANN		Linear Regression		Baseline
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	4	0.478	46.416	0.490	0.995
2	5	0.480	46.416	0.486	1.002
3	4	0.487	46.416	0.488	0.997
4	5	0.461	46.416	0.484	1.006
5	5	0.477	46.416	0.484	1.004
6	4	0.470	46.416	0.492	0.998
7	5	0.477	46.416	0.494	0.992
8	4	0.490	46.416	0.489	1.006
9	5	0.510	46.416	0.496	1.004
10	5	0.462	599.484	0.487	0.996
Generalization error	0.479		0.489		0.999

Tabel 1: Two-level cross-validation table used to compare the three models

Here it is seen that the error for the ANN classifier and the linear regression is smaller than for the baseline model. This is aligned with our expectations, since the baseline, merely predicts on the mean. Hence, the ANN and linear regression seems to outperform the baseline model. Furthermore, it appears that the ANN model has the lowest generalization error amongst the three models. However, to ensure that this is correct, we will do some statistic evaluation of the models. We are using the statistical **setup II**.

Model A	**Model B**	**CI**	**t-statistic**	p**-value**	**Statistic difference**
Baseline	Linear Regression	$[0.427, 0.615]$	12.510	$5.396 \cdot 10^{-7}$	Yes
Baseline	ANN	$[0.461, 0.561]$	23.143	$2.498 \cdot 10^{-9}$	Yes
Linear Regression	ANN	$[-0.095, 0.075]$	-0.265	0.797	No

Tabel 2: Statistic evaluation of the tree models

We have used a threshold of $\alpha = 0.05$, hence calculating the 95%-confidence intervals. Firstly, there seems to be a significant difference in all comparisons with the baseline. Comparing the linear regression model and the baseline model, there is a significant statistical difference, since 0 is not in the confidence interval, and the p-value, $p = 5.396 \cdot 10^{-7}$,is incredibly small compared to $\alpha = 0.05$. The same applies to the comparison of the ANN model and the baseline model. There is also a significant difference between these, which is to be expected. Interestingly, there is not a significant difference between the ANN model and the linear regression model. However, the generalization error is lower for the ANN model, which could be an argument for using this model.

2 Classification

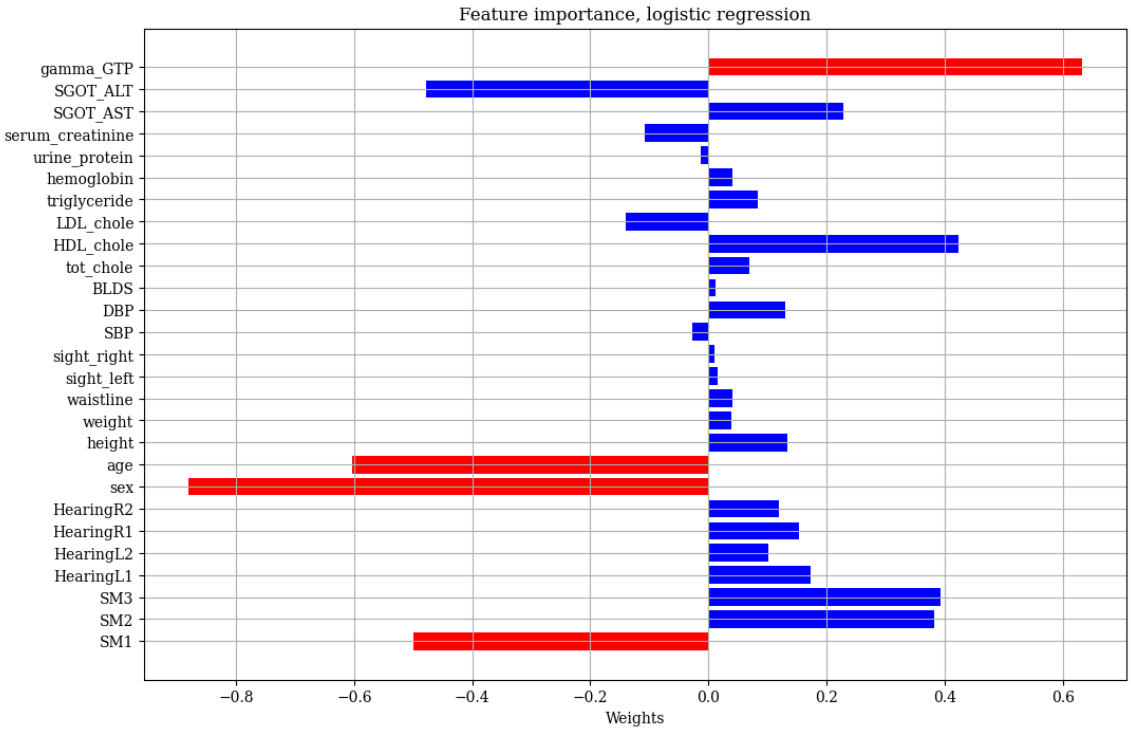
In the classification segment, we address a binary problem - identifying whether a person drinks or not. This implies that 'y' across all the classification models will represent the binary attribute indicating whether the individual is a drinker or not (**Y**es / **N**o). Our objective is to build three distinct classification models and conduct statistical evaluations for comparison. Specifically, we will develop a baseline model, a logistic regression model, and as a third model we will create a k-Nearest Neighbors (KNN) classification model.

