Diamond Price Classification and Prediction

Runwei Wang, Menghan Lin, Hongyu Zhu, Fang Zhou

April 2019

1 Abstract

People are attracted by huge, and sparkling diamonds. Value of diamonds is decided by its rarity, which is usually assessed by 'The Four Cs': Cut, Color, Clarity, and Carat Weight. Besides that, each diamond's shape has different attributes that affects its price and quality grade. A certification is another observable variable that has an essential effect on the price of diamond. In our project, we are going to predict the price of diamond based on 4Cs and variables related to diamond's shape. This data set comes from Kaggle, which contains one response: price with 9 attributes of 53,920 diamonds. We implement three machine learning methods: KNN, random forest, and SVM to classify price of diamonds. We use 90% of data as our training set, and the rest as testing set. Model with best performance is random forest with all predictors, which eventually achieves a testing error rate of 2.97%.

2 Exploratory Data Analysis

2.1 Data Transformation and Summary Statistics

Before doing exploratory data analysis, we first split our response: price into three groups in order to apply classification methods.

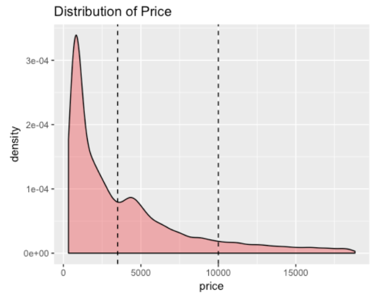


Figure 1: Density plot of price

Density plot is bi-modal, and we would use \$3,500 as one threshold between two modals, and \$10,000 as another cutoff to define expensive diamonds. Hence, there are three levels of price: low (price $\leq \$3,500$), medium(\$3,500 $<$ price $\leq$\$10,000) and high (price $>$ \$10,000).

Table 1: Summary Statistics - continuous variables

	carat	depth	table	x	y	z
min	0.20	43.00	43.00	3.73	3.71	1.07
max	5.01	79.00	95.00	10.74	58.90	31.80
median	0.70	61.80	57.00	5.70	5.71	3.53
sd	0.47	1.43	2.24	1.12	1.14	0.70

Among nine predictors, 6 are continuous and 3 are categorical(ordinal). It's clear that from the summary above that the continuous predictors are on different scales, so we may want to preprocess data by scaling all predictors in some of distance-based models.

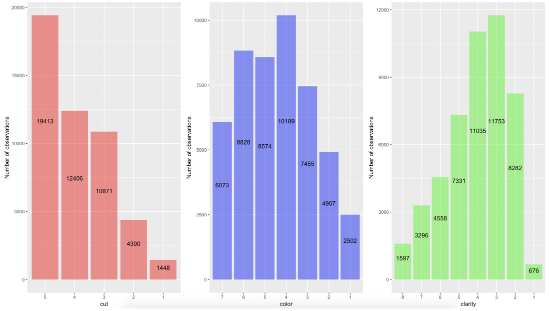


Figure 2: Number of observations in 3Cs

For three ordinal variables: cut, clarity and color, a higher numeric value corresponds to a better quality. Plot above shows the number of observations in each cut, clarity, and color group. From the left to the right, the quality becomes poorer. Except for the quality of cut, the majority of diamonds have a medium quality of color or clarity.

