FALL 2019 , DS - 5110 , HW-5

PART A

Problem 1

Choose one of the "miniposters" created by your fellow classmates and posted on Piazza for Homework 3. Cite both the name of the student whose miniposter you chose and the original source of the dataset used in that miniposter.

Download and import that dataset into R, put it into a tidy format (if necessary), and print the first ten observations of the dataset.

Solution: I have chosen the following miniposter:

Name - Sonal Jain

Data scource - https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data

```
# Import dataset
df = read_csv('E:/FALL_2019/DS-5110-DM/HW5-DM/AB_NYC_2019.csv')
```

```
## Parsed with column specification:
## cols(
##   id = col_double(),
##   name = col_character(),
##   host_id = col_double(),
##   host_name = col_character(),
##   neighbourhood_group = col_character(),
##   neighbourhood = col_character(),
##   latitude = col_double(),
##   longitude = col_double(),
##   room_type = col_character(),
##   price = col_double(),
##   minimum_nights = col_double(),
##   number_of_reviews = col_double(),
##   last_review = col_date(format = ""),
##   reviews_per_month = col_double(),
##   calculated_host_listings_count = col_double(),
##   availability_365 = col_double()
## )
```

```
# Remove rows containing na values
df <- drop_na(df)

# Print first 10 rows
head(df, 10)
```

```
## # A tibble: 10 x 16
##       id name  host_id host_name neighbourhood_g~ neighbourhood latitude
##    <dbl> <chr>   <dbl> <chr>     <chr>            <chr>            <dbl>
##  1  2539 Clea~    2787 John      Brooklyn         Kensington        40.6
##  2  2595 Skyl~    2845 Jennifer  Manhattan        Midtown           40.8
##  3  3831 Cozy~    4869 LisaRoxa~ Brooklyn         Clinton Hill      40.7
##  4  5022 Enti~    7192 Laura     Manhattan        East Harlem       40.8
##  5  5099 Larg~    7322 Chris     Manhattan        Murray Hill       40.7
##  6  5121 Blis~    7356 Garon     Brooklyn         Bedford-Stuy~     40.7
##  7  5178 Larg~    8967 Shunichi  Manhattan        Hell's Kitch~     40.8
```

```
##  8  5203 Cozy~    7490 MaryEllen Manhattan        Upper West S~     40.8
##  9  5238 Cute~    7549 Ben       Manhattan        Chinatown         40.7
## 10  5295 Beau~    7702 Lena      Manhattan        Upper West S~     40.8
## # ... with 9 more variables: longitude <dbl>, room_type <chr>,
## #   price <dbl>, minimum_nights <dbl>, number_of_reviews <dbl>,
## #   last_review <date>, reviews_per_month <dbl>,
## #   calculated_host_listings_count <dbl>, availability_365 <dbl>
```

