Econ 412 Project 2

Xinxian Li 005643690, Xinwei Hu 105641676,

Xiuqi Li 605638474, Zhiting Chen 505433562

l. Classification

Data Preparation

We used the dataset from Kaggle to analyze the relationship between the quality of wine and its influential factors. These include 2127 samples and 17 variables.

Variables	Definition
Fetal health	1 - Normal 2 - Suspect 3 - Pathological
Baseline value	Baseline Fetal Heart Rate (FHR)
accelerations	Number of accelerations per second
Fetal movement	Number of fetal movements per second
Uterine contractions	Number of uterine contractions per second
Light decelerations	Number of LDs per second
Prolongued_decelerations	Number of PDs per second
Abnormal short-term variability	Percentage of time with abnormal short-term variability
Mean value of short term variability	Mean value of short term variability
Percentage of time with abnormal long-term variability	Percentage of time with abnormal long-term variability
Mean value of long term variability	Mean value of long term variability

Histogram width	Width of the histogram made using all values from a record
Histogram number of peaks	Number of peaks in the exam histogram
Histogram number of zeroes	Number of zeroes in the exam histogram
Histogram mean	Hist mean
Histogram variance	Hist variance
Histogram tendency	Histogram trend

Table 1. Variables

The dependent variable of this project is Fetal health and it has 3 groups: 1 for health (normal), 2 for suspicion of illness, and 3 for illness. In order to test the performance of each model, we separated the dataset into training and test set randomly, where 70% data is included in training set, and 30% in testing set.

1.1. Logistic Regression

Firstly, we load the package of "nnet" and use the function "multinom" to construct a logistic regression, based on the training samples. Then, we input testing dataset into the model to see its performance. By constructing a confusion matrix, we found that precision for Normal, Suspect, and Pathological is 96%, 52% and 70% respectively, which show the percentage of data is true in group i when we label them as group i. And the recall for Normal, Suspect, and Pathological is 91%, 67% and 72% respectively. It indicates the percentage of data in group i is predicted as group i. In addition, by using (467+49+39)/638 we could calculate the accuracy for the whole testing dataset, which is 86.99% in our model.

	$\hat{y}$=Normal	$\hat{y}$=Suspect	$\hat{y}$=Pathological
y= Normal	467	38	6
y= Suspect	14	49	10
y=Pathological	7	8	39

Table 2. Result of Logistic Regression

What’s more, we use bootstrap to validate the accuracy of the model. We get the result of 0.8698, which is close to the model we build, and shows the good result for the model.

1.2. Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is commonly used as dimensionality reduction technique in the pre-processing step for pattern-classification and machine learning applications. The goal is to project a dataset onto a lower-dimensional space with good class-separate ability. In order to avoid overfitting (“curse of dimensionality”) , and reduce computational costs.

By using “lda” in the package of “MASS” in R, we could calculate the coefficients for each discrimination function. From the “proportion of trace”, LD1 explains 79.8% of the separation, but LD2 only explains 20.2%.

Through visualizing the scatterplot of the two discriminant functions, we noticed that although there are overlapping parts, the LDA model roughly distinguishes the three different groups in the dataset. Till now, the LDA model performs well.

