SVM_Boston_Airbnb

Alex Wei

12/4/2021

```
dat <- read.csv('df.csv')
head(dat)
```

```
##   X host_response_rate host_acceptance_rate host_is_superhost
## 1 1               100                  100                0
## 2 2               100                   88                1
## 3 3               100                   50                0
## 4 4               100                  100                1
## 5 5               100                   95                1
## 6 6                98                   96                0
##   host_listings_count host_total_listings_count host_verifications
## 1                   1                         1                 7
## 2                   1                         1                 4
## 3                   1                         1                 3
## 4                   1                         1                 4
## 5                   2                         2                 4
## 6                   5                         5                 5
##   host_has_profile_pic host_identity_verified is_location_exact room_type
## 1                    1                      1                 1         1
## 2                    1                      1                 1         1
## 3                    1                      0                 0         1
## 4                    1                      1                 1         1
## 5                    1                      1                 1         1
## 6                    1                      1                 0         2
##   accommodates bathrooms bedrooms beds bed_type price security_deposit
## 1            2       1.0        1    1        1    65               95
## 2            2       1.0        1    1        1    65                0
## 3            4       1.0        1    2        1    75              100
## 4            2       1.5        1    2        1    79                0
## 5            2       1.0        1    1        1    75                0
## 6            3       1.0        1    2        1   100                0
##   cleaning_fee guests_included extra_people minimum_nights maximum_nights
## 1           10               0            0              2             15
## 2            0               1           20              3             45
## 3           50               2           25              1           1125
## 4           15               1            0              2             31
## 5           30               1            0              2           1125
## 6            0               1           25              1           1125
##   availability_30 availability_60 availability_90 availability_365
## 1              26              54              84              359
## 2              19              46              61              319
## 3               6              16              26               98
## 4              13              34              59              334
## 5               5              28              58               58
## 6              22              39              69              344
##   number_of_reviews review_scores_rating review_scores_accuracy
## 1                36                   94                     10
## 2                41                   98                     10
## 3                 1                  100                     10
## 4                29                   99                     10
## 5                 8                  100                     10
## 6                57                   90                     10
##   review_scores_cleanliness review_scores_checkin review_scores_communication
## 1                         9                    10                          10
## 2                         9                    10                          10
```

```
## 3                       10                   10                          10
## 4                       10                   10                          10
## 5                       10                   10                          10
## 6                       10                   10                          10
##   review_scores_location review_scores_value requires_license instant_bookable
## 1                      9                    9                0                1
## 2                      9                   10                0                0
## 3                     10                   10                0                0
## 4                      9                   10                0                0
## 5                      9                   10                0                0
## 6                      9                    9                0                0
##   cancellation_policy require_guest_profile_picture
## 1                   2                             0
## 2                   2                             1
## 3                   2                             0
## 4                   3                             0
## 5                   3                             0
## 6                   1                             0
##   require_guest_phone_verification calculated_host_listings_count
## 1                                0                              1
## 2                                0                              1
## 3                                0                              1
## 4                                0                              1
## 5                                0                              1
## 6                                0                              3
```

```
set.seed(42)
a <- dat[sample(1:nrow(dat)),]
train <- a[1:2440,]
test <- a[2440:3050,]
```

```
## Linear SVM

svmfit = svm(price~ host_is_superhost + host_verifications + room_type + accommodates + bathroom
s + bedrooms + beds + bed_type + number_of_reviews + review_scores_rating, data = train, kernel=
"linear")

# summary(svmfit)

y_pred <- predict(svmfit,test,decision.values = TRUE, probability = TRUE)
y_true <- test$price

RMSE(y_pred, y_true)
```

```
## [1] 77.0731
```

```
## Radial SVM

svmfit = svm(price~ host_is_superhost + host_verifications + room_type + accommodates + bathroom
s + bedrooms + beds + bed_type + number_of_reviews + review_scores_rating, data = train, kernel=
"radial")

# summary(svmfit)

y_pred <- predict(svmfit,test,decision.values = TRUE, probability = TRUE)
y_true <- test$price

RMSE(y_pred, y_true)
```

```
## [1] 74.8069
```

```
## Sigmoid SVM

svmfit = svm(price~ host_is_superhost + host_verifications + room_type + accommodates + bathroom
s + bedrooms + beds + bed_type + number_of_reviews + review_scores_rating, data = train, kernel=
"sigmoid", scale = FALSE)

# summary(svmfit)

y_pred <- predict(svmfit,test,decision.values = TRUE, probability = TRUE)
y_true <- test$price

RMSE(y_pred, y_true)
```

```
## [1] 109.4618
```

Analysis

Support Vector Machine is a type of supervised learning used for classification, regression and outliers detection. The reason why we chose to use support vector machines here is that SVM is Effective in high dimensional spaces, and here in the Boston Airbnb price prediction model, there are relatively large number of variate.

The train set and test test are separate based on the ratio 8:2, which 80% of the data was randomly assigned as the train data set, and 20% of the data was randomly assigned as the test data set.

The variables we chose to use in the support vector machine are: host_is_superhost, host_verifications, room_type, accommodates, bathrooms, bedrooms, beds, bed_type, number_of_reviews, and review_scores_rating. Since these variables make more sense when predicting the price, and when adding more variables, the result of svm worse off.

Here, we utilized three kernels in the Support Vector Machines, which are Linear, Radial, and Sigmoid kernel. The error measurement we use to determine the effectiveness of the model is Root Mean Squared Error (RMSE), the best performed model based on RMSE is the Radial kernel Support Vector Machine Regression Model, which has the RMSE of 74.8.

The result of the support vector machine here in predicting boston airbnb price is overall unsatisfying. The main result of the unsatisfied result is might caused by the relative low dimension of the data. Though we have many variables in the data set, it's still much fewer than the observation number, which might leads to the malfunction of support vector machine regression model.