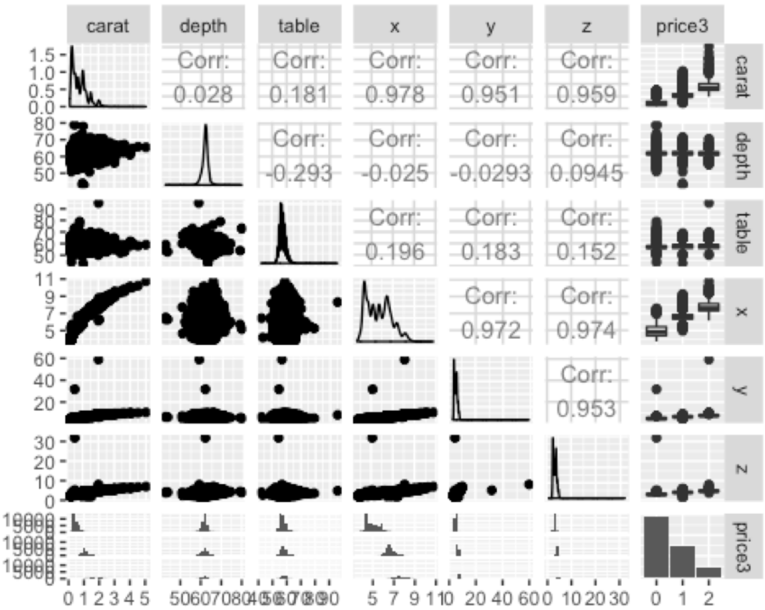


Figure 3: Summary plot

Indicated by the pair-wise scatter plot, diamond width, length, and depth (in mm) are highly correlated to each other as well as weight (carat). This phenomenon makes sense because if we assume all diamonds have same density, then weight equals to product of density and volume. Existence of outliers implying that there are diamonds which may have different shapes, and they may not follow the quadratic relationship. Plots on the bottom row are group-wise distributions, carat and length fall into various numeric ranges, while others are almost the same. Total numbers of observations are not balanced for each price group.

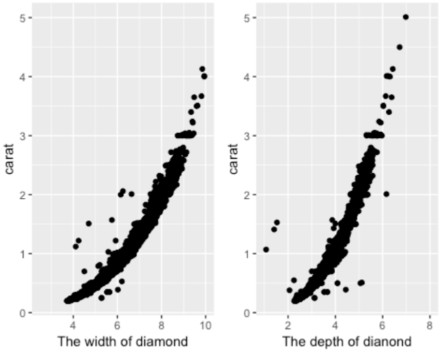


Figure 4: Relationship between width, depth and price

From figure 4, we see that after zooming in the plot to get rid of outliers, we conceive that the length and depth also show a quadratic relationship with price.

2.2 Balanced 4Cs? Hard!

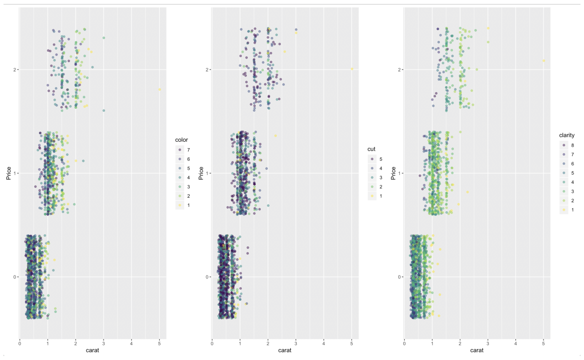


Figure 5: Relationship between width, depth and price

We randomly picked 2000 points from the training set, and as shown in the first and third plots, the heavier the diamond is, the poorer color it has. Whereas, cut doesn't follow this pattern, because it is controlled by people rather than nature.

3 Model Comparison

3.1 Support Vector Machines

3.1.1 Model Description

SVM is a distance-based classification method which aims to find the best classification function to distinguish between members of classes in the data. For a multi-class SVM learning task, currently there are two types of approaches. One is by constructing and combining several binary classifiers, while the other is by directly considering all data in one formulation. Here we use the first approach with “one vs rest” method to build a three-class SVM model.

3.1.2 Model Selection

SVM with linear, polynomial and radial kernels are performed, and the best model is selected via 10-fold cross validation.

