Exercises 2

Jordan Despain and Jack Freeman

02/22/2023

Saratoga House Prices

We first looked at the "medium" model which includes all the variables except *pctCollege*, *sewer*, *waterfront*, *landValue*, and *newConstruction*. We ran 30 simulations where we split the data and trained and tested this model. We then averaged the RMSE from each simulation and got an average RMSE value of 67105. We then thought of ways to change this model to make it much better and achieve a lower average RMSE. Here is the model we came up with,

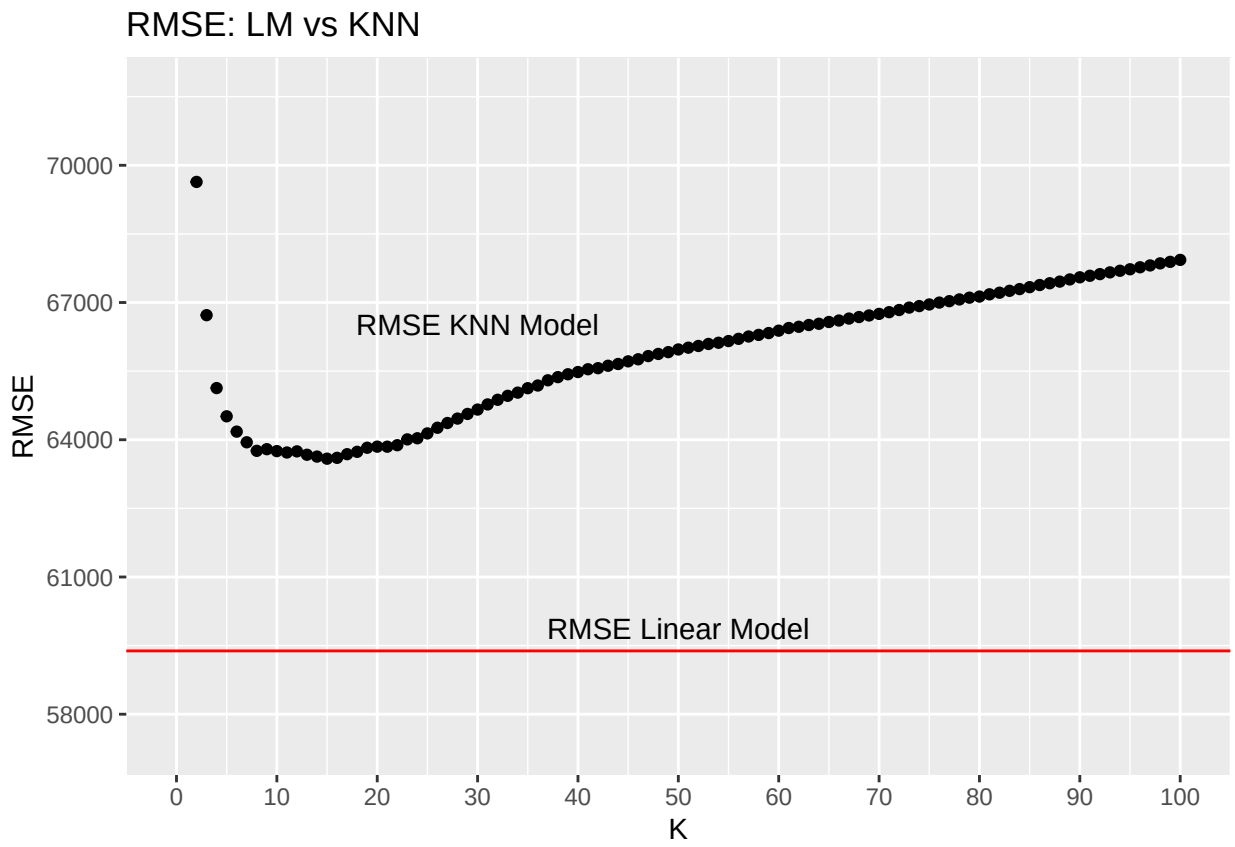
$$lm(price \sim . - sewer - fuel - heating + fireplaces * heating + landValue * lotSize + bedrooms * rooms + bathrooms * rooms + livingArea * centralAir + rooms * centralAir, data = saratoga\_train)$$

We chose to only exclude *sewer*, *fuel*, and *heating*. We figured those didn't have much impact on the price by themselves. We also include some interactions that seemed to make sense, like between the number of fireplaces and the type of heating the home has. This new model was a pretty nice improvement over the old. We did 30 simulations with this model as well, and we got an average RMSE value of 59384.

