Logistic Regression Homework

Alireza Afzal Aghaei

April 6, 2021

Abstract

the second exercise of the data mining course contains four different sections. The first section requires us to implement K-fold cross validation along with ridge and lasso regression on the Immoscout24 dataset. Section 2 questions and some of questions of section 3 and 4 are related to theoretical machine learning subjects. Section three wants to implement a logistic regression model to solve a mobile price classification task. The final section, asks some question about section three. In this report we answer these questions.

1 Section 1

In the previous exorcise we reported the mean absolute error of the living space regression task in four different cases: small houses, big houses, average of two previous cases and total dataset. Table 1 recalls the obtained results. In the table 2 we reported the cross validation accuracy of this problem using linear, lasso and ridge regressions. It is shown that adding the regularization parameter reduced our prediction accuracy. Also, Lasso regression is usually a weaker model than Ridge regression.

2 Section 2

Here we will answer the second section questions:

1, 2) In the linear regression model, we try to learn the coordinates of a hyperplane in which the error minimize. This model does not worry about the magnitude of the hyperplane's normal vector norm. Increasing the normal vector norm leads to some problems such as numerical instability, overfitting, etc. To handle this problem scientists suggested to impose a regularization term to the model's loss function. This term which is usually is a function of weight (coefficients) vector, prevents the mentioned problem. The choice of regularization functions leads to different models. For example, Ridge regression model uses

House Type	Mean Absolute Error	Mean Squared Error	Mean Epsilon Insensitive Error	Accuracy
Small	3.24	17.38	2.35	88.48%
Big	4.1	24.83	3.17	99.03%
Mean	3.67	21.11	2.76	93.76%
All	3.81	22.28	2.89	95.01%

Table 1: Approximation Error of living space feature

Model	α	**Fold Number**	**MAE (Var)**	**MSE (Var)**	**Accuracy (Var)**
LinearRegression	-	5	3.84(0.03)	23.22(1.13)	95.04(0.11)
Ridge	1	5	6.79(0.03)	78.54(1.05)	77.05(0.31)
Lasso	1	5	16.61(0.21)	458.84(10.75)	43.09(0.53)
Ridge	0.1	5	5.38(0.04)	49.51(0.94)	85.33(0.36)
Lasso	0.1	5	10.04(0.10)	171.51(3.38)	61.32(0.27)
Ridge	0.01	5	4.17(0.03)	27.79(0.46)	93.15(0.26)
Lasso	0.01	5	8.10(0.04)	108.49(1.21)	69.96(0.34)
Ridge	0.001	5	4.00(0.02)	24.05(0.27)	94.71(0.11)
Lasso	0.001	5	6.94(0.03)	81.34(1.00)	76.12(0.26)
Ridge	0.0001	5	3.97(0.02)	23.66(0.16)	94.86(0.13)
Lasso	0.0001	5	4.51(0.04)	33.75(0.63)	90.86(0.33)
LinearRegression	-	10	3.84(0.03)	22.98(0.94)	95.07(0.17)
Ridge	1	10	6.75(0.05)	77.61(1.37)	77.29(0.44)
Lasso	1	10	16.60(0.23)	458.80(11.14)	43.01(0.60)
Ridge	0.1	10	5.29(0.05)	47.82(1.02)	85.91(0.40)
Lasso	0.1	10	10.04(0.10)	171.51(3.18)	61.32(0.36)
Ridge	0.01	10	4.14(0.03)	27.31(0.45)	93.34(0.25)
Lasso	0.01	10	8.10(0.05)	108.46(1.28)	69.98(0.38)
Ridge	0.001	10	3.99(0.02)	23.99(0.30)	94.74(0.11)
Lasso	0.001	10	6.94(0.05)	81.32(1.33)	76.12(0.39)
Ridge	0.0001	10	3.97(0.02)	23.65(0.17)	94.87(0.14)
Lasso	0.0001	10	4.51(0.05)	33.74(0.67)	90.86(0.33)

Table 2: Approximation Error of living space feature for different values of regularization coefficient α on 5 and 10-fold cross-validation

L2-norm as the penalty term, Lasso regression imposes the L1-norm, and ElasticNet model uses both L1 and L2 norms. This term usually appears along with a regularization coefficient. This coefficient trades-off between model simplicity and fitting data. Many researches developed methods to find the best regularization term for their problem. Trial & test, L-curve, grid search or using meth-heuristic algorithms are some of well-known methods.