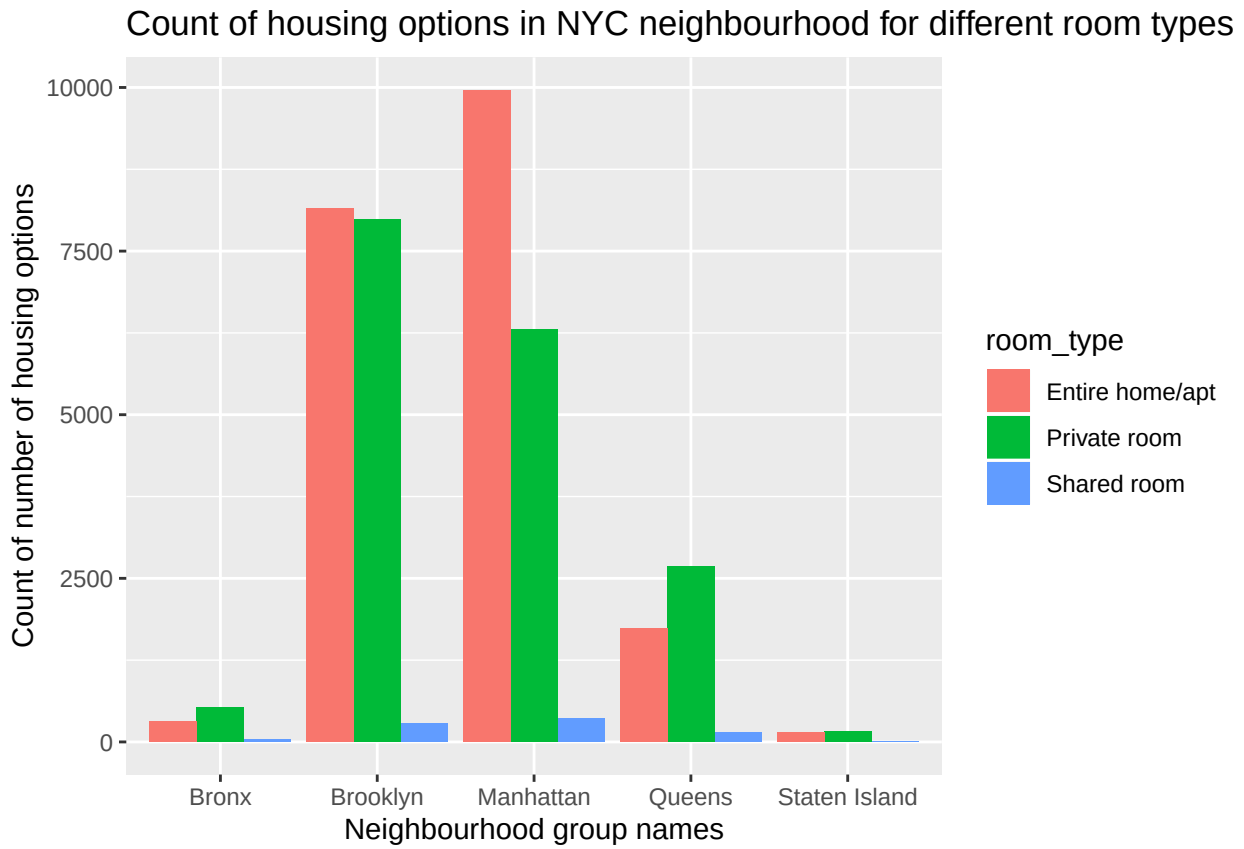
Problem 2

To the best of your ability, reproduce the figures from the miniposter you chose. You may contact the author of the original miniposter; if you do, cite and describe any information you receive from them.

Solution:

Graph 1

```
ggplot(df) +
  geom_bar(aes(x = neighbourhood_group, fill = room_type), position = 'dodge') +
  labs(x = 'Neighbourhood group names', y = 'Count of number of housing options',
       title = 'Count of housing options in NYC neighbourhood for different room types')
```

