Technical University of Denmark

31015 Introduction to Machine Learning and Data Mining

Report 2: Regression and Classification

Morten Nissov, s163962
Emma Demarecaux, s176437
Johan Dybkjaer-Knudsen, s180049

Introduction

When doing data analysis, a common framework is used as described in ”Introduction to Machine Learning and Data Mining (Herlau, Schmidt, Mørup (2018), p. 15). Data analysis requires five steps to be done: Data understanding, data preparation, data modelling, evaluation and results.

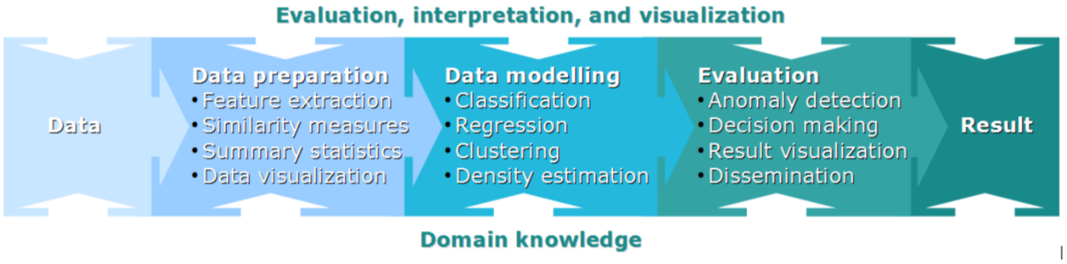

Figure 1: Common framework

We will be focusing on the Data Modelling and Evaluation steps in this paper. The data set in question contains salary information for residents of the United States in the 1980s. This paper is divided up into two parts, one which will focus on fitting the data to a regression, and another which will focus on fitting the data to a classification model. The models will then, in each of their respective sections, be evaluated and discussed.j

1 Regression

1.1 Regression problem

As mentioned in the last report, the purpose of this project is to predict the monthly wages of people based on seven features (hours, IQ, educ, exper, tenure, age, black). In our case, we have to predict a continuous response y given a training set comprised of 935 observations x:

Attribute Name	Description	Discrete/Continuous	Type
`wage`	monthly earnings	Continuous	Ratio
`hours`	average weekly hours	Continuous	Ratio
`IQ`	IQ score	Discrete	Interval
`educ`	years of education	Discrete	Ratio
`exper`	years of work experience	Discrete	Ratio
`tenure`	years with current employer	Discrete	Ratio
`age`	age in years	Discrete	Ratio
`black`	=1 if black	Discrete	Nominal

Table 1: Data Attributes and Their Attribute types

We wish to come up with a way to predict y from x:

$$y = f(x, w) + \epsilon$$

1.2 Linear regression with forward selection

First, we will try to predict the monthly wages of people using linear regression with feature subset selection. To measure how well we can predict the monthly wage, we will use the squared error between the true and estimated wage. In our estimation we will use two levels of cross-validation:

- on the outer layer, we use 10-fold cross-validation to estimate the performance of our model, i.e., we compute the squared error averaged over 7 test sets;
- in the inner layer, we use 10-fold cross-validation too.

For the rest of this section the whole dataset was normalized.
First, we use KFold() function to set up the cross-validation partitions needed. For each split we compute the squared error with all features selected (no feature selection). We can see in the following figure how the forward selection is made. A feature is added to the existing subset until no further improvement is possible. In this case (the selected model) there are 5 features in the subset which are ”age”, ”black”, ”expr”, ”educ” and ”iq”.

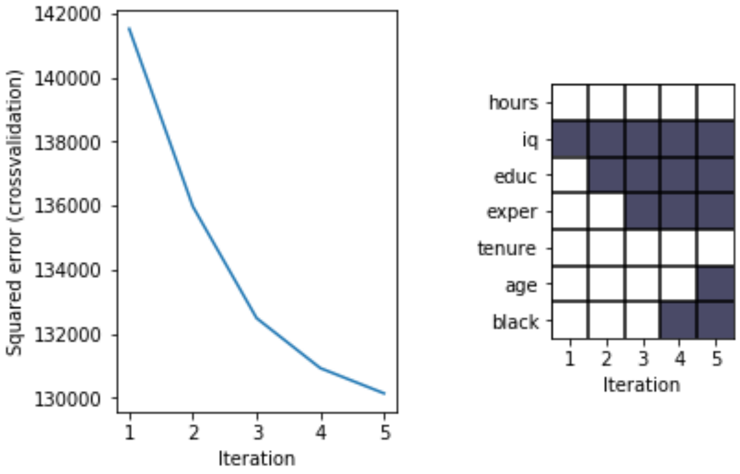

Figure 2: Forward selection (selected model)