3, 4) Increasing number of folds in K-fold cross validation increases the average accuracy variance. Leave One Out Cross Validation (LOOCV) is an special case of KFold cross validation in which K equal to the number of samples. In this method every sample would be tested singly. For more detailed information about the variance of K-fold cross validation, we refer you to this link.

5) In cross validation we divide the data randomly into kfold and it helps in overfitting, but this approach has its drawback. As it uses random samples so some sample produces major error. In order to minimize CV has techniques but its not so powerful with classification problems. Bootstrap helps in this, it improves the error from its own sample check. The difference between CV and bootstrapping comes down to variance and bias (as usual). CV tends to be less biased but K-fold CV has fairly large variance. On the other hand, bootstrapping tends to drastically reduce the variance but gives more biased results (they tend to be pessimistic). Other bootstrapping methods have been adapted to deal with the bootstrap bias. For large sample sizes, the variance issues become less important and the computational part is more of an issues.

6) 5x2 CV refers to a 5 times repeated 2-Fold cross validation. it popularised by the paper Approximate statistical tests for comparing supervised classification learning algorithms by Dietterich as a way of obtaining not only a good estimate of the generalisation error but also a good estimate of the variance of that error.

7) The question does not clarified what does rank means. If rank refers to the model complexity, i.e. number of layers in a multi-layer neural network, finding the rank using Elbow method does not yield any useful information. the lower rank model usually have a high bias.

3 Section 3

In this section we describe the mobile price classification dataset, then we propose our preprocessing method and finally report the obtained results. Decision theory explanation for our model is also added to the end of section.

Column Name	Column Description
battery_power	Total energy a battery can store in one time measured in mAh
blue	Has bluetooth or not
clock_speed	speed at which microprocessor executes instructions
dual_sim	Has dual sim support or not
fc	Front Camera mega pixels
four_g	Has 4G or not
int_memory	Internal Memory in Gigabytes
m_dep	Mobile Depth in cm
mobile_wt	Weight of mobile phone
n_cores	Number of cores of processor
pc	Primary Camera mega pixels
px_height	Pixel Resolution Height
px_width	Pixel Resolution Width
ram	Random Access Memory in Megabytes
sc_h	Screen Height of mobile in cm
sc_w	Screen Width of mobile in cm
talk_time	longest time that a single battery charge will last when you are
three_g	Has 3G or not
touch_screen	Has touch screen or not
wifi	Has wifi or not
price_range	This is the target variable with value of 0 (low cost), 1 (medium cost), 2 (high cost) and 3 (very high cost).

Table 3: Column descriptions

3.1 Data-set Definition

The dataset in this section defines a mobile phone classification task. The dataset contains some features of mobile devices such as number of cores, display size, etc. and defines the price level of that phone by numbers 0 to 3 in which 0 means cheapest and 3 means a very expensive phone.

The complete description of the dataset is reported in table 3.

3.2 Preprocessing

Although it seems there is many noise data in the dataset (high level phones without camera, zero-pixel resolution phones, zero-with display size etc.), we just remove 2 phones with zero-pixel resolution. The modification of the dataset is as follows:

-	train score	test score	precision	recall	f1 score
mean	0.986987	0.977477	0.977824	0.977469	0.977464
std	0.001875	0.009508	0.009429	0.009493	0.009477
min	0.984983	0.960000	0.960769	0.960000	0.960192
25%	0.985685	0.971137	0.971303	0.971173	0.971005
50%	0.986656	0.982462	0.982739	0.982398	0.982443
75%	0.988042	0.985000	0.985261	0.985000	0.984970
max	0.991101	0.985000	0.985483	0.985000	0.985049