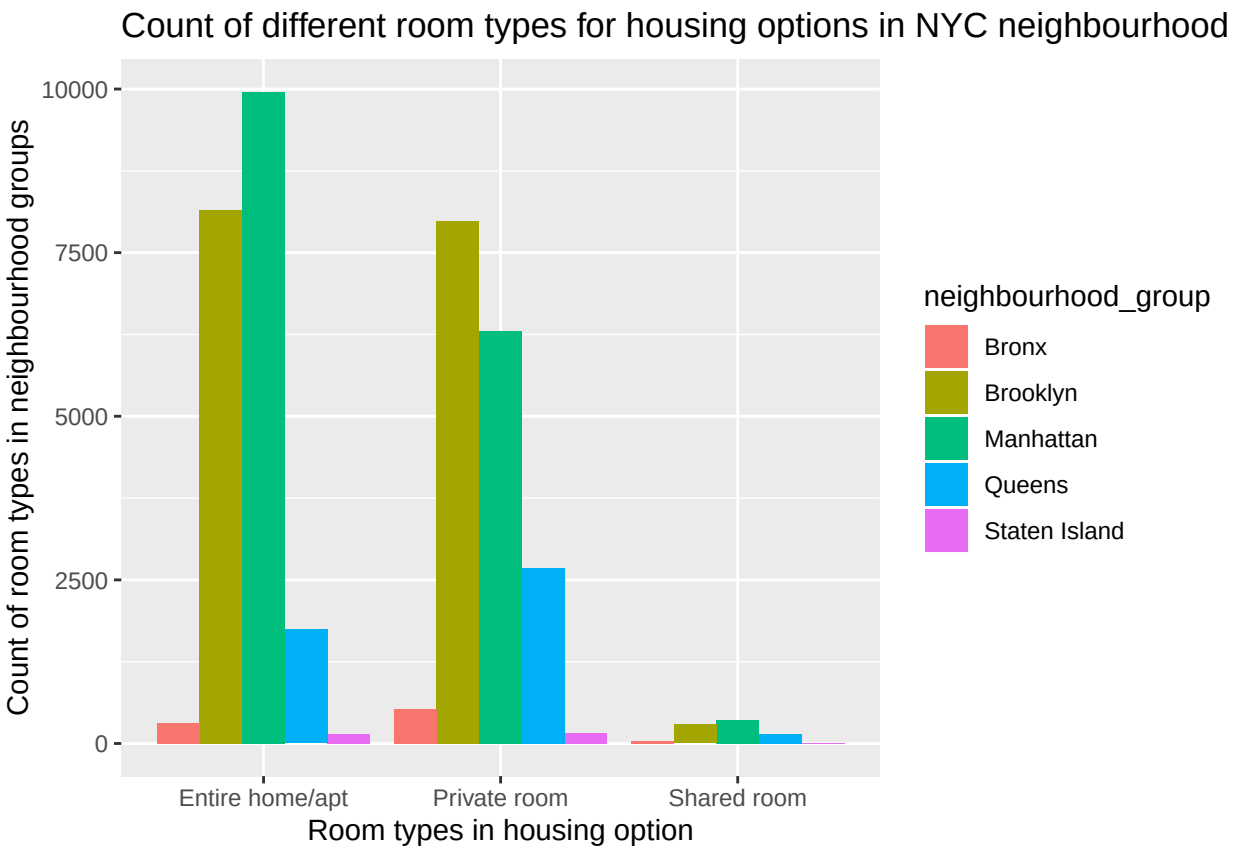


Conclusion

We can see that manhattan and brooklyn have on average more number of houisng options compared to other neighbourhood groups.

Graph 2

```
ggplot(df) +
  geom_bar(aes(x = room_type, fill = neighbourhood_group), position = 'dodge') +
  labs(x = 'Room types in housing option', y = 'Count of room types in neighbourhood groups',
       title = 'Count of different room types for housing options in NYC neighbourhood')
```