Table 2: SVM Parameters Selected

	Cost	Degree	Gamma
linear	$0.01, 0.1, 1, 10, 100$		
polynomial	$0.01, 0.1, 1, 10, 100$	$1, 2, 3, 4, 5$	
radial	$0.01, 0.1, 1, 10, 100$		$1, 2, 3, 4, 5$

Firstly, we use all of nine variables based on the no-scaled data. The best model is with radial kernel cost = 1, gamma = 1, and achieves a smallest test error of 3.58%. Considering that SVM is a distance-based method, we then use the scaled data to build the model and see if it provides more accuracy. The smallest test error of scaled data is 3.56%, which is slightly better than the previous model. Finally, since the price of a diamond may only depend on the 4Cs metrics, we simplify our model by only using these four variables as predictors. The results are shown below. The smallest test error rate is 3.15%, which is the lowest between the of all SVM models based on scaled and non-scaled data set.

Table 3: SVM Error of Scaled Data by 4Cs

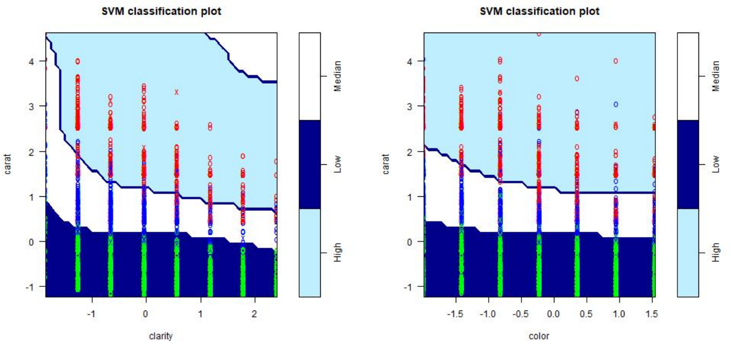
Kernel	Cost	Gamma	Degree	Training Error	Testing Error
linear	10	0.25	NA	4.45%	4.25%
polynomial	10	0.25	3	4.76%	4.25%
radial	1	1	NA	3.07%	3.15%

3.1.3 Results Analysis

Comparing the test errors of all models above, we choose radial kernel SVM model with cost = 1, gamma = 1 using scaled 4Cs as our best model. It has 5477 support vectors and the following function:

svm(formula = price $\sim$ carat + cut + color + clarity, scale(data), kernel = ”radial”, cost = 1, gamma = 1)

Applying our best model to test data, classification plots below show that our hyperplane classifies most data points correctly, though there are still some misclassified points, like some red points(high price) fall in the medium-price range. There are about 170 points misclassified, which is a relatively small compared to a total number of 5392 observations.



(a) Carat vs Clarity

(b) Carat vs Color

Figure 6: SVM Classification Plots

3.2 Random Forest

3.2.1 Model Description

Random forest is an ensemble method which corrects for decision trees' habit of overfitting, and improves bagging tree through de-correlating variables in each tree. Unlike other distance-based models, random forest inherits the advantages of tree structure and can be applied to variables of different types. Here we would build our model using all nine attributes.

3.2.2 Model Selection

In random forest, it is not necessary to use cross-validation because each tree is constructed using different bootstrap samples from the original data. Since we have 9 predictors in total, it's reasonable to randomly select around p/3 variables in each node. After comparing trees with 3, 4, and 5 variables, we get the minimal test error rate using 4 variables. Thus, here we set mtry = 4, and build 200 trees.

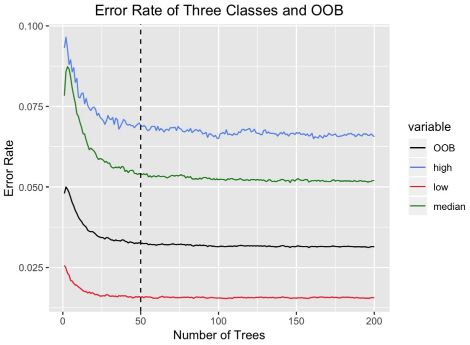


Figure 7: Error Plot of Three Classes and OOB Error

Figure above shows that with the growth of number of trees, all three classes' error rate and out-of-bag error decrease accordingly, and tend to be flat after we have 50 trees, while class 'high price'(blue line) has the highest error rate.

