Danmarks Tekniske Universitet

DTU

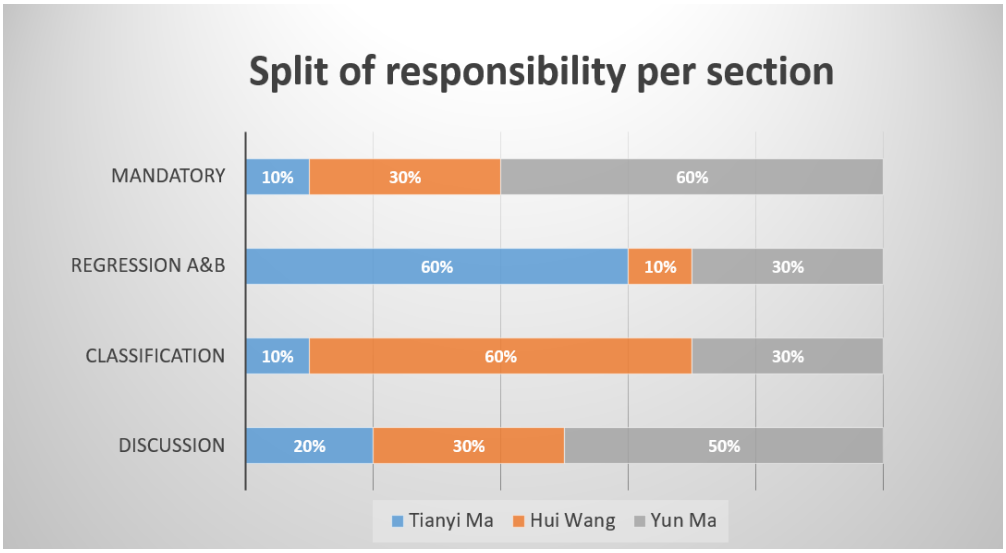
02450 Introduction of machine learning and data mining–Project 2

Authors

Tianyi Ma - s210316
Hui Wang - s210331
Yun Ma - s202707

April 19, 2021

Contents



1 Mandatory

In this part, we choose to resolve Question 1, 2, 4, 5, and 6.

Question 1, the answer is C. In order to obtain the ROC curve, we need to compute the false positive rate (FPR) and true positive rate (TPR). When we consider FPR = 0.5, the possible TPR of prediction A is 0.5, the possible TPR of prediction B is 0.5, 0.75 and 1, the possible TPR of prediction C is 0.25, 0.5 and 0.75, the possible TPR of prediction D is 0.25 and 0.5. So the prediction C corresponds to the ROC curve.

Question 2, the answer is C. To solve this question, we need to use the classification error function to derive the impurity according to the indication:

$$ClassError(v) = 1 - maxP(c|v) \tag{1}$$

There are totally $33+4+28+2+1+30+3+29+25 = 135$ samples in this question. When $x_7 = 2$, there is only 1 sample in class 2 and 0 sample for other classes, which means there is 1 sample in one branch and 134 samples in the other branch. The max $P(c|v)$ should be $\frac{134}{135}$, according to the classification error function above, the impurity should be $\frac{1}{135}$, namely 0.0074.

Question 4, the answer is D. From the structure of decision tree, the key step to solve this question should be step C. Because it is clear that if step C is true, the Congestion level 4 will be split out at once. When $b_1 \geq -0.16$, the step C is true, so the option should be D.

Question 5, the answer is C. The inner fold $K_2 = 4$, the outer fold $K_1 = 5$, there are 5 λ and 5 h for logistic regression and ANN. In addition, in each outer fold, one more time for calculating the generalization error for the optimal parameters is needed. Thus, the whole time is $4 \times 5 \times 5 \times 25 + 4 \times 5 \times 5 \times 9 + 5 \times (25 + 9) = 3570$.

Question 6, the answer is B. The first step of resolving this question is to put the value of b and ω_k into the given equation to compute the $\hat{y}_k$, for $k = 1, 2, 3$ respectively. Then we need to put the value of $\hat{y}_k$, for $k = 1, 2, 3$ into the given softmax function to calculate the $P(y = k|\hat{y})$, for $k = 1, 2, 3, 4$ respectively. According to the calculation of option B, the approximate probabilities of the 4 classes are 0.05, 0.06, 0.15, 0.73. The class $y = 4$ has the largest probability, so the option should be B.

