Capstone project – New York City Real Estate

Ben Neely

March 10, 2019

OVERVIEW

Data scientists are often tasked with navigating large data sets to develop predictive models. In the current project, a database consisting of 84,548 property sales in New York City between 1 September 2016 and 31 August 2017 was analyzed. The purpose of this exercise was to develop and evaluate a predictive model consisting of various factors on property price. Specifically, the objective was to develop a predictive algorithm from a training data set (25% of total properties) that minimized root mean squared error (RMSE). Data cleansing and consolidation reduced the raw data set to 54,019 records. Model parameters included borough, number of residential and commercial units, land and gross area, decade built, tax.class, building class, day of week the property was sold, and month the property was sold. When all predictive factors were included in the model, RMSE = 0.331. The model was then applied to the validation data set (75% of total properties) and accuracy was assessed as the percent of predictions within 1 SE of mean property price and the percent of predictions within 2 SE of mean property price. Despite somewhat low RMSE, only 10.5% of predictions were within 1 SE of mean property price and 20.6% were within 2 SE of mean property price. These results suggest that the variables measured herein are suitable for providing an approximation of property price, but additional factors would need to be considered if increased accuracy was the objective.

METHODS

Data used for this exercise were contained in the New York City property sales database developed by the City of New York and hosted on kaggle. The initial step following data import was data cleansing. This primarily consisted of grouping data (e.g., lumping number of residential units per property into categories), converting numeric and character data to factors, and removing unnecessary columns. Residental units were grouped into 0, 1, 2, 3, 4 to 10, 11 to 50, and 51 or greater. Commercial units were grouped into 0, 1, 2, 3, and 4 or greater. Both land area and gross area were grouped into 0 sq ft, 1 to 2,000 sq ft, 2,001 to 4,000 sq ft, and 4,001 sq ft and greater. Decade the property was built was also identified to estimate property age at time of sale. Continuous data were converted to factors to allow compartmentalized algorithm development and application. The following script was used to conduct the above mentioned steps.