For the outer cross-validation, we can see in the following figure which attribute were selected from inner cross-validation for each fold. There is a king of tendency that often all attributes are selected except ”hours” and sometimes ”tenure”. Therefore it seems that all the features except hours have their importance in predicting the monthly wages.

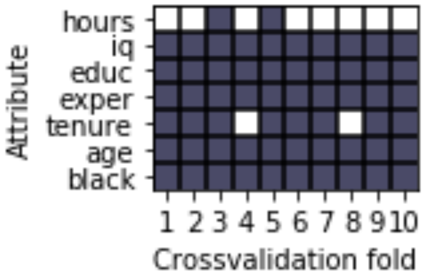

Figure 3: Outer cross-validation

Then training and test errors were computed with and without feature selection. Training and tests errors are very similar maybe due to the fact often all the features are selected with the feature selection. The results are displayed in the following table:

	Training error	**Test error**	**R^2 train**	**R^2 test**
Linear regression without feature selection	132689	136200	0.188	0.161
Linear regression with feature selection	133100	137014	0.185	0.156

Table 2: Results of the linear regression

Finally, we inspect the selected feature coefficients effect on the entire dataset and plot the fitted model residual error as function of each attribute to inspect for systematic structure in the residual.

We can see on the following figure that the residual errors don't follow any pattern for the 5 selected attributes so there is no need to combine or transform our attributes.

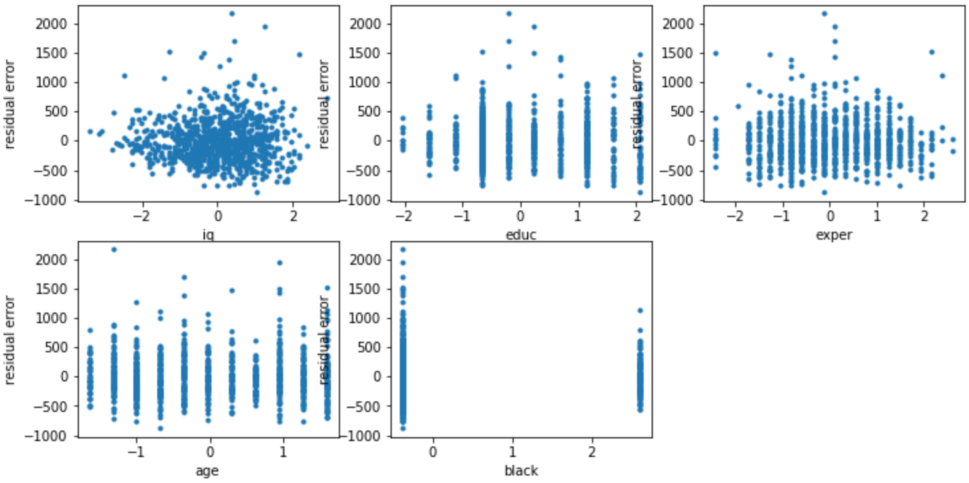

Figure 4: Residual error vs selected attributes

1.3 Prediction of a new data observation

As we said in the previous subsection, we have chosen through forward selection 5 features among the 7 features we have. Our model prediction is then a linear combination of the 7 attributes (with coefficients 0 before "hours" and "tenure". It can be written as:

$$y = w_0 + \sum_{i=1}^{7} w_i x_i$$

where the w_i are the parameters of the model and the x_i are the given observations.
In our case, we have the following result:

$$y = 61.97x_{iq} + 115.72x_{educ} + 49.28x_{exper} + 41.56x_{age} - 41.69x_{black}$$

The data were standardized so "educ" (highest coefficient) is the attribute which influence the most this model. It is quite consistent with the principal component analysis as the first principal component is mostly influenced by the same attribute "educ".