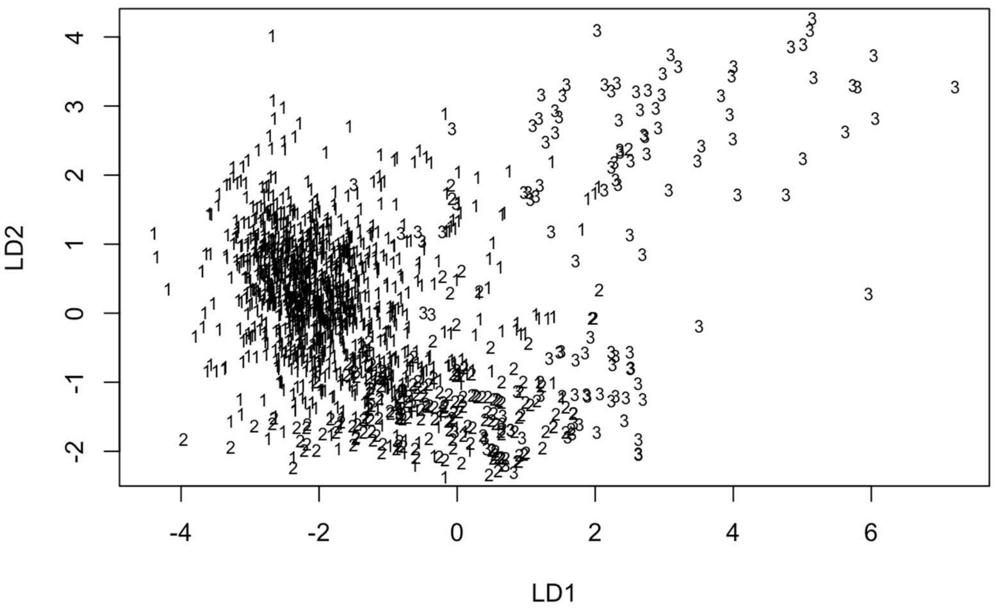

Figure 1. LDA Result

In addition, we could also check the accuracy of this model. By constructing the confusion matrix, we find the accuracy is 87.14% ((468+52+36)/638). The precision for each group is 95%, 54% and 65%, respectively. The recall for Normal, Suspect, and Pathological is 91%, 63% and 80%, respectively. Through bootstrap, we found the accuracy of the model is 87.15%, which is the close to the accuracy rate in the matrix.

	$\hat{y}$=Normal	$\hat{y}$=Suspect	$\hat{y}$=Pathological
y= Normal	468	38	5
y= Suspect	16	52	14
y=Pathological	4	5	36

Table 3. Result of LDA

Comparing with the testing accuracy in Logistic Regression, we found that the accuracy of LDA is slightly higher than the accuracy of Logistic Regression. We guess that it is because the original dataset has the features of normal distribution, in which case the LDA is more suitable.

1.3. Quadratic Discriminant Analysis

By using "qda" in package "MASS" in R, we could calculate the best coefficients for each class to separate the training samples. From the confusion matrix, we get the accuracy for this model is 85.4%. Meanwhile, the accuracy in testing samples of each group are 89%, 83% and 54%. The recalls are 96%, 55% and 68% respectively. Also, through bootstrap, we found the accuracy of the model is 85.4%, which is the same as the accuracy rate we calculate before. Because the accuracy for the testing samples is lower than LDA and Logistic Regression model, the decision boundary is more likely to be linear.

	$\hat{y}$=Normal	$\hat{y}$=Suspect	$\hat{y}$=Pathological
y= Normal	436	13	3
y= Suspect	41	79	22
y=Pathological	11	3	30

Table 4. Result of QDA

1.4. KNN

To pick up the most suitable k, we run an iteration to find out which one has the max accurate rate. We check the performance of different k (from 1 to 20) and found that the training set has the lowest error mean when k=1.

Therefore, we set up k=1 to calculate the accuracy of training and testing set. The accuracy of KNN model is 90.1%. Meanwhile, we also found that the precision of each group in testing set is 93%, 71% and 90%. The recall for each group is 95%, 67% and 85% respectively. Through bootstrap, we found the accuracy of the model is 90.1%, which is the same as the accuracy rate we calculate before.

Because the accuracy of testing samples for KNN (90.1%) is higher than Logistic Regression and LDA, we can infer that the decision boundary is highly non-linear and complicate.

	$\hat{y}$=Normal	$\hat{y}$=Suspect	$\hat{y}$=Pathological
y= Normal	464	21	3
y= Suspect	29	64	2
y=Pathological	4	4	47

Table 5. Result of KNN

Methods Comparation

For now, we can compare the results of different model. We list the outcome in the following table. We can see that the KNN model performs well over the others. Therefore, we may infer that the decision boundary is a non-linear boundary.

	Testing accuracy	Bootstrap accuracy
Logistic Regression	0.8699	0.8698
LDA	0.8714	0.8715

QDA	0.8542	0.8540
KNN	0.9012	0.9013

Table 6. Result Comparation

1.5.K-means

In order to choose best K for K-means model, we need to build different models and minimize total within-cluster sum of square (WCSS) to choose the best model. We draw the plot of number of clusters and WCSS.