2 Introduction

2.1 A recall of our dataset

There are severn attributes in our dataset:

1. X1: Transaction date.
2. X2: House age in year.
3. X3: The distance to the nearest MRT station in meter.
4. X4: The number of convenience stores in the living circle on foot
5. X5: Latitude of geographic coordinate in degree.
6. X6: Longitude of geographic coordinate in degree
7. Y: The house price per area[1]

Table 1: A demo for a piece of our dataset

X1	X2	X3	X4	X5	X6	Y
2012.916667	32.0	84.87882	10	24.98298	121.54024	37.9
2012.916667	19.5	306.59470	9	24.96903	121.53951	42.2
2013.583333	13.3	561.98450	5	24.98746	121.54391	47.3
2013.500000	13.3	561.98450	5	24.98746	121.54391	54.8
2012.833333	5.0	390.56840	5	24.97937	121.54245	43.1

2.2 Feature transformation

As shown of the Table 1, there is no need for us to apply one-of-K coding since all of our attributes are already nice integers or floats. We standardize the dataset by subtracting the mean and dividing by standard deviation for each attribute, thus each attribute has a mean of 0 and a variance of 1.

[1]The unit is 10000 New Taiwan Dollar/Ping, where Ping is a local area unit, 1 Ping = 3.3 square meter

3 Regression

3.1 Part A

In this part, we use X1, X2, X3, X4, X5 and X6 as features to predict the value of Y, the house price per area. The following linear regression model will be applied:

$$y_i = f(w_i, \omega) = \tilde{x_i}\omega \tag{2}$$

Based on the above linear regression model, we discuss how regularization parameter λ influence the performance of our regression model. According to equation 14.4 on the textbook, the relationship of the weights and λ is:

$$\omega = (\hat{X}^T\hat{X} + \lambda I)(\hat{X}^T\hat{y}) \tag{3}$$

When the λ becomes larger, the weights are getting smaller, the features would have a smaller influence on the model, so there is more likely to lead to high bias and low variance and vice versa. The selection of λ range:

$$\lambda \in [10^{-2} : 10^{8}]$$

The linear regression model will be trained on the dataset with 1 level cross validation (Algorithm 5 on textbook), and K-fold cross validation will be used. Since the dataset is small, $K = 5$ will be applied in this report. In each fold, the squared loss per observation will be computed based on different λ as the performance evaluation for both training set and test set. After that, 2 lists of average squared loss based on different λ of 5 folds will be returned (one for training errors and one for test errors). The λ corresponds to the least mean squared error among the test errors will be selected as the optimal λ.

The formula of squared loss per observation used in this project:

$$E = \frac{1}{N}\sum_{i=1}^{N}(y_i - \tilde{y_i})^2 \tag{4}$$

The correlation of errors and regularization factor λ can be shown in the figure:

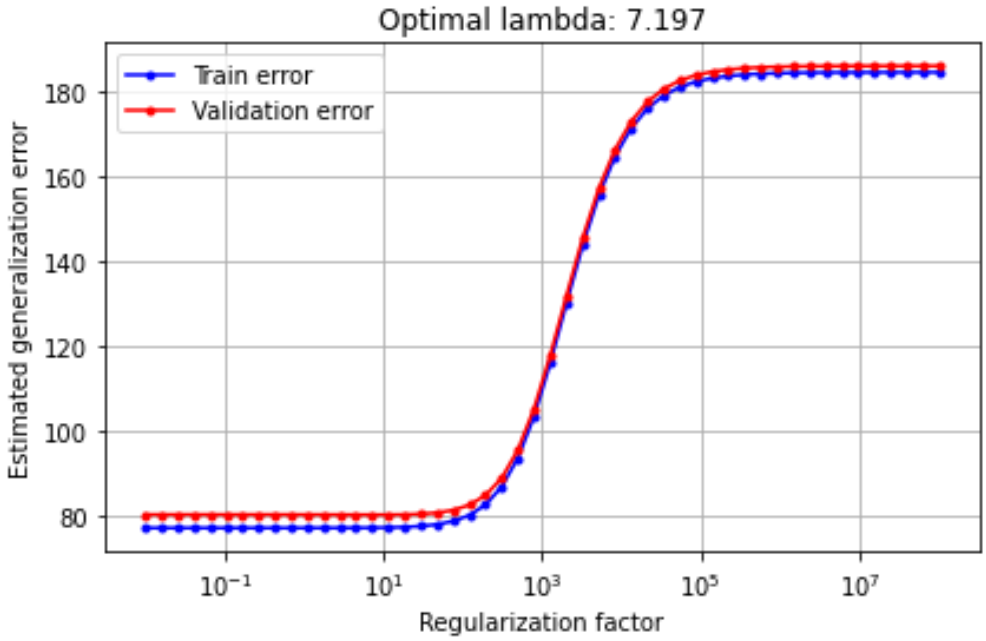


Figure 1: Squared error with different regularization factor

The weights for all the features with optimal regularization parameter $\lambda = 7.197$ can be found in the table 2. From the weight, X3 distance to the nearest MRT station has the biggest influence on the predicted value. It can explain the fact that in the last project, after PCA analyse, PC1 has a large coefficient for attributes X3 (distance to the nearest MRT station). What's more, X4 number of convenience stores, X2 house age, X5 latitude and X1 transaction date also have great influence on predicted value.