We now want to run a KNN regression model for price and see how this compares to our linear model's results. We took away all the interaction variables, standardized all the non-dummy variables, then included the new standardized variables along with the dummy variables in the model. We then set up this KNN model using K-values 2-100,

$$knnreg(price \sim ., data = s\_train\_scaled, k = x)$$

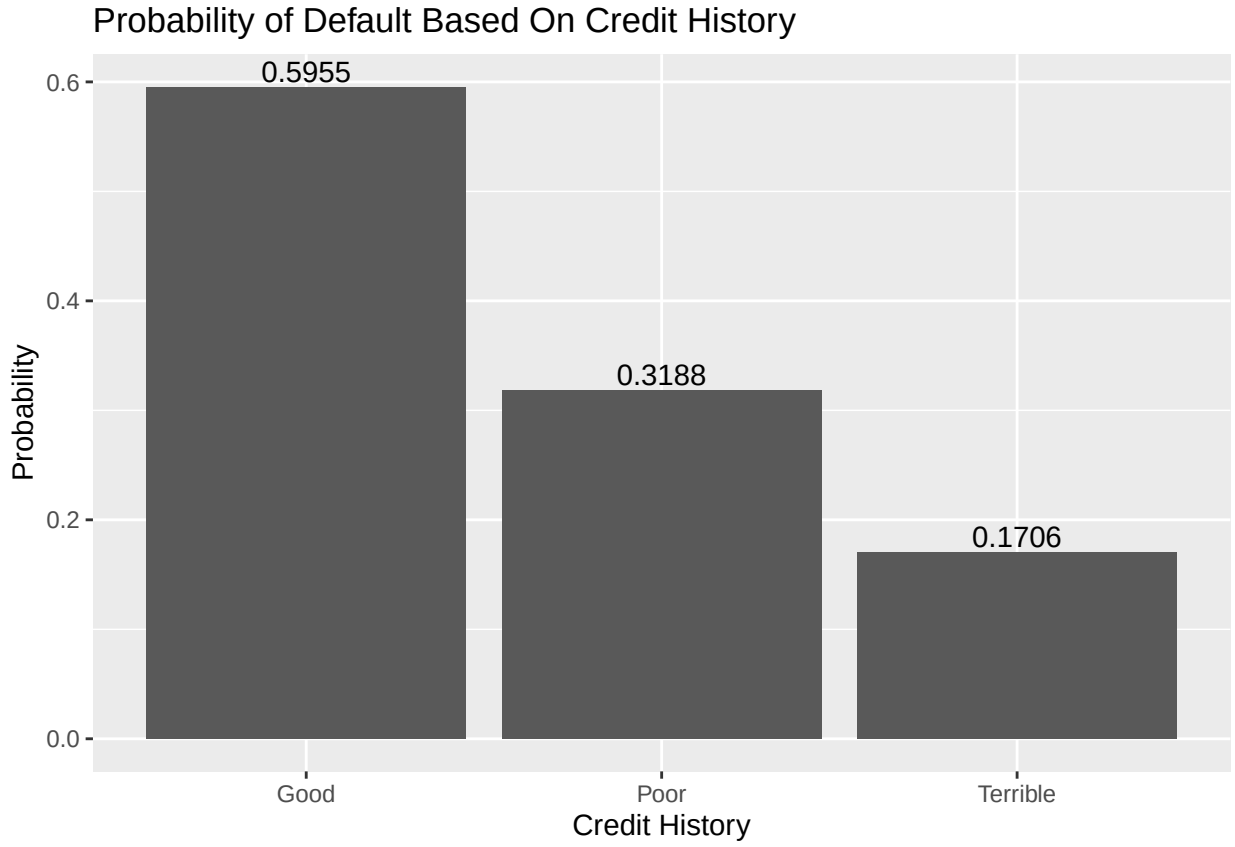
We again simulated this process 30 times and took the average RMSE for each K value. We now plot the results and compare them with the RMSE value we obtained from our new linear model. Here is what this looks like,