For the logistic regression model we introduce the regularization parameter $\lambda \in [10^{-5}, 10^{5}]$. We chose this range, through trial and error, and found that smaller and larger values, only had insignificant differences. We utilized the LogisticRegression model from sklearn, wherein the parameter C represents the inverse of the regularization strength. We conducted a two-level cross-validation with 10 outer folds and only 5 inner folds due to computational complexity. However, this meant we were able to leverage all the available data, approximately 1 million data points.

The complexity-controlling parameter for the KNN classifier is k, representing the number of nearest neighbors. The values to be examined for k will be in the range of $[50, 80]$, with 6 models to train. This interval was determined as appropriate following multiple runs of different ranges starting with $k = 2$. For the KNN classifier, we used the KNeighborsClassifier model from sklearn. It takes the parameter 'n_neighbors', which corresponds to the k value mentioned earlier. Due to the relatively high values of 'k', the computational complexity significantly increases. Consequently, we employed 10 outer folds and 5 inner folds once again. Additionally, we restricted our dataset to only 20%, yielding around $200,000$ data points, however this still turned out to be sufficient.

The logistic regression model works by estimating probabilities to determine the likelihood of belonging to a particular class. The trained model contains a vector of weights, aligned with the same dimensions as the data points, which, when combined with a given input, calculates a value z. This z represents a linear combination of the weights and input vector. Then we compute $\sigma(z)$ producing a probability p for the input's classification as either class $C : 0$ or its complementary probability, $1-p$, for the input's classification as class $C : 1$.
Moreover, as $z \to -\infty$ the sigmoid function will act, $\sigma(z) \to 0$, and $\sigma(z) \to 1$ vice versa, which gives us the opportunity to interpret the weights of the model.

For instance, upon training our logistic regression model, the following weights represent the best model trained on the outer test-set in the last outer fold of our two-level cross-validation:



Figur 3: Weights of the best logistic regression model with 5 largest absolute weights in red

From this, we can interpret the important features, amongst others, to be *sex*, *age*, *SM1* (never smoked) and *gamma_gtp* (liver enzyme). Drinkers are categorized yes: 1, no: 0. A large positive weight of *gamma_gtp* (liver enzymes) indicates, that if the corresponding attribute is positive, it would result in a higher probability of classifying the patient to be a drinker. A similar statement can be inferred for all other attributes that bear a significantly negative or positive weight. However, we must exercise caution in drawing conclusions based solely on this information. The notably high negative weight associated with gender should not lead us to assume that all or even most women are non-drinkers. Such assumptions cannot be made solely from this weight. It's plausible that this gender weight compensates for other attributes where women typically have higher values than men, but without comprehensive knowledge of how all technical attributes differ on average between male and female bodies, we cannot really interpret anything definitively from a single weight. However, by combining certain weights, we can gain insights about a very specific group of people. Specifically, a non-smoking (SM1), elderly women who are short in height, that exhibit high SGOT_ALT, low gamma_GTP, and low HDL_cholesterol levels. Such individuals are almost certainly non-drinkers. Conversely, individuals opposite to this profile would likely be drinkers. This is all based on the understanding of how the weights influence 'z' and how the sigmoid function works, as previously described.