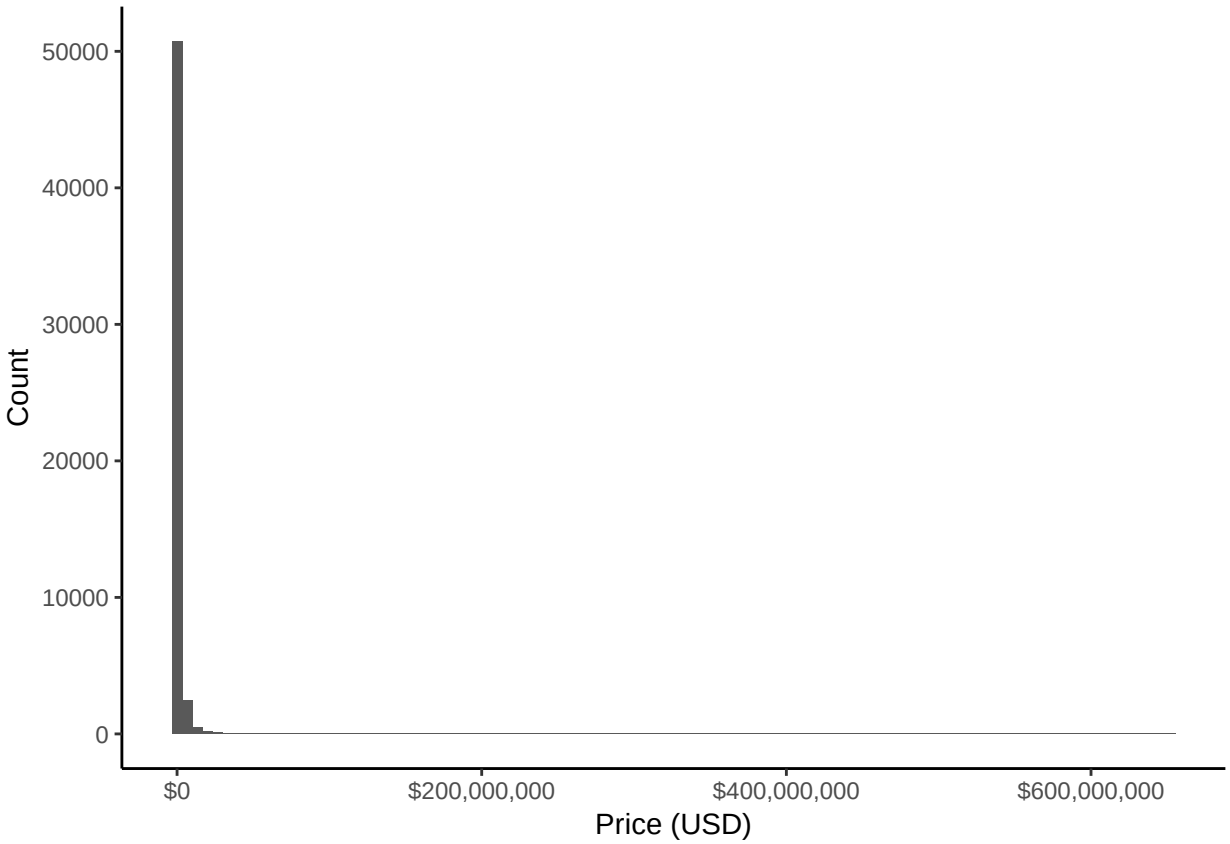
```
nyc=dat%>%
  select(-id)%>%
  mutate(borough=as.factor(borough))%>%
  select(-hood,-bldg.class.cat,-tax.class,-block,-lot,-easement,
         -bldg.class.prsnt,-address,-apt.num)%>%
  mutate(zip=as.factor(zip),
         resunits=ifelse(res.units==0,"0",
                  ifelse(res.units==1,"1",
                  ifelse(res.units==2,"2",
                  ifelse(res.units==3,"3",
                  ifelse(res.units>=4&res.units<=10,"4:10",
                  ifelse(res.units>=11&res.units<=50,"11:50",
                  ifelse(res.units>=51,"51+",NA)))))),
         resunits=as.factor(resunits),
         comunits=ifelse(com.units==0,"0",
                  ifelse(com.units==1,"1",
                  ifelse(com.units==2,"2",
                  ifelse(com.units==3,"3",
                  ifelse(com.units>=4,"4+",NA))))),
```

```
        comunits=as.factor(comunits),
        rescom=ifelse(res.units>=com.units,"res","com"),
        rescom=as.factor(rescom))%>%
  select(-res.units,-com.units,-tot.units)%>%
  mutate(land.area=as.numeric(land.area),
        landareacat=ifelse(land.area==0,"0",
                    ifelse(land.area>0&land.area<=2000,"1:2000",
                    ifelse(land.area>2000&land.area<=4000,"2001:4000",
                    ifelse(land.area>4000,"4000+",NA)))),
        landareacat=as.factor(landareacat),
        gross.area=as.numeric(gross.area),
        grossareacat=ifelse(gross.area==0,"0",
                     ifelse(gross.area>0&gross.area<=2000,"1:2000",
                     ifelse(gross.area>2000&gross.area<=4000,"2001:4000",
                     ifelse(gross.area>4000,"4000+",NA)))),
        grossareacat=as.factor(grossareacat),
        decade=as.factor(yr.built-(yr.built%%10)),
        tax.class=as.factor(tax.class.sale),
        bldg.class=as.character(bldg.class.sale),
        bldg.class=substr(bldg.class,1,1),
        bldg.class=as.factor(bldg.class),
        datetime=as.Date(datetime),
        day=as.factor(weekdays(datetime)),
        month=as.factor(month(datetime)))%>%
  select(-land.area,-gross.area,-yr.built,-tax.class.sale,-bldg.class.sale,-datetime)%>%
  mutate(price=gsub(" -  ",NA,price),
        price=as.numeric(price))%>%
  filter(price>=10000&price<1000000000)%>%
  drop_na(price)
```

This routine was followed by a secondary data check that revealed potential problems with the decade each property was built and building codes. Specifically, properties built before 1890 were removed due to small sample size and missing data, and only building codes with greater than 100 records were retained. These data cleansing steps resulted in a tidy set of variables that could be used to predict property sale price.