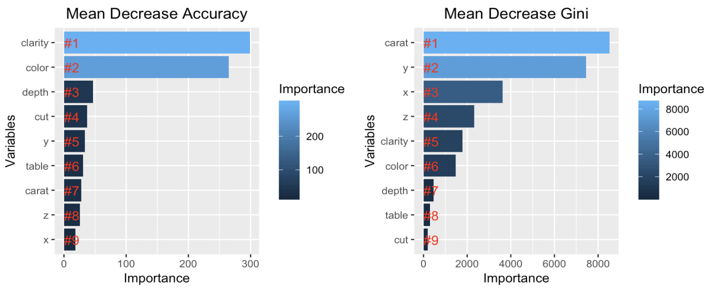
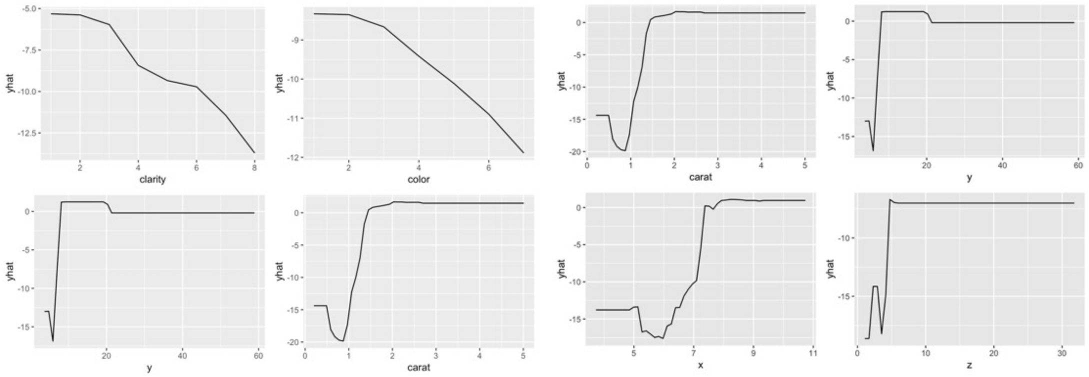


Figure 8: Importance Plot

Above shows variable importance plot of two measures: mean decrease accuracy and mean decrease Gini. From this plot we can see that these two methods give very different results. Hence, partial dependence plot is considered to determine which method is more reliable. We select the first four variables of both measures and their partial dependence plot show below:



(a) Mean Decrease Accuracy

(b) Mean Decrease Gini

Figure 9: Partial Dependence Plot

Partial dependence plots imply that mean decrease accuracy should be trusted because lines of mean decrease Gini are constant near zero, which shows that variables have no effect on the model.

3.2.3 Results Analysis

Achieved the minimum test error rate of 2.87%, we attain our best model as random forest with mtry = 4.

3.3 K Nearest Neighbors

3.3.1 Model Description

As the K-nearest neighbors (KNN) is to find the K closet neighbors to x_0 to determinate its class by the majority, it's a model based on our measurement of distance.

3.3.2 Model Selection

As mentioned above, we perform KNN on scaled data with all variables, comparing it to a model with only 4Cs as predictors.