We will now compare the three models, based on the error E_i obtained when testing the trained model in the i'th outer fold on D_{test}. Where the error E_i is the fraction of misclassification.

$$E = \frac{\{\text{Number of misclassified observations}\}}{N^{test}}$$

Outer fold	KNN		Logistic Regression		Baseline
i	k_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	68	0.284	0.278	0.273	0.501
2	80	0.278	3.59	0.273	0.502
3	74	0.278	0.278	0.275	0.502
4	80	0.275	3.59	0.272	0.503
5	74	0.286	0.278	0.273	0.498
6	80	0.284	46.4	0.272	0.500
7	80	0.277	3.59	0.272	0.502
8	74	0.287	$1.00 \cdot 10^{-5}$	0.278	0.499
9	62	0.285	$1.00 \cdot 10^{-5}$	0.272	0.499
10	80	0.285	3.59	0.274	0.500
Accuracy	71.81 %		72.66%		50.01%

Tabel 3: Two-level cross-validation table used to compare the three models

The accuracy is calculated as $1 - \frac{1}{n}\sum_{i=1}^{n} E_i^{test}$.
It's observed that the accuracy achieved by both the KNN classifier and the logistic regression surpasses that of the baseline model. This aligns with our expectations since the baseline merely predicts the majority class, and we have nearly perfect class balance. Consequently, both the KNN and logistic regression models seem to outperform the baseline.

In Table 3, it's evident that the complexity control parameter, k_i, exhibits significant variation. This variability results in similar performance among models trained with k-nearest neighbors within the range of 50 to 80. However, in ranges with lower k values, the best model consistently emerged with the highest k within that range. Consequently, for our specific task, the KNN model performs optimally at higher k values.

Moreover, it's apparent that the logistic regression model obtains the highest accuracy among the three models, all though only slightly ahead of the KNN model. To validate this, we will conduct statistical evaluations of the models. Here the statistical **setup II** is used:

Model A	**Model B**	**CI**	**t-statistic**	***p*-value**	**Statistic difference**
Baseline	Logistic Regression	$[0.213, 0.242]$	36.117	$4.734 \cdot 10^{-11}$	Yes
Baseline	KNN	$[0.189, 0.249]$	16.612	$4.615 \cdot 10^{-8}$	Yes
KNN	Logistic Regression	$[-0.020, 0.037]$	0.676	0.516	No

Tabel 4: Statistic evaluation of the tree models

We've used a threshold of $\alpha = 0.05$, enabling the calculation of 95% confidence intervals. When comparing the baseline model to the logistic regression model, a statistically significant difference is evident for a 95% confidence interval. This is supported by the interval $[0.213, 0.242]$, where 0 doesn't fall within it, and the p-value of $4.734 \cdot 10^{-11}$ is smaller than $\alpha = 0.05$. A similar outcome arises when comparing the baseline model to the KNN model. Although Table 4 indicates the superiority of the logistic regression model over the other two models, there isn't a statistically significant difference between the KNN model and the logistic regression model. Therefore, we can't conclusively make this statement. However, we can affirm that both the KNN and logistic regression models outperform the baseline model.

3 Discussion

Throughout the creation of various regression and classification models, we gained valuable insights into our data set. One crucial decision involved the treatment of the "Smoking Status"attribute. We argued it was a nominal attribute, leading us to implement one-out-of-K encoding. This approach ensured that the levels of smoking weren't ranked against each other, like this:

(SM1, 1): *Never*
(SM2, 2): *Former*
(SM3, 3): *Current*

By employing this encoding method, we aimed to prevent implying any superiority of one smoking status over another. Interestingly, the weights attributed to these three smoking levels varied significantly after encoding. Specifically, 'Currently smoking' (SM3) had a large absolute weight in the regression tasks, whereas 'Never smoked' (SM1) had a large absolute weight in the classification tasks. Had we chosen not to encode the attribute, it might have suggested a hierarchy among the smoking statuses, and confused the models. This assumption could be relevant in health-related contexts, as we discovered a connection between higher levels of hemoglobin in the blood and 'Never' smokers. This finding follows existing research indicating smoking's impact on hemoglobin levels (source) [2]. Despite this discovery, we opted for encoding to maintain neutrality and simplicity in our analysis.