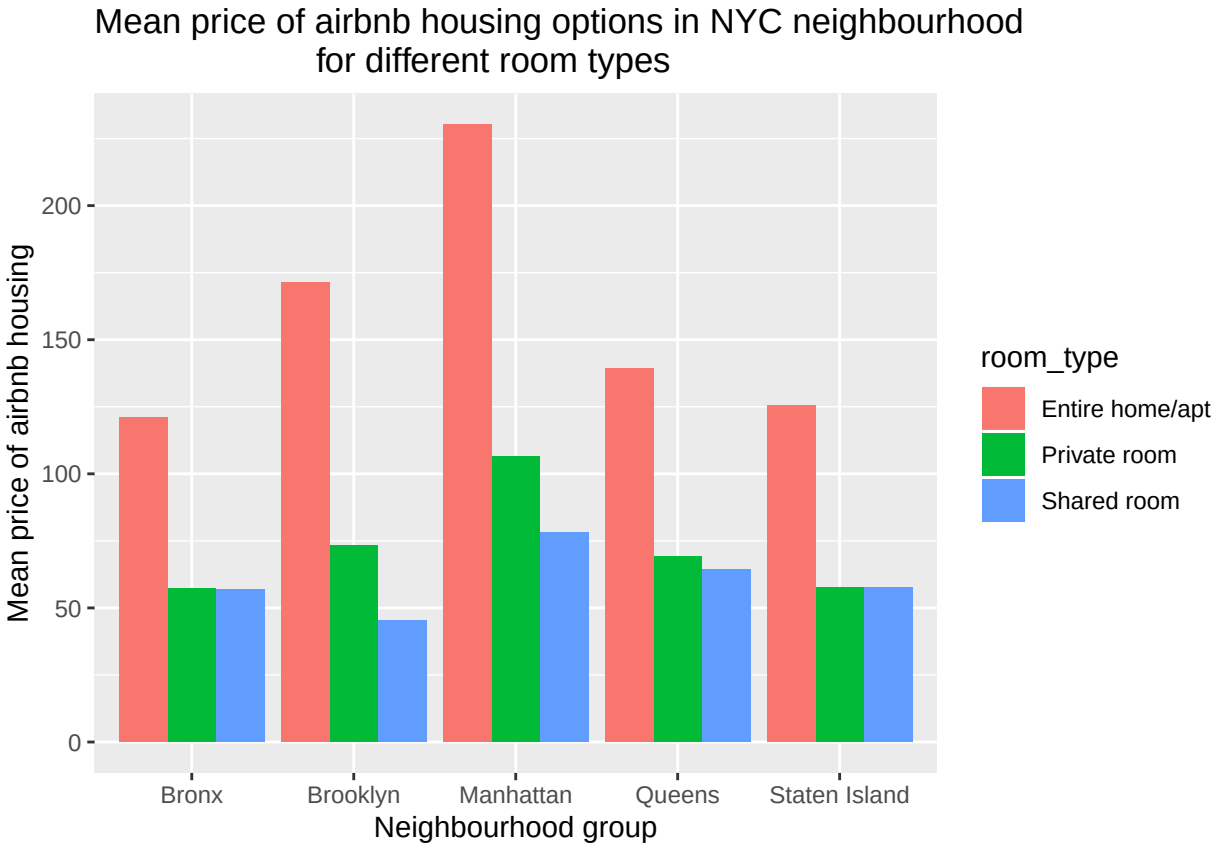


Conclusion

We can see that there are more entire home/apt and private rooms compared to shared rooms. Highest count is for Entire home/apt.

Graph 3

```
df %>%
  group_by(neighbourhood_group,room_type) %>%
  summarize(mean = mean(price)) %>%
  ggplot() +
    geom_col(aes(x = neighbourhood_group, y = mean,fill = room_type), position = 'dodge') +
    labs(x = 'Neighbourhood group', y = 'Mean price of airbnb housing',
      title = 'Mean price of airbnb housing options in NYC neighbourhood
                      for different room types')
```