Table 2: the weights of optimal λ

	Parameter	Weight
1	Offset	37.976
2	X1 transaction data	1.412
3	X2 house age	-3.007
4	X3 distance to the nearest MRT station	-5.332
5	X4 number of convenience stores	3.34
6	X5 latitude	2.821
7	X6 longitude	0.061

3.2 Part B

3.2.1 Models and parameters

In this part, the comparison will be made between linear regression in the last part, ANN (Artificial Neural Network) and baseline. The 3 models are evaluated based on 2 level cross validation (Algorithm 6 on textbook). The inner folds aim to select the optimal complexity controlling parameter, λ for linear regression model and h for ANN model. The outer folds calculate the squared loss based on the optimal complexity controlling parameter selected by its inner folds. K_1 and K_2 are set to 5 because our dataset is relatively small.

In the **linear regression** model, the λ range used during this comparison:

$$\lambda \in [1:10]$$

In **ANN** model, there is only one hidden layer, and the range of hidden unit h is:

$$h \in [1:6]$$

The **baseline** is basically linear regression model with no features, it computes the mean of y on the training dataset, and predict every sample in the test dataset to that value.

The squared loss per selected parameter in each outer fold for three models can be seen in the table 3:

Table 3: Two-level CV summary for three regression

	Linear Regression		ANN		Baseline
i	λ_i^*	E_i^{test}	h_i^*	E_i^{test}	E_i^{test}
1	7.7	44.468	3	161.691	151.786
2	7.7	166.507	5	306.542	277.172
3	7.7	56.652	5	153.608	150.158
4	7.7	85.117	5	173.042	170.355
5	6.0	53.45	4	143.639	142.563

According to the result, the best test error of Linear regression model is 44.468, the best test error of ANN model is 143.595, the best test error of Baseline is 142.563. From figure 2, we can directly see the test error of three different models and linear regression seems to be the best model in this problem.

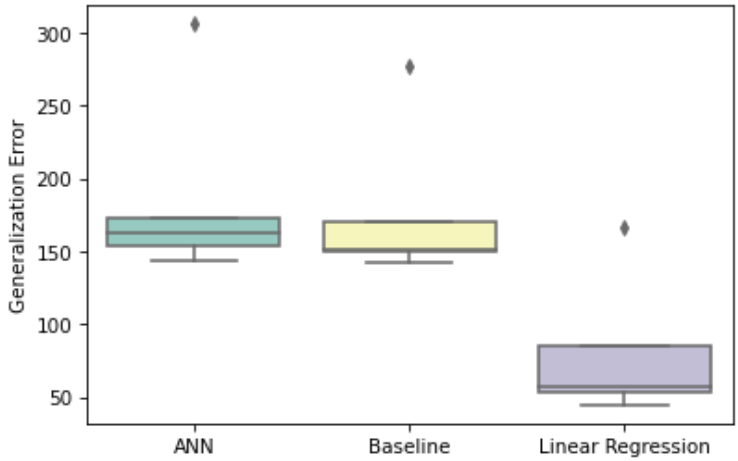


Figure 2: Boxplot of generalization errors from outer level of cross-validation for Linear Regression, ANN and the baseline model

According to the above results, it can be concluded that the linear regression has the best performance among the three models, the performance of ANN model and Baseline model are similar and are far worse than the other, the better one is baseline, The performance of ANN was different from what I expected, the more complex model turned out to be not as good as the simpler model. What's more, the best set of result in linear regression is 44.468, which λ in that folds is 7.7 and hide level is 3, while other two models' best set of result both in fold 5, the λ is 6.0 and hide level is 4. When λ is small, the weights are large indicating high variance and low bias. So, we can predict when we use flexible mode will have better test error with a low λ.

3.2.2 Statistical evaluation

We apply setup I for performance evaluation. In order to statistically compare the performance of the models a paired t-test was applied using $\alpha = 0.05$. We can see the result in table 4, linear regression is the best model and two p-values for the the paired t-test are both very low indicating strong evidence against the models being similar.