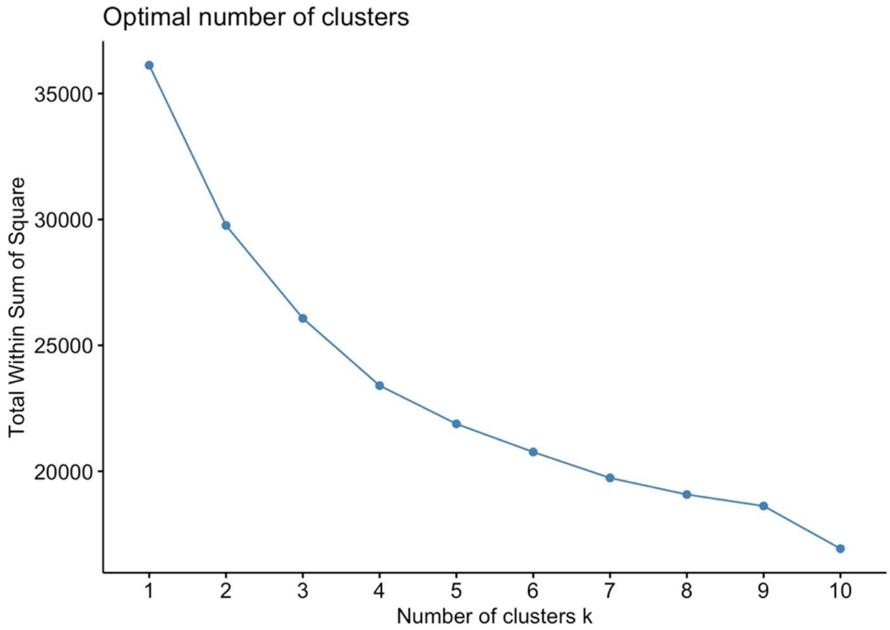

Figure 2. Optimal Number of Clusters in K-Means

From this picture, we can see the inflection point of the slope is when k=4. Therefore, we can conclude that the optimal number of clusters is 4. We build the model with 4 clusters and assign the label to each row in the data frame. This table shows the mean of each variable in 4 clusters.

Variables	k=1	k=2	k=3	k=4
Baseline value	0.7631609	-0.2756487	-0.4549192	-0.1388712
accelerations	-0.7587954	0.3013243	-0.6432496	0.2704412
Fetal movement	-0.1398127	0.1394277	0.7909481	-0.1191377

Uterine contractions	0.61310196	0.3977861	0.39115891	0.01030028
Light decelerations	-0.5880218	1.0137958	1.2920785	-0.4642516
Prolongued decelerations	-0.2686911	-0.101659	3.210655	-0.2515346
Abnormal short-term variability	1.0221962	-0.3919252	0.3919778	-0.3082001
Mean value of short term variability	-0.9716116	0.8653306	1.1589204	-0.1865497
Percentage of time with abnormal long-term variability	1.4867991	-0.4497321	-0.5115308	-0.3694134
Mean value of long term variability	0.24390362	0.07470818	0.8720176	0.27164208
Histogram width	-0.9495038	0.9686113	1.0746148	-0.2427121
Histogram number of peaks	-0.6552333	0.8678194	0.8869669	-0.2976619
Histogram number of zeroes	-0.3188742	0.5392849	0.2657932	-0.1874019
Histogram mean	0.5941373	-0.3260947	-2.129369	0.1926976
Histogram variance	-0.5896268	0.678233	2.0603195	-0.3808044
Histogram tendency	0.00854803	0.25532124	0.93060496	0.01238778
Fetal health	0.8580767	-0.4507903	2.3194167	-0.4590996