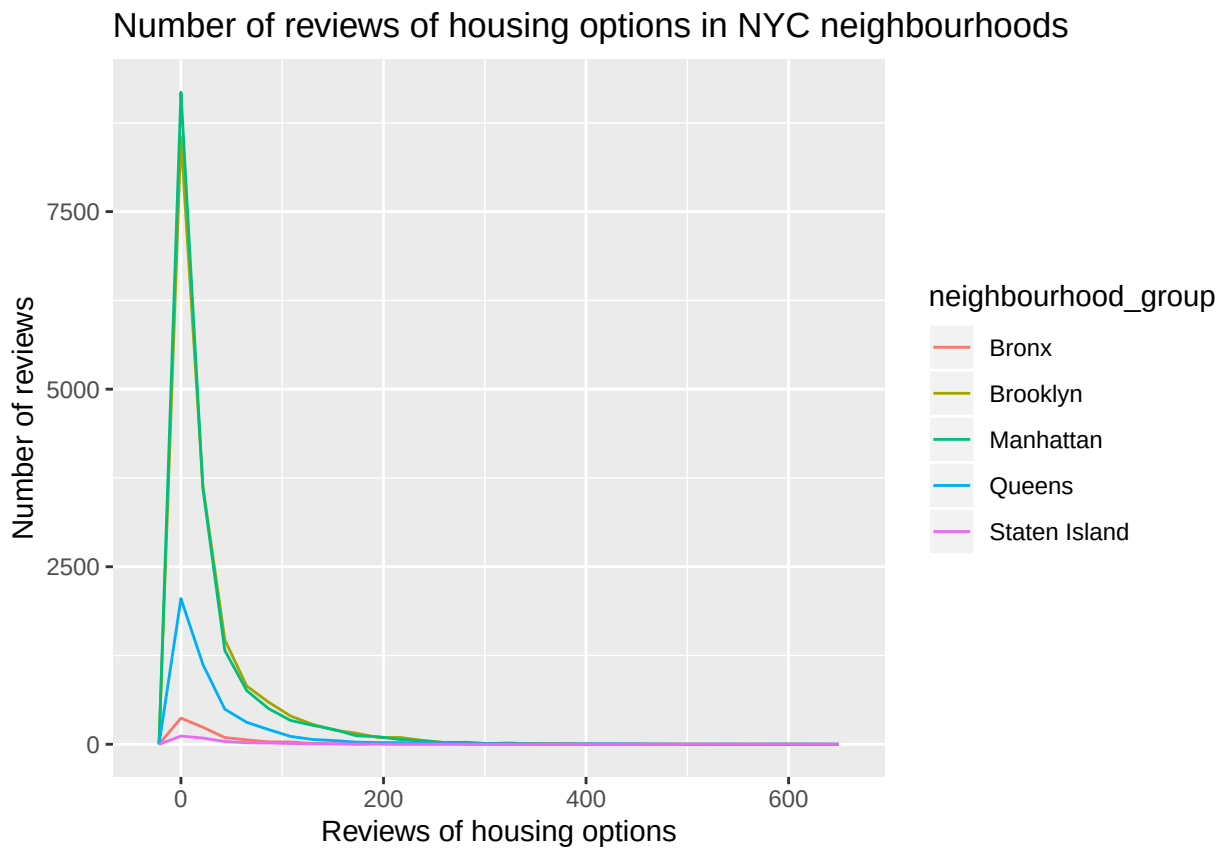
Conclusion

We can see that manhattan has highest average price for all room types. So we can say that manhattann is costliest location followed by Brooklyn.

Graph 4

```
ggplot(df) +
  geom_freqpoly(aes(x = number_of_reviews,color = neighbourhood_group)) +
  labs(x = 'Reviews of housing options', y = 'Number of reviews',
      title = 'Number of reviews of housing options in NYC neighbourhoods')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

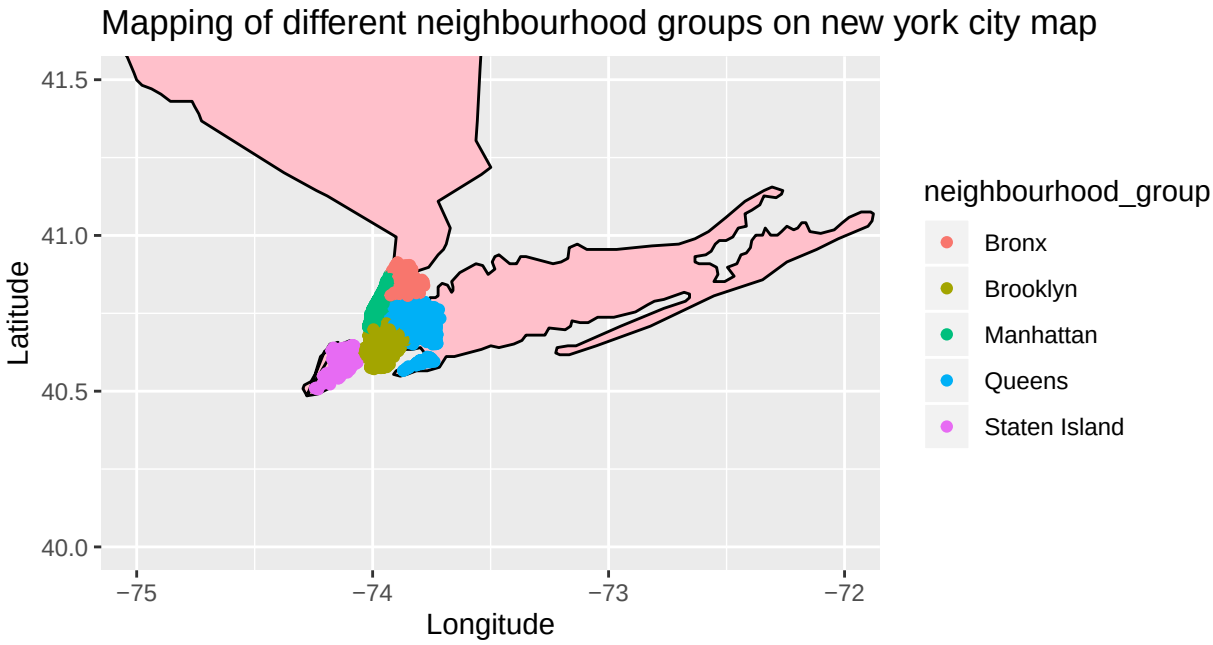


Conclusion

We can see that manhattan and brooklyn have almost same number of reviews. Staten island has least number of reviews.

Graph 5