Table 4: Pair evaluation of the three models using Setup I

Model A	Model B	Confidence Interval	p-value
Baseline	ANN	[-18.36 -0.27]	0.02
Baseline	Linear Regression	[80.3 114.07]	$2 \cdot 10^{-26}$
Linear Regression	ANN	[-123.88 -89.13]	$4 \cdot 10^{-29}$

4 Classification

4.1 Models and parameters

The purpose of this section is to classify the Y (house price) given X1, X1, X3, X4, X5 and X6 into three classes, which are Low price, Medium price and High price. Thus our classification problem should be a multi-class classification problem. Logistic regression model, KNN (K-Nearest-Neighbors) model and baseline are used to accomplish the classification.

We define the labels according to the following criteria, where Q_1 represents 1/4 points of Y, Q_3 represents 3/4 points of Y.

Table 5: Criteria of house price classification

	$Y \leq Q_1(27.7)$	$Q_1(27.7) < Y < Q_3(46.6)$	$Y \geq Q_3(46.6)$
label	Low Price	Medium Price	High Price
y	0	1	2

In **Logistic regression** model, we use One-vs.-rest strategy for reducing the problem of multiclass classification to multiple binary classification problems. One-vs.-rest strategy involves training a single classifier per class, with the samples of that class as positive samples and all other samples as negatives [1]. The complexity controlling parameter is λ, the regularization factor. After several rounds of trial run, this range of λ is selected:

$$\lambda \in [-3, 2]$$

In **KNN** model, the the complexity controlling parameter is K. After several rounds of trial run, this range of K is selected:

$$K \in [1, 20]$$

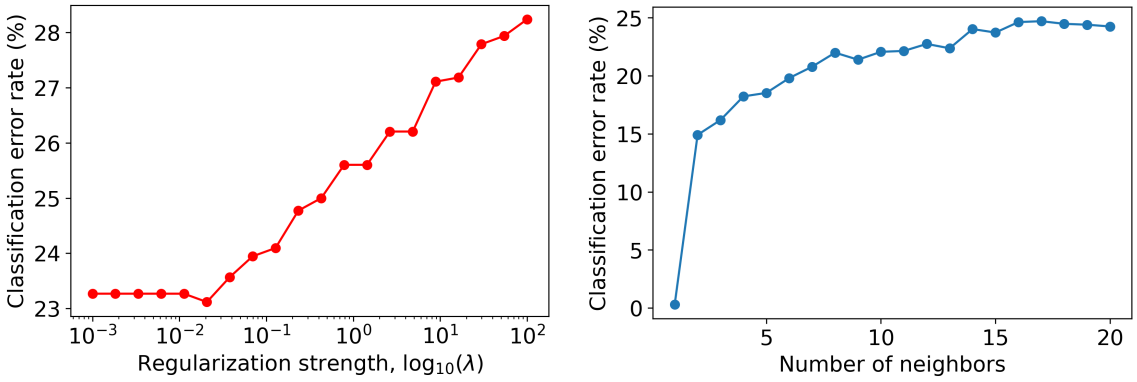
The **baseline** computes the largest class on the training data, and predict everything in the test-data as belonging to that class. That is logistic regression model with a bias term and no feature.

Two level K-fold cross validation is applied just the same as the previous section, $K_1 = K_2 = 5$. In the inner folds, the optimal complexity controlling parameter will selected, λ for logistic regression and K for KNN. In the each outer fold, the error rate will be calculated based on the selected parameter of its inner folds. The error rate calculation method:

$$E = \frac{Number\ of\ misclassified\ observations}{N^{test}} \tag{5}$$

From figure 3, the optimal complexity controlling parameter for each model can be found from each inner fold calculation. The parameter with minimum error will be selected

to train the outer level of the cross-validation. The optimal regularization parameters in figure 3 is $\lambda \approx 0.021$ and the optimal k-nearest neighbor is $K = 7$.



(a) Logistical Regression (b) KNN

Figure 3: Classification test error rate for complexity controlling parameters

The error rate per selected parameter in each outer fold for three models can be seen in table 6. From this table, we can clearly see that Baseline model has the worst performance with average classification error rate of $\approx$ 50%. Meanwhile, the logistical regression and KNN models perform better with an average classification error rate of $\approx$ 27% and $\approx$ 31%, respectively. In order to have a more intuitive description, a boxplot of all the errors from table 6 is plotted, as shown in figure 4.

Table 6: Two-level cross-validation table used to compare the three models in the classification problem.