Table 7. Mean of Each Variable in K-Means

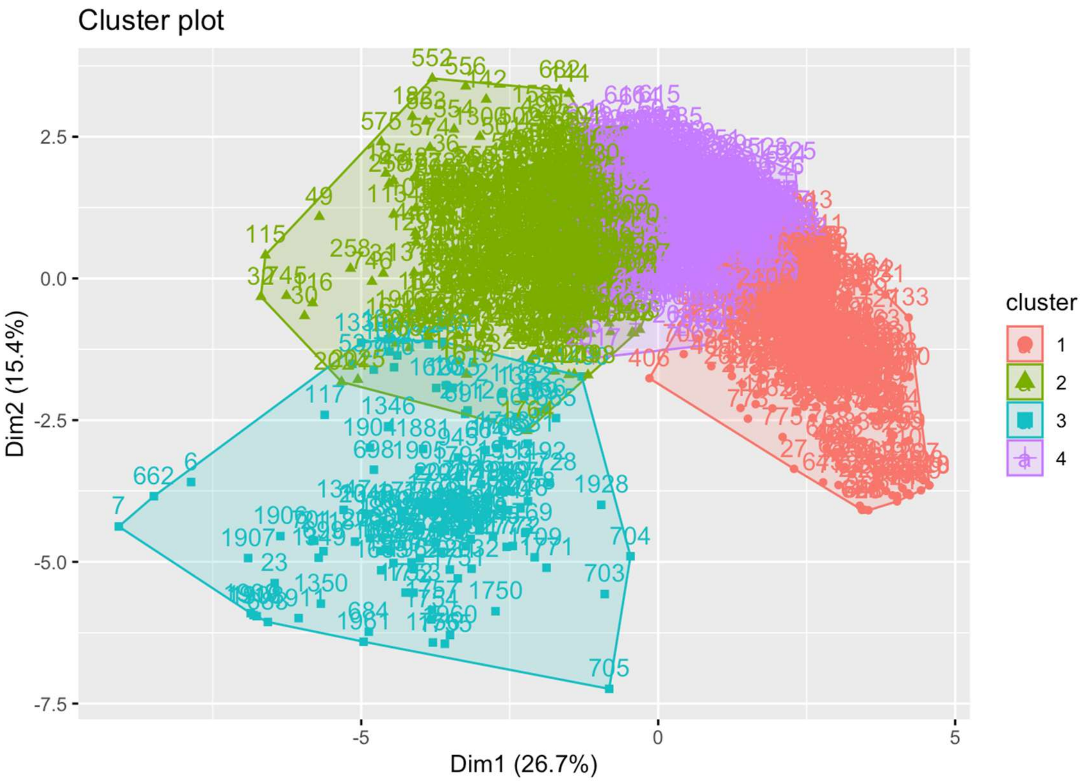

Figure 3. Cluster Plot of K-Means

From the figure 3, we can see that the four clusters are well divided. Although there is a little overlap between cluster 2 and cluster 4, on the whole, we can say that the performance of K-means model is good.

II. Regularization Data Preparation

This dataset is part of the data in real estate market from May to July 2014. The data frame has 1085 observations of house price on 18 variables, including at street names, number of bedrooms, numbers of bathrooms, price and so on. We will use 11 of them to evaluate the model.

Variable	Definition
date	The date when the house was sold
price	The property's sale price in dollars
bedrooms	Numbers of bedrooms in the house
bathrooms	Numbers of bathrooms in the house
sqft_living	Living size in square feet

sqft_lot	Lot size in square feet
floors	Numbers of floors in the house
condition	The living condition of the house
sqft_above	Lot size in square feet above basement level
sqft_basement	Basement size in square feet
yr	The house ages
renovated yr	The house age since last renovation

Table 8.

The dependent variable is the price of house and the distribution plot is as follows. It could be found that the distribution is right-skewed with extreme large values. Next, the data was separated into training sample and testing sample by 70% and 30% in order to evaluate the overfitting problem and also the bias-variance tradeoff.