```
map <- map_data('maps::state', region='new york')
ggplot() +
  geom_polygon(map,mapping=aes(x = long, y = lat, group = group), fill='pink',color="black") +
  geom_point(df, mapping = aes(x = longitude, y=latitude, color = neighbourhood_group)) +
  labs(x = 'Longitude', y = 'Latitude',
      title = 'Mapping of different neighbourhood groups on new york city map') +
  coord_quickmap(ylim = c(40,41.5), xlim = c(-75,-72))
```



Conclusion

Plot represents area of different neighbourhood groups on new york city map.

Problem 3

Write a function that performs cross-validation for a linear model (fit using lm) and returns the average root-mean-square-error across all folds. The function should take as arguments (1) a formula used to fit the model, (2) a dataset, and (3) the number of folds to use for cross-validation. The function should partition the dataset, fit a model on each training partition, make predictions on each test partition, and return the average root-mean-square-error (RMSE).

Solution:

Code

```
cross_val_kfold <- function(formula, dataset, k){
  set.seed(1)
  df_cv <- crossv_kfold(dataset,k)
  # fit model
  df_cv <- df_cv %>%
    mutate(fit = map(train,~ lm(formula,data = .)))
  # add predictions
  df_cv <- df_cv %>%
    mutate(pred = map2(test,fit,~add_predictions(.x,.y)))
  # calculate rmse on test set
  df_cv <- df_cv %>%
    mutate(rmse_test = map2_dbl(fit, test, ~ rmse(.x, .y)))
  return(mean(df_cv$rmse_test))
}
```

Conclusion

The above function will return the average root mean squared error.

Problem 4

Using 5-fold cross-validation, report the cross-validated RMSE of the model you proposed in Homework 4, Problem 5.

Now consider only the predictors age, dis, rad, tax, and ptratio. Use cross-validation to perform stepwise model selection with these variables to predict crim. Then create a plot showing the RMSE at each step of variable selection.

What was the most predictive model using these variables? How does it compare to the one proposed in Homework 4?

Solution:

Calculating cross validated RMSE of model proposed in HW4.

Model: lm(log(crim) ~ log(dis) + log(lstat))

```
hw4_rmse <- cross_val_kfold(log(crim) ~ log(dis) + log(lstat), BostonHousing, 5)
cat("RMSE for model proposed in HW4 is:", hw4_rmse)
```

```
## RMSE for model proposed in HW4 is: 1.35522
```

Consider variables age, dis, rad, tax, and ptratio.

Step 1

Define rmse vector

```
# Empty vector to store best rmse at each step
rmse_test <- c()

# Step 1
cat("RMSE with age is:",cross_val_kfold(log(crim)~age, BostonHousing,5),"\n")
```

```
## RMSE with age is: 1.628781
```

```
cat("RMSE with dis is:",cross_val_kfold(log(crim)~log(dis), BostonHousing,5),"\n")
```

```
## RMSE with dis is: 1.44862
```

```
cat("RMSE with rad is:",cross_val_kfold(log(crim)~rad, BostonHousing,5),"\n")
```

```
## RMSE with rad is: 1.126708
```

```
cat("RMSE with tax is:",cross_val_kfold(log(crim)~tax, BostonHousing,5),"\n")
```