During the regression part of our report, both the ANN model and the linear regression model achieved relatively low generalization errors - compared to the baseline model, that was extremely close to a random guess, since we have almost perfect class balance in the data set. During the classification part, we created three classifiers, the KNN model, the logistic regression and a baseline model. Notably, the more intricate models outperformed the baseline. For the KNN model, we limited our usage to roughly $20\% \approx 200,000$ points of the entire data set to reduce computational load. We might have seen a slightly different result had we used all the available data, but given the data follows observable pattern, 200.000 data points should suffice to capture it.

Additionally, when constructing both regression and classification models, we incorporated all 27 features from the dataset, excluding only the target attribute. However, as observed in the feature importance plots, not all features significantly impacted predictions. We could have utilized PCA here, to identify principal components explaining the majority of the data set's variance. Subsequently, disregarding certain insignificant features, such as 'eye sight,' could have reduced computational complexity. This approach would practically streamline the process of gathering data for a new patient, making it less extensive and faster - given that we want to predict if a person drinks or not and hemoglobin levels. For other attributes 'eye sight' might not be a so useless attribute.

Examining previous analyses of our dataset, we observe that numerous attempts have been made to create a classifier for predicting whether patients are drinkers or not. One particular classifier, using XGBoost, achieved an accuracy comparable to ours. ($\approx 74\%$) [3]. Another logistic regression classifier got an accuracy of ($\approx 71\%$) [4]. The majority of the found studies, use the data set for classification tasks predicting either the drinking or smoking status. We found one study, using linear regression to predict the hemoglobin level [5], The individual claimed a 100% accuracy, a figure that appears highly unlikely. Most classification studies align with our findings, highlighting the significance of attributes like 'SM1, never smoked' and 'gender' in predicting the drinking status. For a feature importance plot from another study, look in appendix.

[2] https://journals.lww.com/aomr/fulltext/2020/32020/effects_of_smoking_on_hemoglobin_and_erythrocytes.5.aspx
[3] https://www.kaggle.com/code/raman209/prediction-of-drinkers-using-body-signals#8.-EVALUATION
[4] https://www.kaggle.com/code/anamibnjafar0/alcohol-drinker-prediction#Model-Training
[5] https://www.kaggle.com/code/rakibhasan3948/hemoglobin-label-predictin-with-100-accuracy

4 Conclusion

Throughout the project, we have created 3 regression models and 3 classification models. In general, the baseline models made poor predictions and classifications, whereas the linear and logistic regressions, along with the ANN and KNN made better predictions and classifications. For the classification tasks the KNN had an accuracy of 71.82%, the logistic regression had 72.66% accuracy, whereas the baseline only had 50.01% accuracy. In the regression part, the ANN model had the lowest generalization error of 0.479, the linear regression had 0.489, whereas the baseline had a generalization error of 0.999.