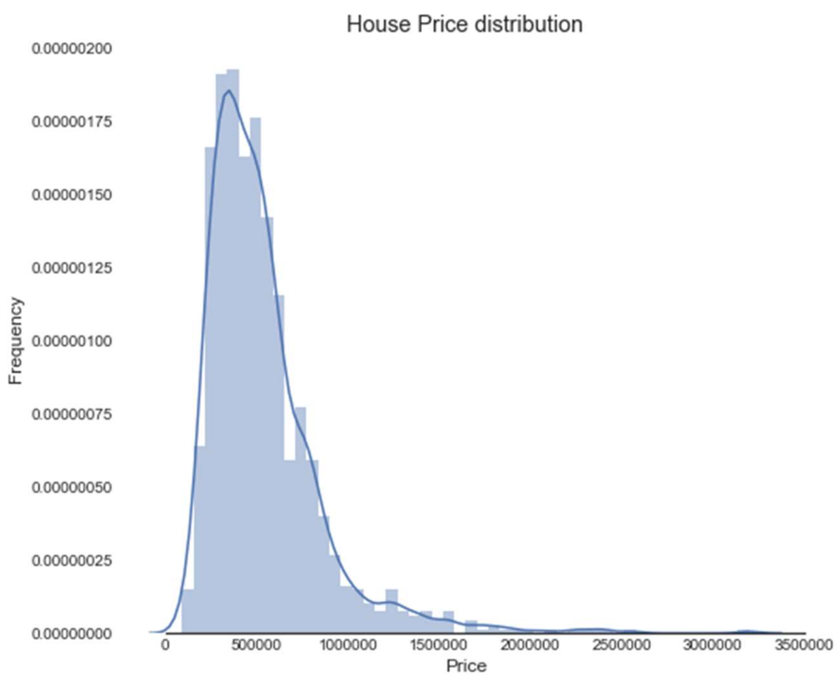

Figure 4. House Price Distribution

2.1. Lasso

Firstly, in order to choose the optimal scale of the regularization, we draw the plot of the relationship of coefficients weights and alpha. It could be found that when alpha is larger than 100, the penalty is too large, and coefficients would lose efficacy. The trade-off of bias and variance would be evident.

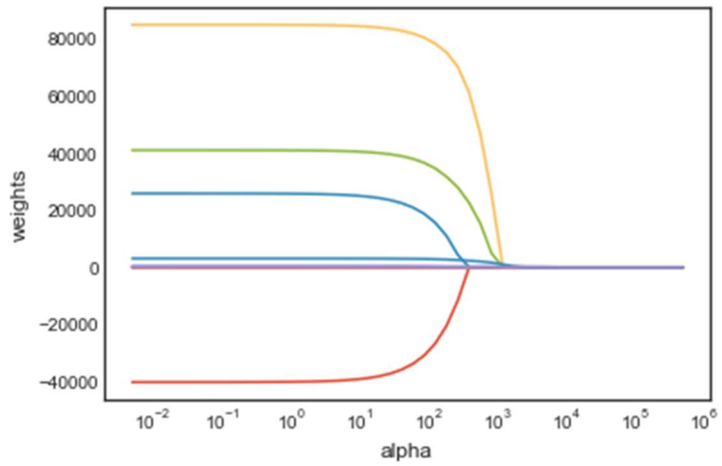

Figure 5.

Then, use the general Lasso method with penalty equals 1. The training score is about 0.59 and testing score is about 0.52, and mean square error is about 53089250448.1.

In order to evaluate the performance of Lasso model, we try the cross validation and the optimal penalty is about 133. Then training score is about 0.59 and testing score is about 0.51, which the difference is increased by CV method. The MSE also increases slightly to 53089339669.1. It's worth mentioning that the model includes 9 variables, instead of 10 variables as before. Because the scores are similar, there's little overfitting problem.

	training accuracy	**testing accuracy**	**MSE**
Lasso	0.589820589	0.517274748	53089250448
LassoCV	0.58783877	0.513972497	53089339669

Table 9.

From the plot below, we could find the differences between two models with different penalties and independent variables, like Lasso include the sqft_basement information into the model and Lasso with cross validation exclude them.