```
## RMSE with tax is: 1.212385
```

```
cat("RMSE with ptratio is:",cross_val_kfold(log(crim)~ptratio, BostonHousing,5),"\n")
```

```
## RMSE with ptratio is: 1.994897
```

We get lowest rmse for rad.

Step 2

```
# Adding previous best rmse
rmse_test <- c(rmse_test,cross_val_kfold(log(crim)~rad,BostonHousing,5))

# Step 2
cat("RMSE with rad,age is:",cross_val_kfold(log(crim)~rad+age, BostonHousing,5),"\n")
```

```
## RMSE with rad,age is: 0.9172634
```

```
cat("RMSE with rad,dis is:",cross_val_kfold(log(crim)~rad+log(dis), BostonHousing,5),"\n")
```

```
## RMSE with rad,dis is: 0.8886042
```

```
cat("RMSE with rad,tax is:",cross_val_kfold(log(crim)~rad+tax, BostonHousing,5),"\n")
```

```
## RMSE with rad,tax is: 1.098825
```

```
cat("RMSE with rad,ptratio is:",cross_val_kfold(log(crim)~rad+ptratio, BostonHousing,5),"\n")
```

```
## RMSE with rad,ptratio is: 1.131738
```

We get lowest rmse for rad, dis.

Step 3

```
# Adding previous best rmse
rmse_test <- c(rmse_test,cross_val_kfold(log(crim)~rad+log(dis),BostonHousing,5))

# Step 3
cat("RMSE with rad,dis,age is:",cross_val_kfold(log(crim)~rad+log(dis)+age,BostonHousing,5),"\n")
```

```
## RMSE with rad,dis,age is: 0.8598151
```

```
cat("RMSE with rad,dis,tax is:",cross_val_kfold(log(crim)~rad+log(dis)+tax,BostonHousing,5),"\n")
```

```
## RMSE with rad,dis,tax is: 0.889432
```

```
cat("RMSE with rad,dis,ptratio is:",cross_val_kfold(log(crim)~rad+log(dis)+ptratio,BostonHousing,5),"\n
```

```
## RMSE with rad,dis,ptratio is: 0.8929391
```

We get lowest rmse for rad, dis, age.

Step 4

```
# Adding previous best rmse
rmse_test <- c(rmse_test,cross_val_kfold(log(crim)~rad+log(dis)+age,BostonHousing,5))

# Step 4
cat("RMSE with rad,dis,age,tax is:",cross_val_kfold(log(crim)~rad+log(dis)+age+tax,
                                                    BostonHousing,5),"\n")
```

```
## RMSE with rad,dis,age,tax is: 0.8604923
```

```
cat("RMSE with rad,dis,age,ptratio is:",cross_val_kfold(log(crim)~rad+log(dis)+age+ptratio,
                                                        BostonHousing,5),"\n")
```

```
## RMSE with rad,dis,age,ptratio is: 0.8621407
```

We get lowest rmse for rad, dis, age, tax but it is higher than step 4.

Step 5

```
# Adding previous best rmse
rmse_test <- c(rmse_test,cross_val_kfold(log(crim)~rad+log(dis)+age+tax,BostonHousing,5))

# Step 5
cat("RMSE with rad,dis,age, tax, ptratio is:",
    cross_val_kfold(log(crim)~rad+log(dis)+age+tax+ptratio,BostonHousing,5),"\n")
```