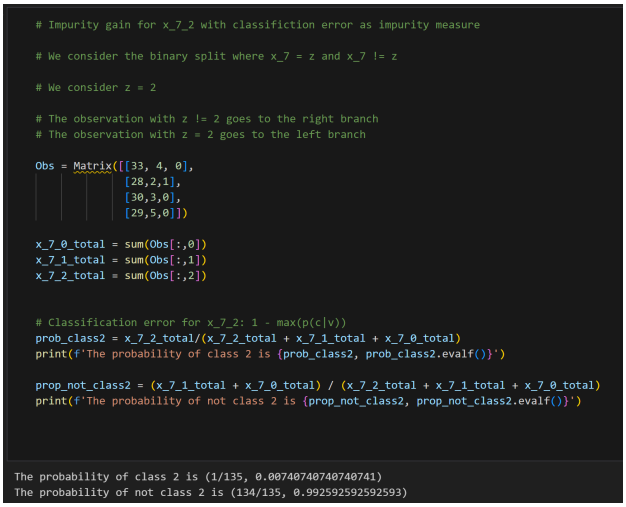
5 Exam Questions

Tabel 5: Answers summary

Question	1	2	3	4	5	6
Answer:	x	C	A	D	C	x

5.1 Question 1. Spring 2019 question 13

5.2 Question 2. Spring 2019 question 15



```
# Impurity gain for x_7_2 with classifiction error as impurity measure

# We consider the binary split where x_7 = z and x_7 != z

# We consider z = 2

# The observation with z != 2 goes to the right branch
# The observation with z = 2 goes to the left branch

Obs = Matrix([[33, 4, 0],
              [28,2,1],
              [30,3,0],
              [29,5,0]])

x_7_0_total = sum(Obs[:,0])
x_7_1_total = sum(Obs[:,1])
x_7_2_total = sum(Obs[:,2])


# Classification error for x_7_2: 1 - max(p(c|v))
prob_class2 = x_7_2_total/(x_7_2_total + x_7_1_total + x_7_0_total)
print(f'The probability of class 2 is {prob_class2, prob_class2.evalf()}')

prop_not_class2 = (x_7_1_total + x_7_0_total) / (x_7_2_total + x_7_1_total + x_7_0_total)
print(f'The probability of not class 2 is {prop_not_class2, prop_not_class2.evalf()}')
```

```
The probability of class 2 is (1/135, 0.00740740740740741)
The probability of not class 2 is (134/135, 0.992592592592593)
```

Answer: **C**

5.3 Question 3. Spring 2019 question 18

We got the ANN with 1 hidden layer with 10 neurons.

The input size is 7. This gives us, $7 \cdot 10 = 70$ weights. For the output layer, we have 4 classes. This adds, $10 \cdot 4 = 40$ weights.

Then we have the biases. There are 10 biases for the hidden layer and 4 biases for the output layer.

In total, 110 weights + 14 biases = 124 parameters.

Answer: **A**

5.4 Question 4. Spring 2019 question 20

For this decision tree question, we begin by looking at split of A.

A should split in such a way, that it splits into two groups of classes, Congestion level 1 and 2, and Congestion level 1, 3 and 4.

The only possible value z is then, -0.76.

This eliminates solution A and C.

Next up, we can look at split B. It should split into Congestion 1 and 2. B should then have the split, $b_2 \geq 0.03$.

Answer: **D**

5.5 Question 5. Spring 2019 question 22

```
# 2-level cross validation
# K2 = 4 ; Inner loop with 4 folds

K1 = 5
K2 = 4
L = 5

models = (K2 * L + 1) * K1
print(f'The number of models is {models}')

# Time to train
train1 = 20
test1 = 5

train2 = 8
test2 = 1

time = models * (train1 + test1) + models * (train2 + test2)
print(f'The time to train is {time}')
```

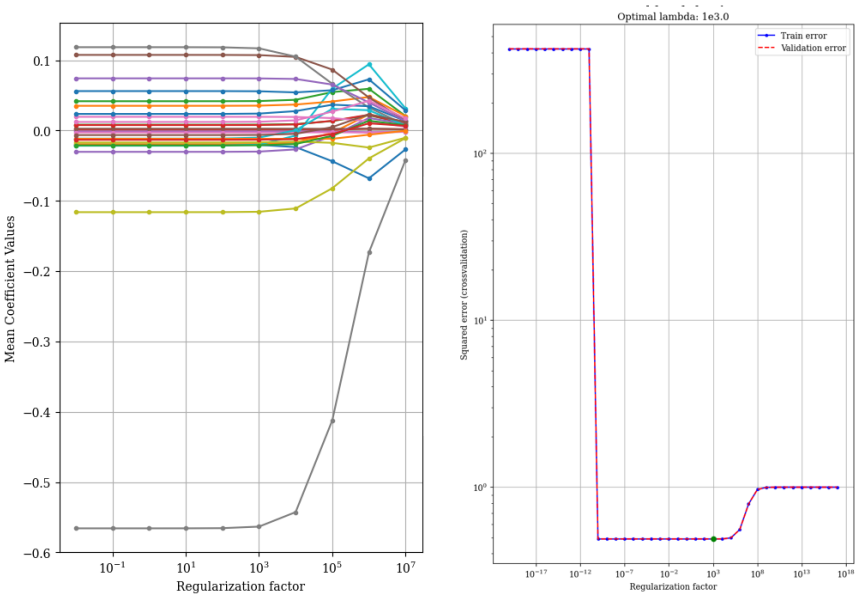
```
The number of models is 105
The time to train is 3570
```