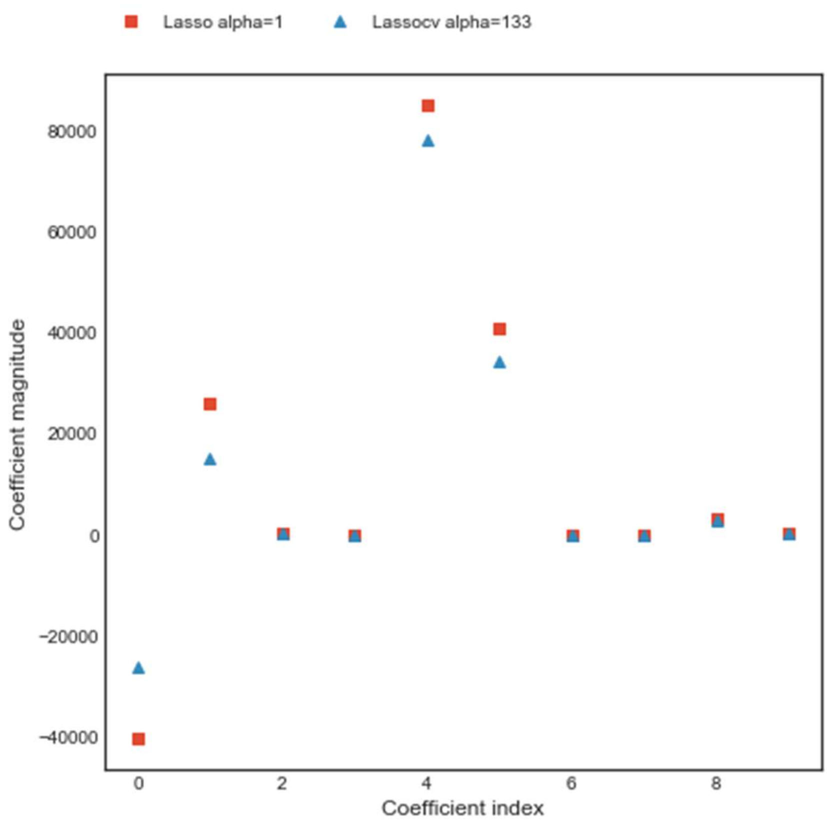

Figure 6.

2.2. Ridge

Similar procedure to check the penalty scale, and the coefficients would lose efficacy with alpha is about 1. Use the general Ridge method with penalty equals 1. The training score is about 0.59 and testing score is about 0.52, and mean square error is about 53079738090.8, which is similar with Lasso method.

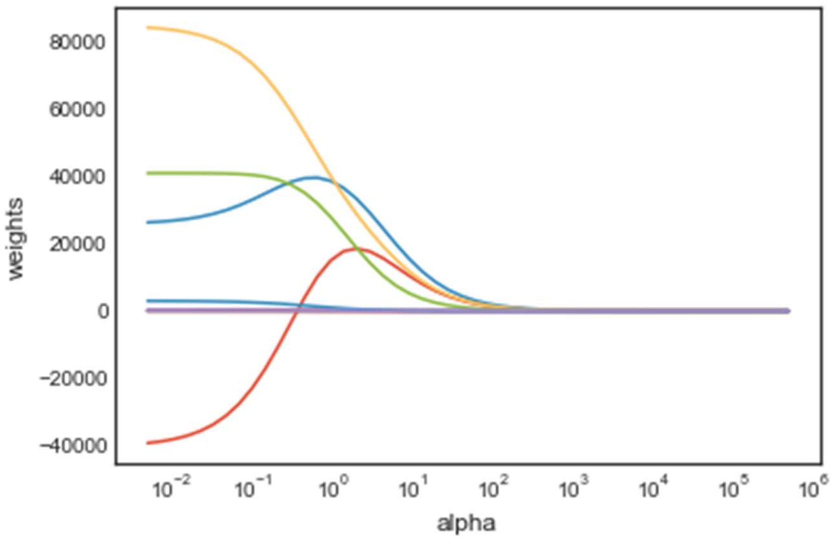

Figure 7.

Also try the cross validation and get the optimal penalty is about 0.07. Then training score is about 0.59 and testing score is about 0.51, slightly increase the difference by CV method. Both models include all the 10 variables and are not much overfitted.

	training accuracy	**testing accuracy**	**MSE**
Ridge	0.58981974	0.517361242	53079738091
RidgeCV	0.587334326	0.510201355	53136549734

Table 10.

The plot below shows the different coefficients with different penalties and independent variables. The Ridge model shrink the coefficient of sqft_lot and lower the effect of it to improve the fit, and it is omitted by the model with larger penalty.