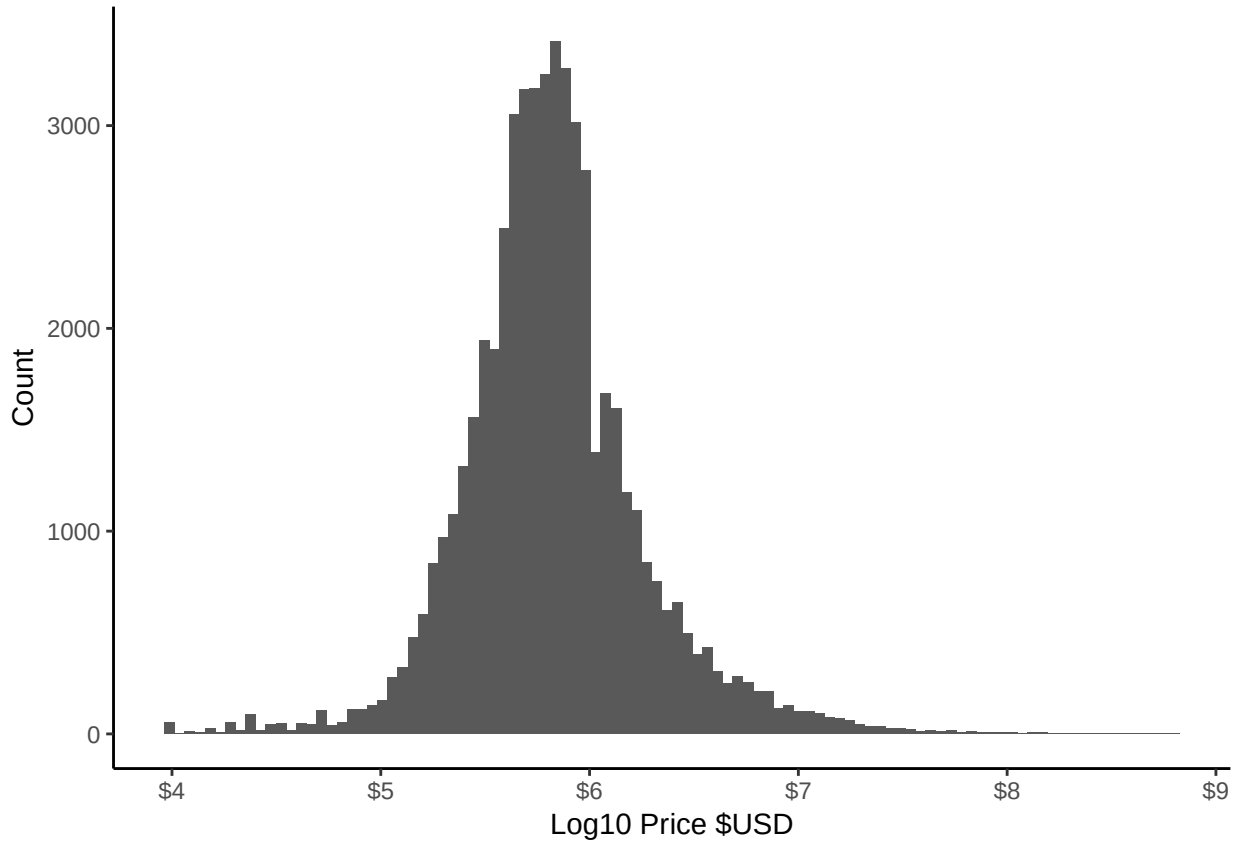
```
keeps=c("A","B","C","D","E","F","G","H","K","O","R","S","V")

nycdat=nyc%>%
  select(-zip)%>%
  mutate(decade=as.numeric(as.character(decade)))%>%
  filter(decade>=1890)%>%
  mutate(decade=as.factor(decade))%>%
  filter(bldg.class %in% keeps)%>%
  droplevels()
```

A cursory view of sales data revealed that minimum property price was \$0 USD and maximum property price was \$2,210,000,000 USD. To mitigate effects of properties that were priced at extreme low or high values, those $<$ \$10,000 USD and those $>$ \$1,000,000,000 USD were removed from analyses. Additionally, properties with no price listed were also removed from analyses.