Answer: **C**

5.6 Question 6. Spring 2019 question 26

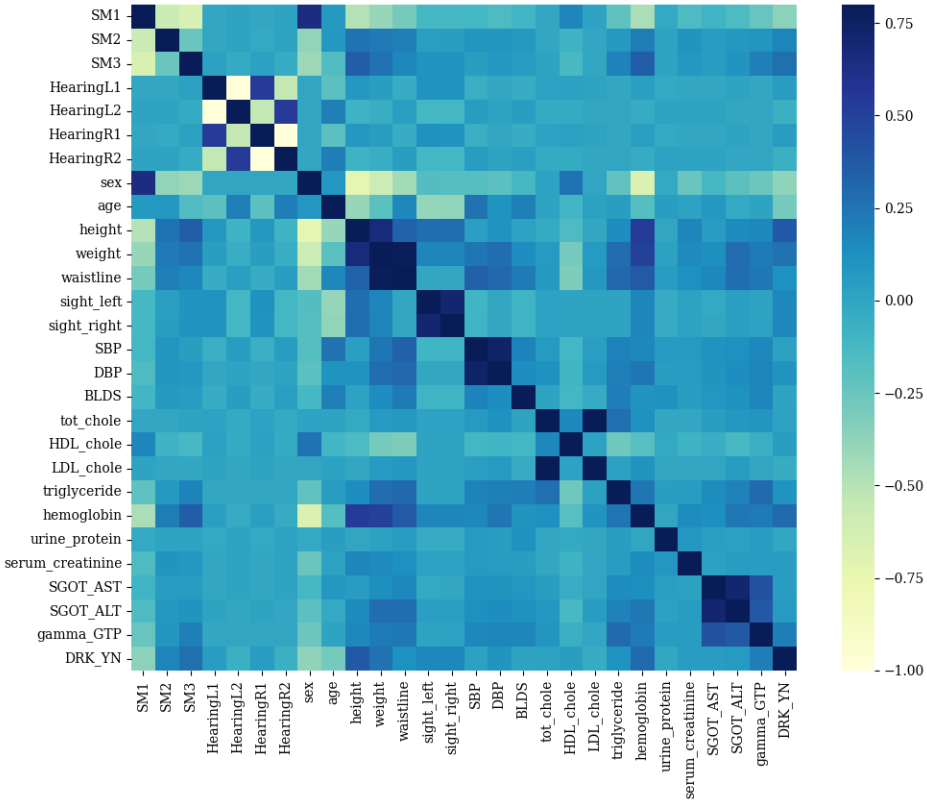
6 Appendix

Regularized linear regression

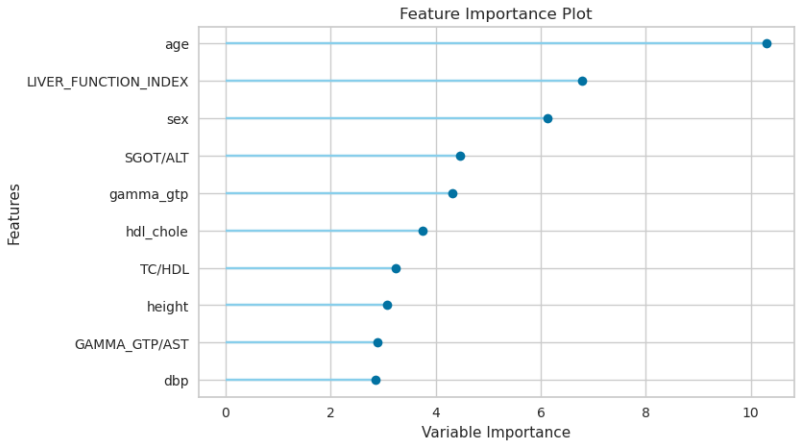


Figur 4: Plot of the regularized linear regression in a larger interval of λ

Correlation matrix, heat map



DIFFERENT STUDY



Link to study.[6]

[6] https://www.kaggle.com/code/furkannakdagg/smoking-drinking-prediction-complete-eda-pycaret#6