```
## RMSE with rad,dis,age, tax, ptratio is: 0.8627901
```

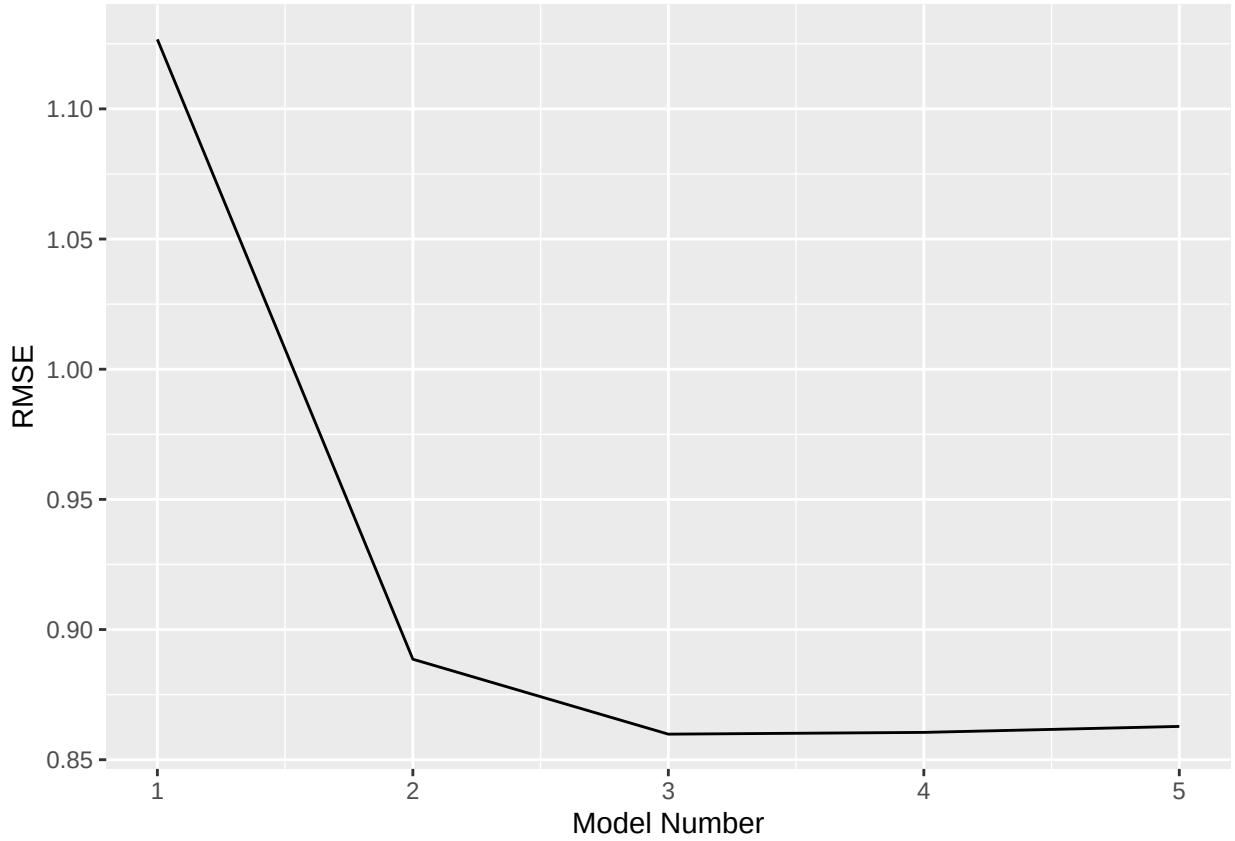
```
# Adding previous best rmse
rmse_test <- c(rmse_test,cross_val_kfold(log(crim)~rad+log(dis)+age+tax+ptratio,
                                                                    BostonHousing,5))
```

We get higher rmse than step 3.

Plot of rmse

```
fits_rmse <- tibble(nvar = 1:5,
                    rmse= rmse_test)

ggplot(fits_rmse) + geom_line(aes(x=nvar, y=rmse)) +
  labs(x = 'Model Number', y = 'RMSE')
```



Conclusion

Most predictive model using these variables was model 3 which is log(crim)~rad+log(dis)+age. Rmse for most predictive model is 0.8598151. Rmse for model proposed in HW4 is 1.35522. We can see that we get less rmse using 3 variable rad, log(dis), age compared to our model in HW4.

Problem 5

Can you report the final cross-validated RMSE for the "best" model that you found in Problem 4 as a good measure of the RMSE we could expect on new data? Why or why not?

Solution:

No. In cross validation we train and test k times on same data by splitting it differently each time. The new unseen data may have different trend/properties, so we may end up getting different rmse. So we can't report the final cross validated rmse for best model as good measure on unseen data. Also we have not considered any variables other then the 5 variables we used in problem 4.