Table 4: Results of mobile price classification

- Replace 3G and 4G columns with Network column which it's values are 0 if the phone has not 3G or 4G, 1 if phone has 3G network and 2 if phone's network is 4G.
- Add LCD size (diagonal size) based on screen width and height.
- Add PPI (Pixel Per Inch) based on display resolution and LCD size.
- Add Aspect Ratio column, based on screen width and height/
- Categorize battery capacity into five categories

The final dataset has 23 features with one dependent variable price range. It's worth to note that removing 2 zero-pixel phones, imbalances the dataset a bit. We handle this problem using class weight feature of sklearn's logistic regression class. There are some other ways to handle this problem. We refer the reader to this link for the explanation of other methods.

3.3 Results

In the table 4 we reported the results of fitting this data on a logistic regression model with 10-Fold cross validation.

Choosing the last Fold fitted model, we plotted the confusion matrix in the figure 1. As you can see, none of the incorrect predictions are completely wrong! The model does not any low price prediction as a high cost phone. It only confused on some similar price range phones (4 devices are wrong in price ranges 1 and 2, 2 wrong predictions are between price ranges 2 and 3).

3.4 Decision theory

By analysing these wrong predictions we found that the probability of these predictions are very close. In other words, for a wrong predicted phone with class 1 as class 2, the probability of classes one and two are 0.49 and 0.51. this means that the model has doubt which one are the real class. If we add "I'm not sure" option to the model prediction function, i.e. if probability greater than 0.65 model definitely answers

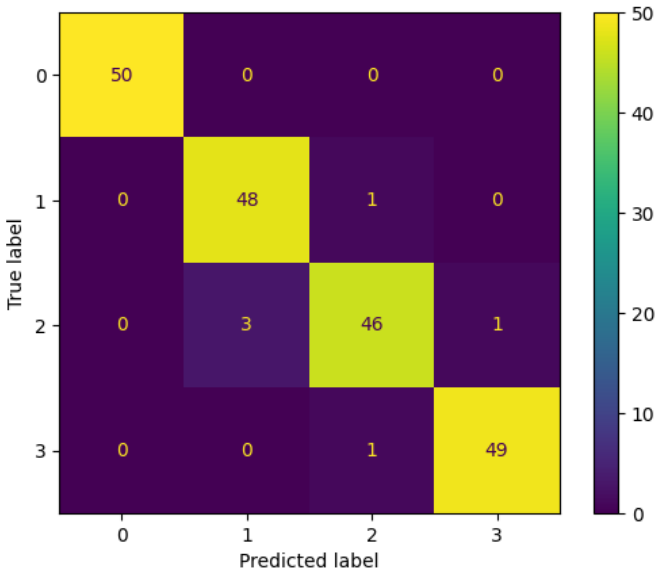

Figure 1: confusion matrix of model results

that case, otherwise model predicts and warns the user that I'm not sure about my prediction. Using this configuration, model reports 8 cases as "I'm not sure" and only predicts 3 wrong cases.

3.5 Modifying dataset

The third question of the exercise asks us to relabel the dataset in which all price ranges 1-3 transformed to 1. The new dataset is a binary classification problem. solving this, results test accuracy 0.9944 which is higher than previous data.

The question also requires that we use feature selection algorithms and report the results. Implementing this method, we saw that the he main dataset has higher accuracy than new sub-data set.

4 Section 4

In this section we should answer some questions and compares some results of previous section.

1) By using OVO & OVR methods, we can solve multi-class problems using logistic regression.

2) No. The results are almost identical.

3) Yes. Removing some features, reduces the model accuracy.

4) Feature selection methods are used to overcome the curse of dimensionality. Feature selection usually is not appropriate for There are many feature selection algorithms such as SelectKBest, SelectFromModel which select some important features based on another ML model, VarianceThreshold, etc.

5) A good replacement of the forward and backward feature selection methods are proposed in "Forward-Backward Selection with Early Dropping" paper.

6) This link explains the LDA method very good.