We see the minimum RMSE for our KNN model is 63586 at the K value 15. The linear model we came up with seems to perform better than the KNN model. This may have to with the fact that we are able build or model with interactions we know to be more important. We may have a better understanding of what effects prices in the real world versus the trained KNN model. We think it is good to have a human's opinion of importance on these variables if they have extensive knowledge in this area. Market values are changing all the time, along with what effects them and the weight of those effects. It is good to have someone who can be on top of this and constantly provide this information. We think this method will always keep the linear model just a bit ahead of the KNN model. The KNN model has to be trained to keep up with the times, where a model built by a human can instantly make any changes necessary.

Classification and Retrospective Sampling

Using the data set "german_credit.csv", we want to first make a bar plot of default probability based on credit history. There are three types of credit history in this data set, "Good", "Poor", and "Terrible". We grouped the data by each type of credit history, counted the number of defaults in each group, and used this to find the probability of defaulting for each group. We then created a bar plot to display the results. Here is what we found,



Next, we want to build a logistic regression model to predict the default probability. The variables we are including are *duration*, *amount*, *installment*, *age*, *history*, *purpose*, and *foreign*. Here is what this model looks like,

$$glm(Default \sim duration + amount + installment + age + history + purpose + foreign, data = german_credit, family = binomial)$$

We ran this model and the results of our coefficients are,

```
                          .
(Intercept)         -0.7075
duration             0.0253
amount               0.0001
installment          0.2216
age                 -0.0202
historypoor         -1.1076
historyterrible     -1.8847
purposeedu           0.7248
purposegoods/repair  0.1049
purposenewcar        0.8545
purposeusedcar      -0.7959
foreigngerman       -1.2647
```

We look at the coefficients for *historypoor* and *historyterrible* and use the exponent function on each to find our results state a poor credit history multiplies the odds of default by 0.3304 and a terrible credit history multiplies the odds of default by 0.1519. So, the results from our logistic model are saying as credit history improves, the odds of default are increasing.

Anyone that knows anything about loans and credit would be confused by the results from our bar plot and logistic regression model. It doesn't make sense for default probability to be decreasing as credit history worsens. We would expect the opposite to be happening. This "weird" result is most likely due to the way the data was collected. Since defaults are rare, the bank sampled a set of loans that defaulted, then tried to match them with a similar set of loans that had not defaulted. This led to a large oversampling of defaults. Because of this, we don't think this data set is appropriate for building a predictive model of defaults. The data was collected in a way that creates bias in the results. The bank should get a true random sample from the data to use for a predictive model. With a true random sample, we would be able to take the results of any predictive model more seriously.

Children and Hotel Reservations

We first want to compare the out-of-sample performance of two baseline models and a linear model we build. The two baseline models are,

$$lm(children \sim market\_segment + adults + customer\_type + is\_repeated_guest, data = hotels\_dev\_train)$$

and

$$lm(children \sim . - arrival\_date, data = hotels\_dev\_train)$$

We run multiple simulations where we split the data and run these models and then take the average RMSE for each model. The results are,

- Baseline 1 RMSE - 0.27
- Baseline 2 RMSE - 0.24

We now want to build a linear model that performs better than both of these baseline models. We excluded variables that we didn't think were very important, like *deposit_type*, and added interactions that we thought may be telling of whether the guests have children or not. The model we ended up with is,