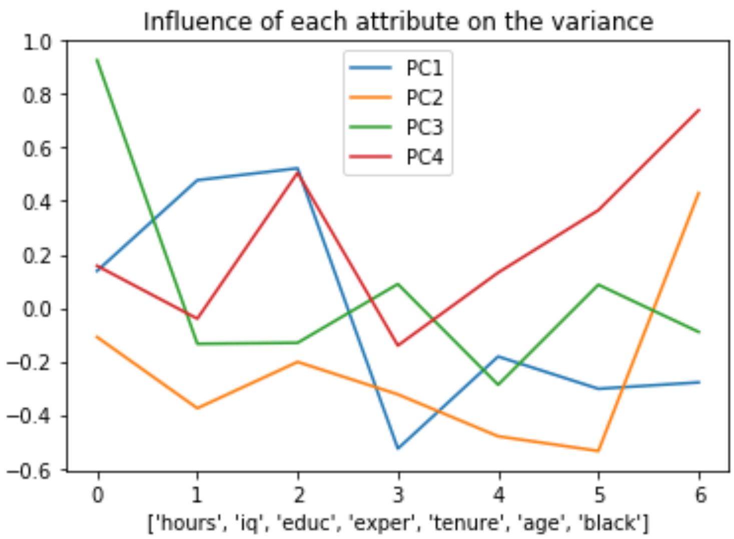

Figure 5: Influence of attributes on principal components, after standardization

We are going to find the number of units that minimize the test error of the best model after using two levels of cross-validation.

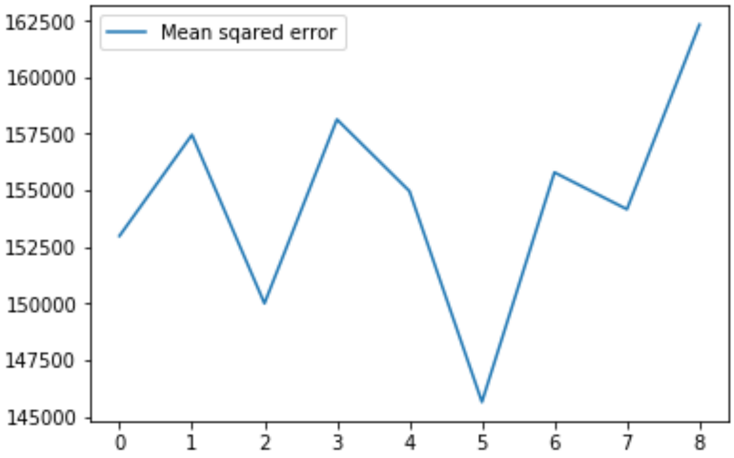

Figure 6: Mean square error depending on the number of hidden units depending

We see that the test error is minimum for 5 hidden units with an error equals to 145643.1. In the following figure the predicted (orange) and real value (blue) of our test set are often very different. It may explain the fact that our test error seems very high.