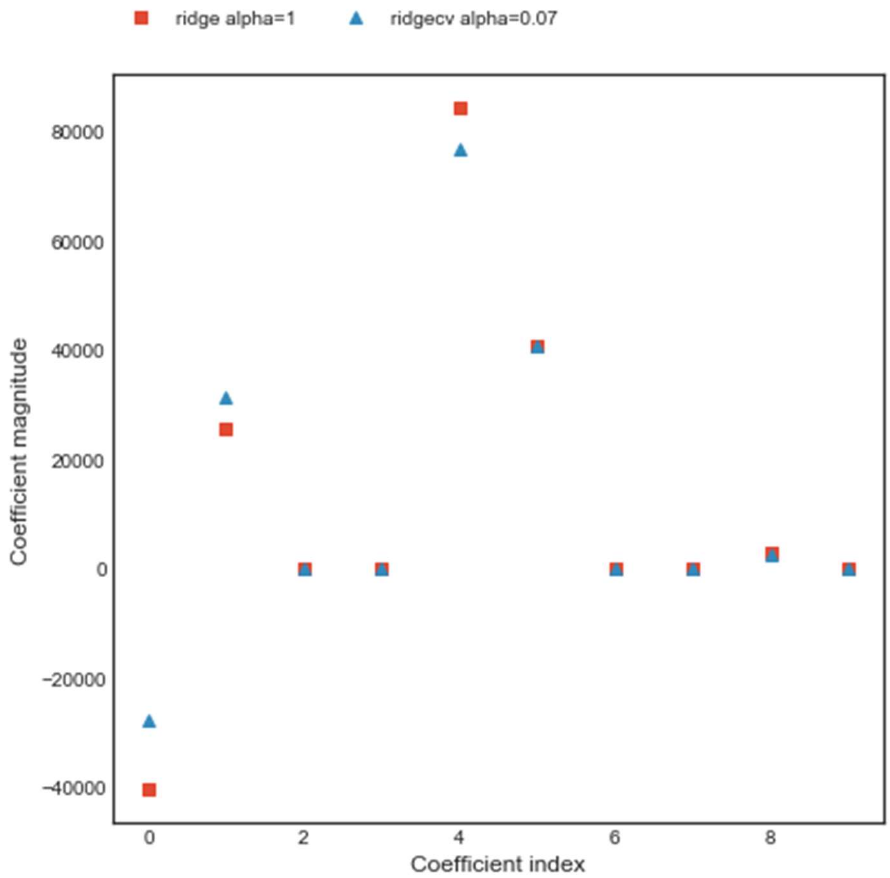

Figure 8.

2.3. Elastic Net

Use the general Elastic Net method with penalty equals 1, l1 ratio equals 0.5. The training score is about 0.58 and testing score is about 0.51, and mean square error is about 53370627815.

Also try the cross validation and get the optimal alpha is about 0.005 and l1 ratio is 0.5. Then training score is about 0.43 and testing score is about 0.37. Both models include all the 10 variables and have the overfitting problem.

	training accuracy	**testing accuracy**	**MSE**
Elastic Net	0.578	0.5147	53370627815
Elastic Net CV	0.4346	0.3653	69805689542

Table 11.

The plot below shows the coefficients magnitude with different penalties, like Elastic Net magnifies most of the effects of independent variables than Elastic Net CV method does.