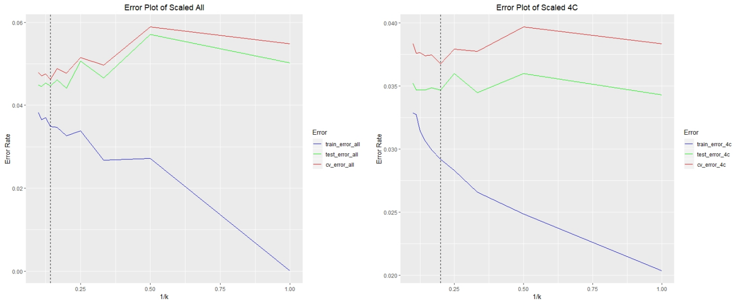


Figure 10: Error Rate Plots with All Variables and Only 4Cs

The red line represents cross validation error, we would select the model with smallest cross validation error rate, which is k=7 when using all variables, and k=5 when only using 4Cs. Specific numbers of error rate are shown below:

Table 4: Error Rate of KNN

	Training Error	Testing Error
Scaled ALL(K=7)	3.49%	4.47%
Scaled 4C(K=5)	2.92%	3.47%

3.3.3 Results Analysis

The best prediction using KNN is provided when setting k=5 and using only 4Cs as predictors of scaled data. Here CV-error rate is slightly greater than the test error rate which may due to the KNN overfitting problem.

4 Result Comparison and Analysis

4.1 Result Comparison

In general, as a consumer, we prefer diamonds with higher quality but lower price, so assuming our model is reasonable, you would like to buy diamonds that are under-priced and misclassified as the points in the red boxes in Figure11 plot (a). The red boxes include 62 green points which are predicted medium-priced but has low price, and

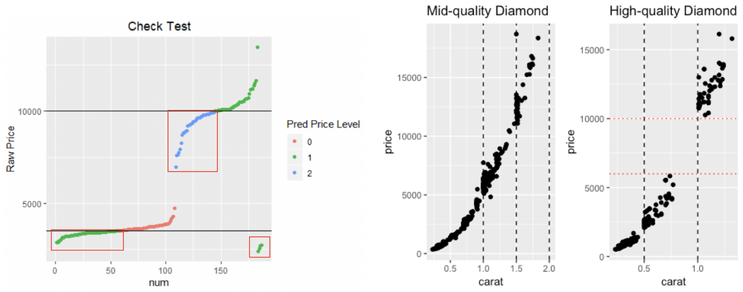
36 blue points which worth high price while having medium price. In general, 98 out of 187 (52.4%) diamonds in test data are underestimated.

Table 5: Error Rate Comparison

Model	Data set	Parameter	Training Error	Testing Error
SVM	Scaled 4Cs	radial kernel, cost = 1, gamma = 1	3.07%	3.15 %
Random Forest	All predictors	mtry = 4, ntree = 200	3.14%	2.87%
KNN	Scaled 4Cs	K = 5	2.92%	3.47%

To sum up, we implemented KNN, SVM, and random forest to perform diamond price prediction on about 50,000 observations. We apply them on 4Cs as predictors, all predictors of scaled, and non-scaled data. Generally, random forest worked the best with testing error rate 2.87%. The other methods also worked well since the difference among testing error was slight, specifically, only 10 more points are misclassified by the second best model: SVM with a radial kernel.

4.2 Further Improvement - Carat Premium and Price Jump



(a) Overprice

(b) Price Jump

Figure 11: Result Plots

The mid-quality diamond: clarity, color and cut to be 4, 4 and 3. The high-quality diamond: clarity, color and cut to be 6,7 and 5.

Let's go back and consider why random forest works better than other two methods. In reality, certification of a diamond indirectly influence its price. In the perspective of an amateur, a certified diamond would be much meaningful than a unidentified one, though they may only differ by 0.01 carat. Since 0.01 difference in carat is hard to see, points here cluster vertically and indicate price jump. The tree captures the phenomenon by using a range of values to classify rather than distance among observations, so it may indirectly consider the step-wise jump. More than that, effect of carat premium is interacted with its overall quality, and combined effect is spontaneously captured by each classification tree.

Above all, further improvements include: 1. Subgroup classification (by diamond shape) 2. Boosting to re-weight misclassified diamonds and 3. Adding new variables (carat ¿ 0.99, carat ¿ 1.49) to indicate these threshold and adding more weights.