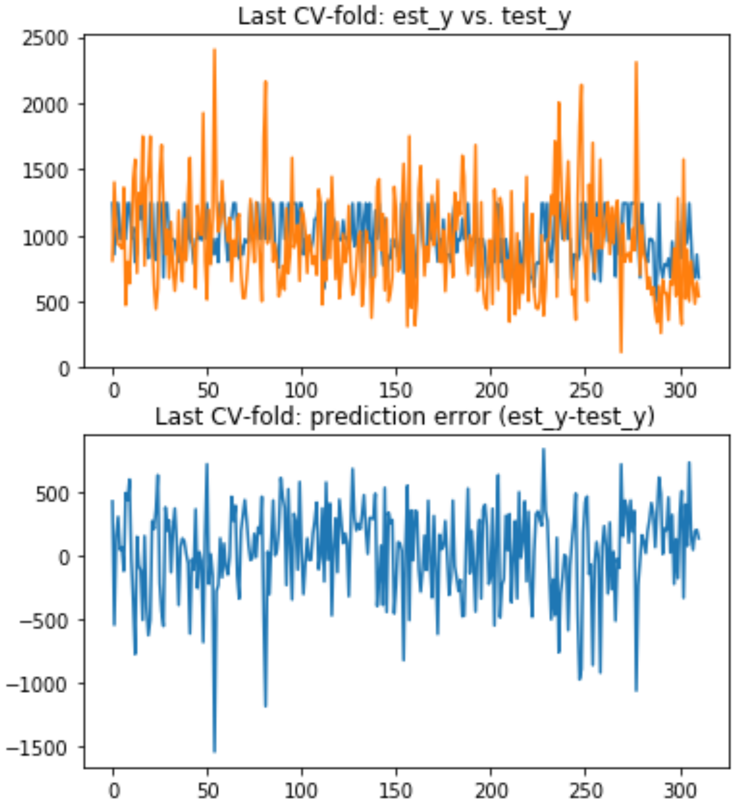

Figure 7

1.4 Model evaluation

We will now evaluate which of the three regression models is the best. It seems obvious to simply choose the model with the lowest generalization error. However, the generalization errors are based on slightly random test and training sets, thus we cannot confidently say that one model will perform better than the other. For this reason, we use credibility intervals based on Bayes' theorem, to assess with a 95% probability that one model performs better than the other.

The credibility interval $[z_L, z_U]$ follows the formula:

$$z_L = \mathrm{cdf}_{\mathrm{st}}^{-1}(\frac{\alpha}{2}|\nu, \bar{z}, \tilde{\sigma}), z_U = \mathrm{cdf}_{\mathrm{st}}^{-1}(1 - \frac{\alpha}{2}|\nu, \bar{z}, \tilde{\sigma})$$

We then check if zero is included in the credibility interval. If this is the case, we cannot confidently say that there is a significant difference between the two.

Comparing the models

We use 10-fold cross validation to test the true accuracy of each selected model, meaning that a two-layer cross validation approach is applied. To be able to asses if a model predicted correctly, we have categorized the wages into four quartiles. We also want to compare if the performance of our models are better than simply predicting the output to be the average of our test set. The true accuracy is given by: $\hat{Acc} = \hat{\theta} = \frac{m}{N}$ and is shown for the models in the box plot below:

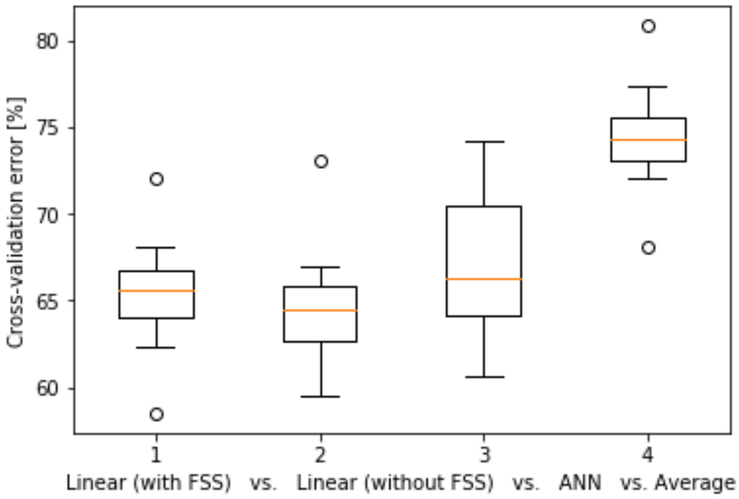

Figure 8

The linear regression without subset feature selection seems to be best performing model. This may also be the case in general. However, we still want to check if there is a significant difference between that model and the ANN. It can be argued that the two linear regression models should be compared since these are the best performing models. We do however think it is more interesting to compare the linear regression with the artificial neural network, since these models are based on a completely different training techniques.
Using formula, 1.4 we get an interval from -5.62 to 0.89 for the two models, and since zero is within the interval, there is no significant difference between the two. In fact, there is no significant difference between all the three models. They are however better than predicting the average output of the test set. This is confirmed when comparing the worst model (ANN) with the predicted average of the test set. Here the credibility interval goes from -11,72 to -3,43. Because zero is not within, it can be concluded that our models are better than using the predicted average.