Outer fold	Logistc Regression		KNN		Baseline
i	λ_i^*	E_i^{test}	k_i^*	E_i^{test}	E_i^{test}
1	0.0005	0.216867	1	0.216867	0.481928
2	0.0005	0.26506	1	0.26506	0.554217
3	0.0005	0.361446	1	0.361446	0.518072
4	0.0005	0.349398	1	0.349398	0.518072
5	0.0212	0.341463	1	0.341463	0.439024

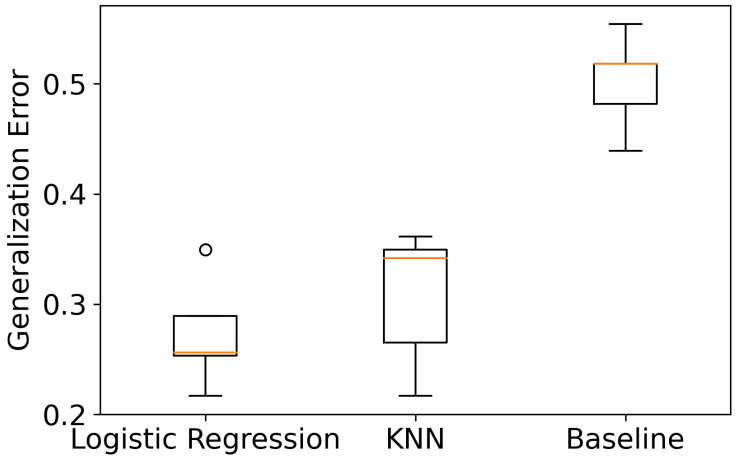


Figure 4: Boxplot of generalization errors from outer level of cross-validation for Logistic Regression, KNN and the baseline model

When the test errors are compared, baseline method is yielding more errors. However, we still can't have comments about which algorithm is performing better than others. Thus we will conduct a comparison of algorithms in the next section.

4.2 Statistical evaluation

We apply setup I to statistically evaluate if there is a significant performance difference between the three models. The McNemar's test was used to estimate the difference in performance $\theta = \theta_A - \theta_B$ between model $\mathcal{M}_A$ and model $\mathcal{M}_B$. If $\theta > 0$, then model $\mathcal{M}_A$ is preferable over model $\mathcal{M}_B$. The result of the three pairwise tests can be seen in table 7.

Table 7: Pairwise statistical evaluation of the three classification models

Model A	Model B	$\hat{\theta}$	Confidence Interval	p-value
Baseline	KNN	-0.2	[-0.26 -0.13]	$8 \cdot 10^{-8}$
Baseline	Logistic Regression	-0.23	[-0.29 -0.17]	$6 \cdot 10^{-12}$
KNN	Logistic Regression	-0.03	[-0.08 0.01]	0.206

From table 7 we can see that there is a relatively large difference in performance between Logistical regression / KNN and baseline model. The performance difference theta is estimated to be around -0.2. Meanwhile 0 is not in the confidence interval and p-value is very closed to zero, which is is very strong statistical evidence that the Logistical regression and KNN models are better than the baseline model.

For the KNN and Logistic regression model, the confidence interval contains 0 which shows a weak evidence towards Logistic Regression has higher accuracy than KNN. However, the p-value is relatively high, indicating the result is likely due to chance. Therefore, we do not have sufficient evidence to conclude Logistic regression is better than KNN.

5 Discussion

In linear regression, the biggest weights attribute of optimal λ is same as the result of PCA analyse in the last report, both of the two methods show the distance to the nearest MRT station is the biggest influence on predicted value. For the same regression problem with same features, different model can have significance performance differences. In the part b of regression, we have compared three models, linear regression, and baseline. Linear regression has the best performance and the performance of ANN is as poor as baseline , which is out of our expectation. Maybe we should try some feature transformation or feature selection before applying ANN.

In the classification problem, the logistical regression and KNN models perform better than baseline model with an average classification error rate of $\approx 27\%$ and $\approx 31\%$, respectively. However,due to the p-value is relatively high, we do not have sufficient evidence to conclude which one of these is best. There are only 414 observations in our dataset, may be it is the reason why there isn't a best model in the classification problem.

The dataset was analysed previously as a regression problem, which was also predicting the house price. That paper[2] pointed to use Quantitative Comparative Approach before applying machine learning methods. The machine learning methods discussed in this paper are linear model, quadratic model, logarithmic model, exponential model and exponential growth model. According to the paper, the performance improved significantly by applying Quantitative Comparative Approach mentioned compared with our models.

References

[1] "One-vs.-rest." `https://en.wikipedia.org/wiki/Multiclass_classification`.

[2] I.-C. Yeh and T.-K. Hsu, "Building real estate valuation models with comparative approach through case-based reasoning," *Applied Soft Computing*, vol. 65, pp. 260–271, 2018.