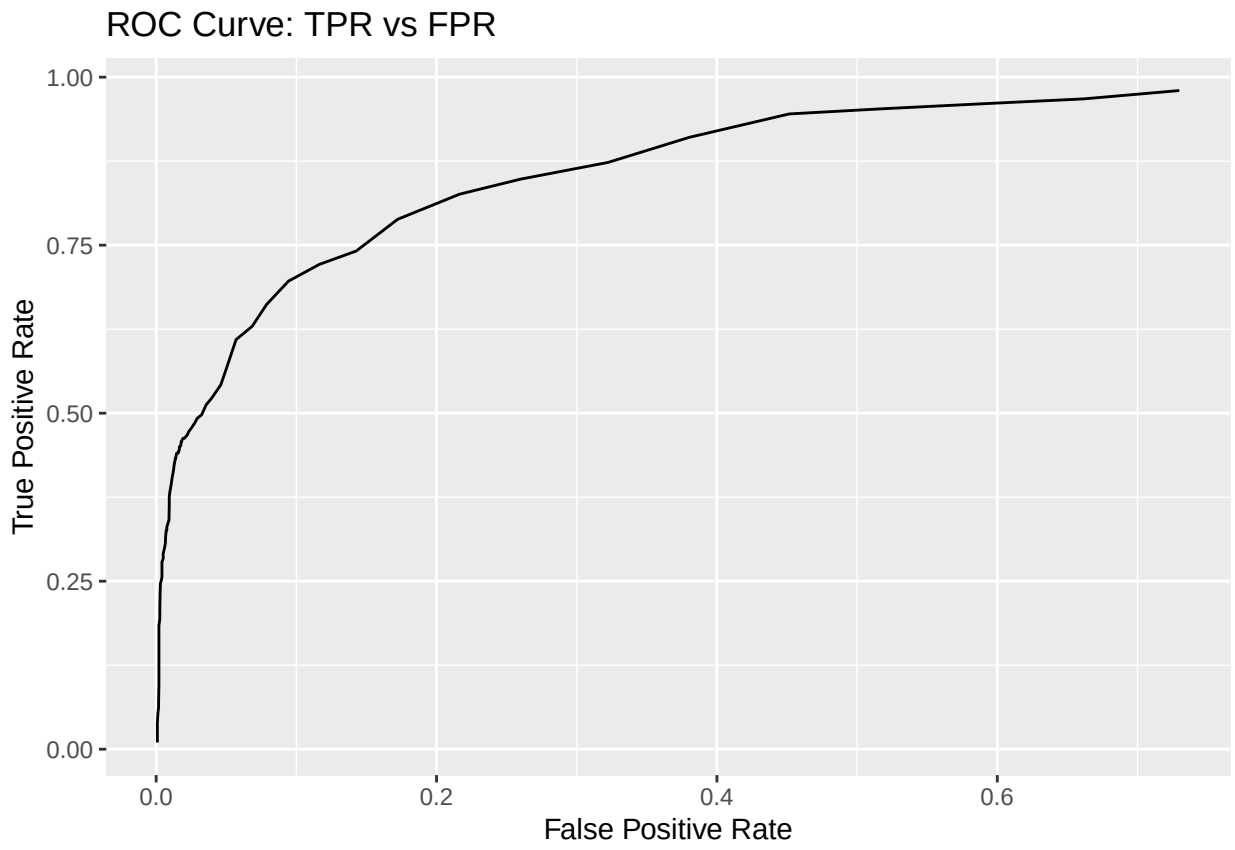
$$lm(children \sim . - arrival\_date - deposit\_type + adults * reserved\_room\_type + hotel * reserved\_room\_type + adults * distribution\_channel + adults * required\_car\_parking\_spaces, data = hotels\_dev\_train)$$

We did multiple simulations for this model as well. The average RMSE for our model is,

- Our Model RMSE - 0.23

We see that our model has a lower RMSE and outperforms both the baseline models.

We now want to validate our model. We create a ROC curve by plotting the TPR versus the FPR. Here is what our results look like,



We now want to create 20 folds of the "hotels_val.csv" data set. We then predict whether each booking will have children on it. We sum up the predicted probabilities, and then compare the expected number of bookings with children with the actual number of bookings with children in each fold. Here are these results,

Fold_ID	Expected	Actual	Difference
1	22.04	19	3.04
2	19.6	18	1.6
3	23	21	2
4	20.54	24	−3.46
5	21.97	24	−2.03
6	18.76	17	1.76
7	20.28	13	7.28
8	21.44	20	1.44
9	23.07	21	2.07
10	25.91	28	−2.09
11	22.52	22	0.52
12	20.15	15	5.15
13	20.68	28	−7.32
14	14.47	12	2.47
15	17.15	16	1.15
16	18.99	17	1.99
17	26.45	29	−2.55
18	22.66	20	2.66
19	21.22	17	4.22
20	20.89	21	−0.11

We see from the difference column that our expected numbers weren't too far off the actual. We think that our predictions are pretty good.