2 Classification

In the following sections the data set will be classified based on our four different class definitions of salary. The salary data has been split into four quartiles, as seen in Table 3, which corresponds to a rough estimate of lower, lower middle, upper middle, and upper class. The division of these classes from a programming standpoint is flexible so analysis based on a more complicated division of wage classes is easily possible. For sake of this report and our limited computing resources the limit has been set at four classes.
So the task is to build a model that can consistently predict a person's income class based on information about that person, detailed and explained in Table 1. We have decided to consider Decision Tree, K-Nearest Neighbor (KNN), and Artificial Neural Network (ANN) based models to accomplish this task. To judge the accuracy of these different models against each other one can compare their generalization error. Through 2-layer, first 5-fold

Percentile	**Meaning**
$>Q_3$	Upper Class
Q_3	Upper Middle Class
Q_2	Lower Middle Class
Q_1	Lower Class

Table 3: Simplified Income Classes

then 10-fold, cross validation that error can be approximated as:

$$\hat{E}^{gen} = \sum_{i=1}^{K_1} \frac{|\mathcal{D}_i^{test}|}{N} E_i^{test} \tag{1}$$

The different generalization errors are show in Table 4. It is clear to see that ANN performed best, that being said, due to lack of computing resources and abundance of features the models ended up not being especially consistent or accurate in their predictions.

Model Type	$\hat{E}^{gen}$ [%]
Decision Tree (gini)	63.96
Decision Tree (entropy)	67.06
KNN	62.89
ANN	61.82

Table 4: Generalization Errors

2.1 Decision Tree

The gini and entropy based decision trees ended up having very similar generalization errors, with the decision tree model based on entropy being slightly better. The cross validation loop calculated that the entropy decision tree with a max depth of 3 lead to the best results. The error vs max depth of this fold is shown in Figure 9

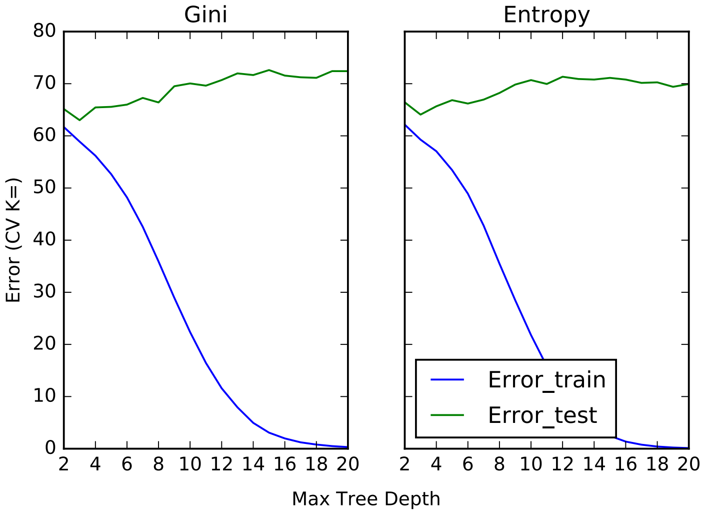

Figure 9: Test Error for Best Decision Tree Model Fold

The tree itself is shown in Figure 10

2.2 K-Nearest Neighbors

The number to be determined, in the case of a KNN type model, is the number of neighbors. The 2-layer cross validation found that a test error minimum is typically found by 27 nearest neighbors for this particular dataset. The best model that was found during this cross validation used the 27 nearest neighbors and had an unbiased test error of 60.76%. The test error plot for that specific fold is shown in Figure 11. This model being used to classify all of our data is shown in Figure 12.