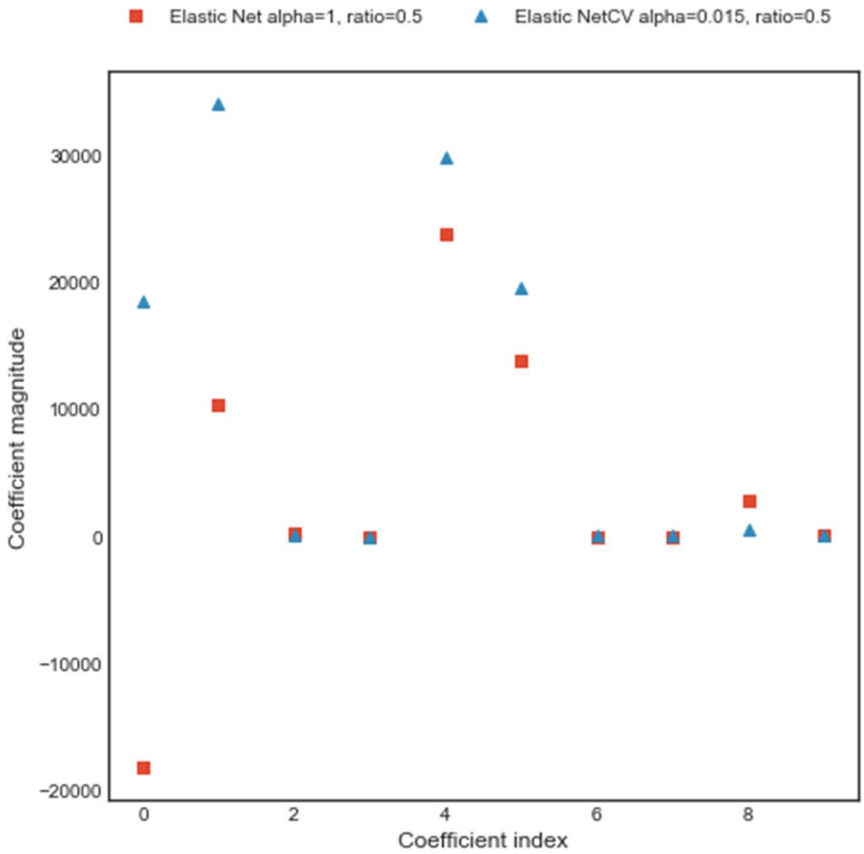

Figure 9.

2.4. PCA

Firstly, with 10-Fold Cross Validation, choose the number of principal components in the regression.

Calculate Mean Square Error using CV for the 10 principle components by adding one component at the time, and the result is shown below.

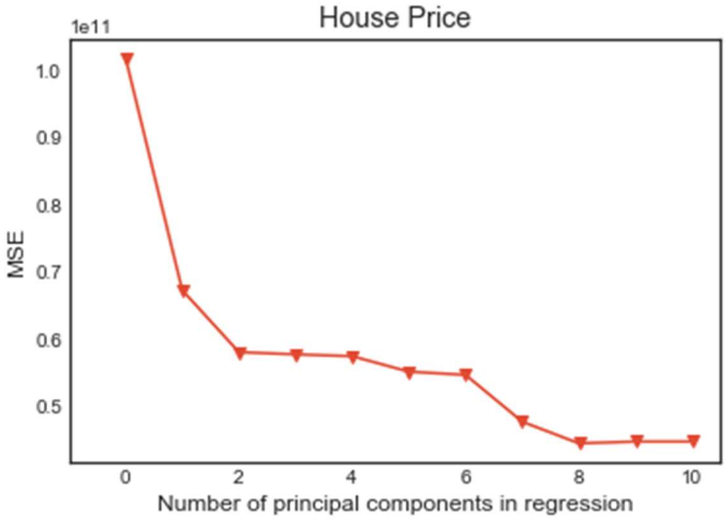

Figure 10.

The calculated regression scores with different number of predictors, shown as below. Combining the two indicators, MSE and score, we observe that the model with 2 components has the lowest MSE, its score also good. Comparatively, for the model with 2 independent variables, there is dramatical decrease in MSE but not significant increase in model score. As a result, we choose 8 predictors to do the following PCA regression.

Figure 11.

For the final regression with PCA, in training sample, MSE is 44482920464 and accuracy score is 0.59; and in testing sample, MSE is 54464675225 and accuracy score is 0.53. This means there is no overfitting problem and bias-variance trade-off is satisfied, even if the model is not accurate enough.

	training accuracy	**testing accuracy**
MSE	44482920464	54464675225
Accuracy	0.58982059	0.534450859

Table 12.

2.5. SVM

In Support Vector Machine regression, we applied four types of kernel, which are linear kernel, polynomial kernel, sigmoid kernel and RBF kernel.

	training accuracy	**testing accuracy**
Linear	0.54043661	0.484202599
Polynomial	0.508338139	0.462796007
RBF	-0.048108431	-0.058428986
Sigmoid	-0.048112605	-0.058428986

Table 13.

According to the training accuracy and testing accuracy scores of three methods, the SVM regression is also overfitting. One of the methods to solve this problem is adding more samples. Comparatively, linear SVM is more appropriate in this project.

Conclusion

According to the model evaluations and bias-variance tradeoff, principal components analysis shows the best performance among the five models, for the highest testing score and the smallest gap between training and testing scores. Therefore, the PCA method is the optimal model for the house price prediction without overfitting problem.

	training accuracy	**testing accuracy**
Lasso	0.589820589	0.517274748
Ridge	0.58981974	0.517361242
Elastic Net	0.578	0.5147
PCA	0.58982059	0.534450859
SVM(Linear)	0.54043661	0.484202599

Table 14.

III. Reference

https://www.kaggle.com/shree1992/housedata

https://www.kaggle.com/andrewmvd/fetal-health-classification?select=fetal_health.csv