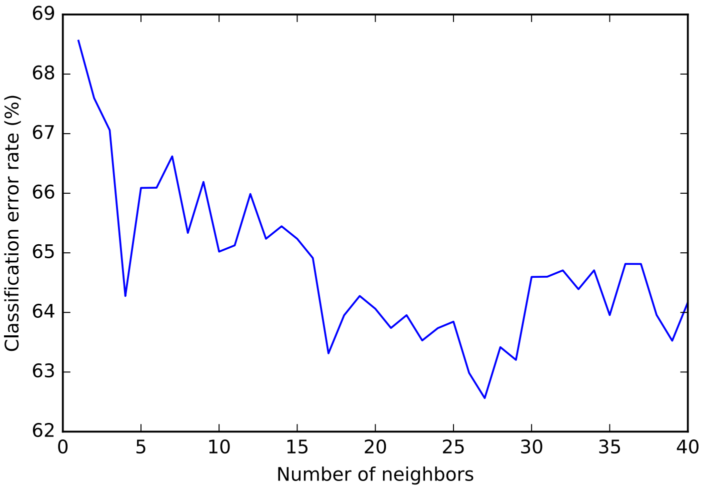

Figure 11: Test Error for Best KNN Model Fold

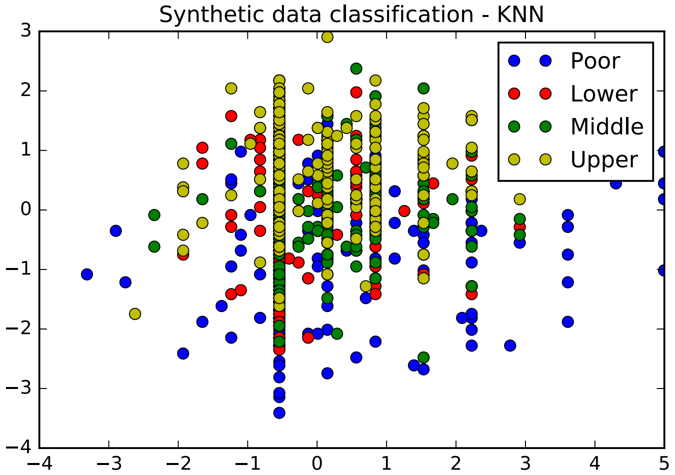

Figure 12: 17 Nearest Neighbor Model Classification

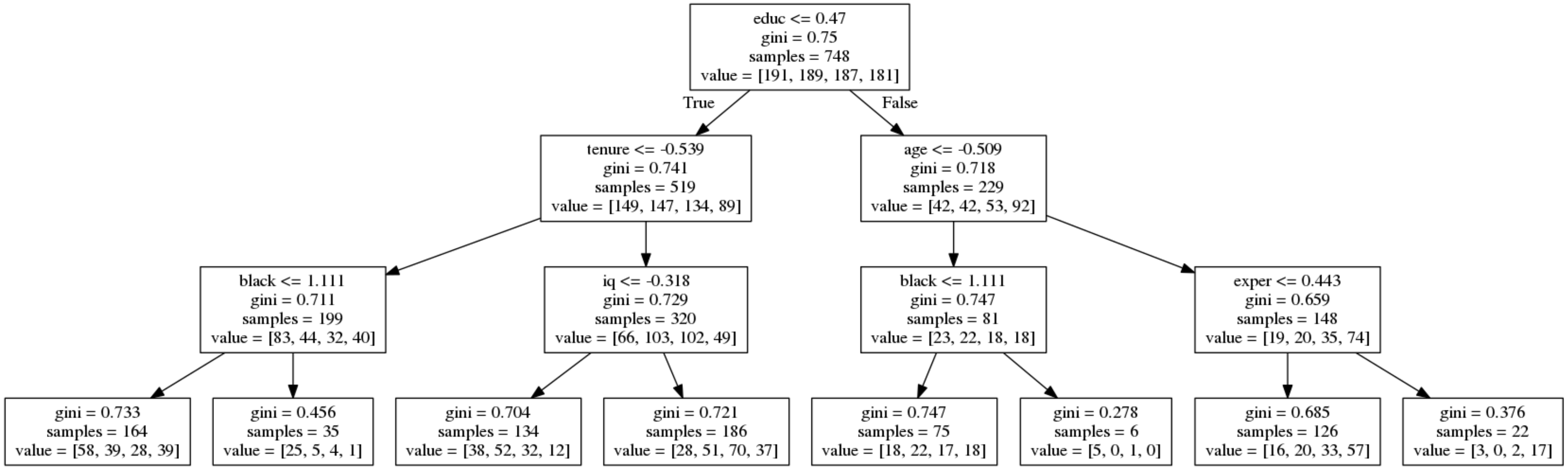

Figure 10: Best Gini Decision Tree wth Max Depth of 2

2.3 Artificial Neural Network

By generalization error, artificial neural networks prove to be the best model type for this set of data, though not beating K-Nearest Neighbor by much. The cross validation algorithm has found an artificial neural network with 3 hidden layers to be the best model for the training and test data. The test error plot for that model's specific fold is shown in Figure 13

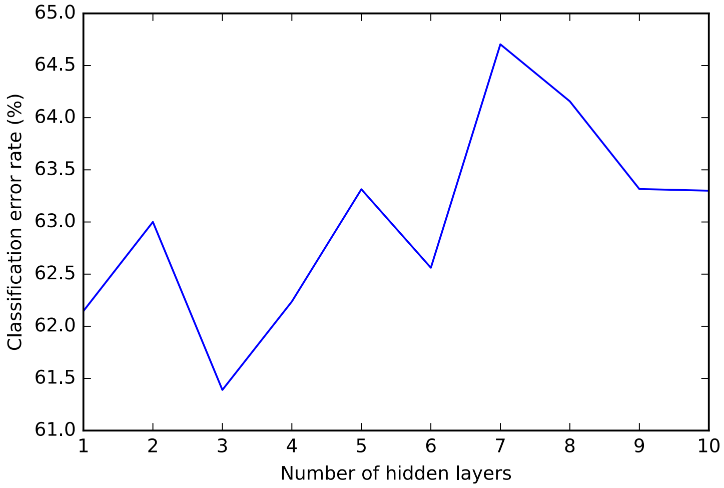

Figure 13: Test Error for Best ANN Model Fold

2.4 Model evaluation

Now that each model have been computed, we want to determine which model is the better. We will also check if these models are better than simply taking the most frequent class in the test data. The approach will be similar to the model evaluation in section 1.4. We therefore test the performance of the models on a new test set. 10-fold cross validation is once again used, making it a two layer cross validation process (in fact a three layer process). The models' performance are shown in the box plot below:

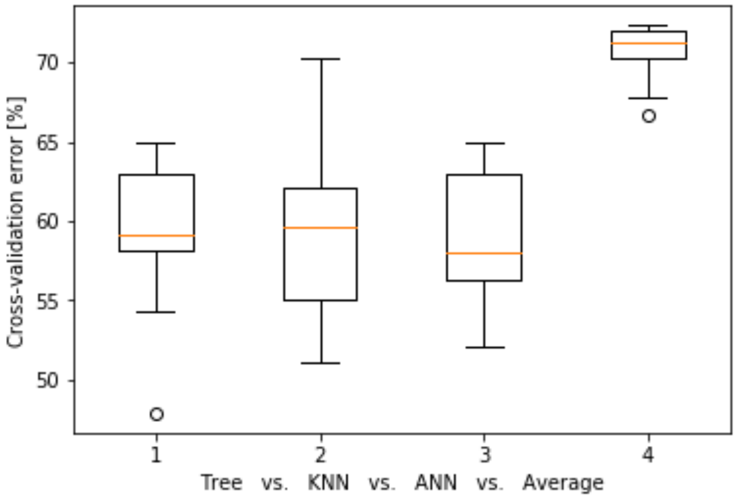

Figure 14

The box plot reveals that there is no big difference between the models. They are however better than predicting the class to be the most frequent class in the test set. As in section 1.4, we will use credibility intervals to compare the models, so that we with certainty can say that there is or is not a significant difference between the models. The matrix below shows the credibility interval when comparing all models to each other.

	Tree	**KNN**	**ANN**	**Average**
Tree	0	[-2.53 , 1.69]	[-2.44 , 2.45]	[-15.55 , -7.73]
KNN	[-2.53 , 1.69]	0	[-2.44 , 3.28]	[-15.48 , -6.97]
ANN	[-2.44 , 2.45]	[-2.44 , 3.28]	0	[-15.67 , -7.62]
Average	[-15.55 , -7.73]	[-15.48 , -6.97]	[-15.67 , -7.62]	0

Table 5: Credibility Intervals Between Models

Table 5 shows that zero is within all credibility intervals between the models, meaning that there is no model that performs significantly better than the others. All models are however better than classifying the the data according to the most frequent class in the test set.

Conclusion

As a conclusion to this paper, we can conclude that we were unable to find a model that performs significantly better than the others. This is true both for the regression part, and the classification part. All models do however perform significantly better than if the output was predicted to be the average of the test set, or the most frequent class. We are thus better off having these models than we were before doing this analysis. However, the performance of these models are still too poor to be used to draw any conclusions on how our features may influence the wage of a person. This paper ought therefore not be used to draw any scientific conclusion on how to predict wages of US citizens. The paper should rather be seen as an example of how selected machine learning models is applied on real life data.