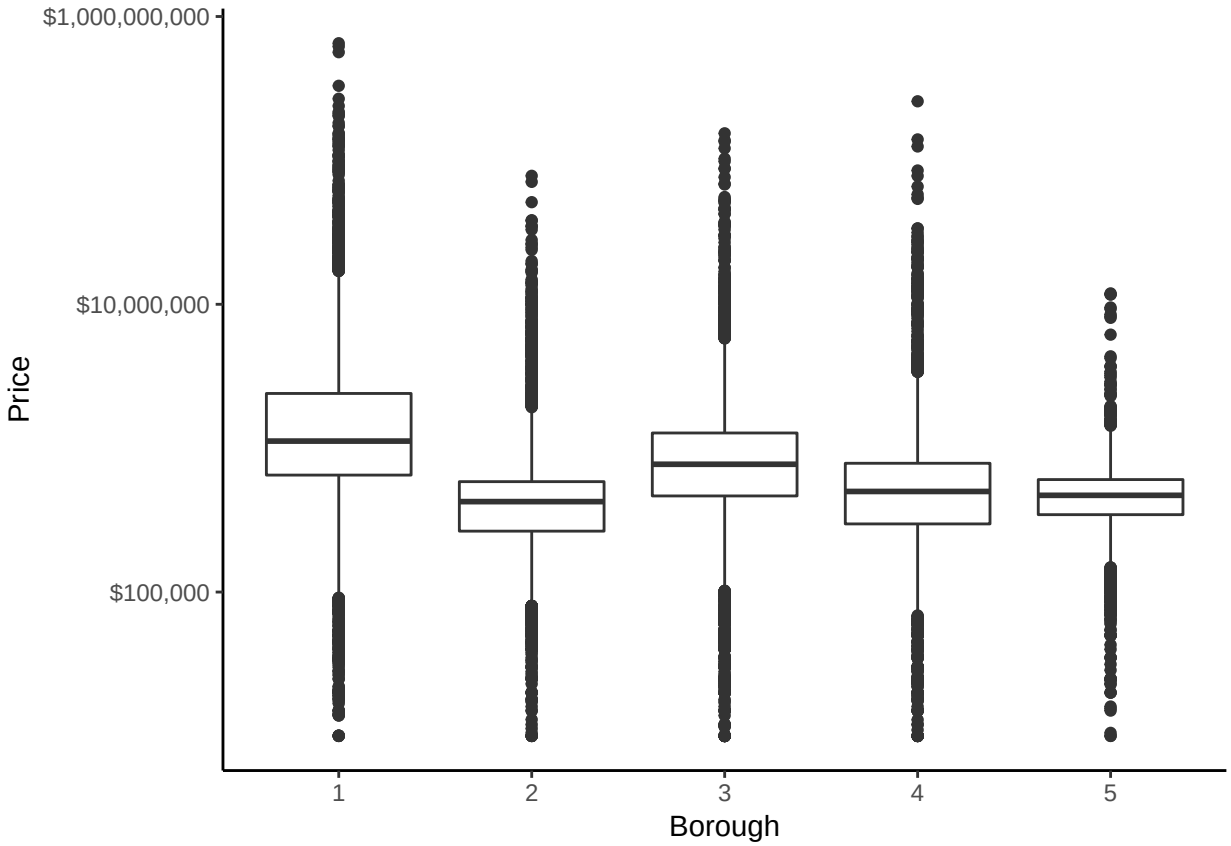


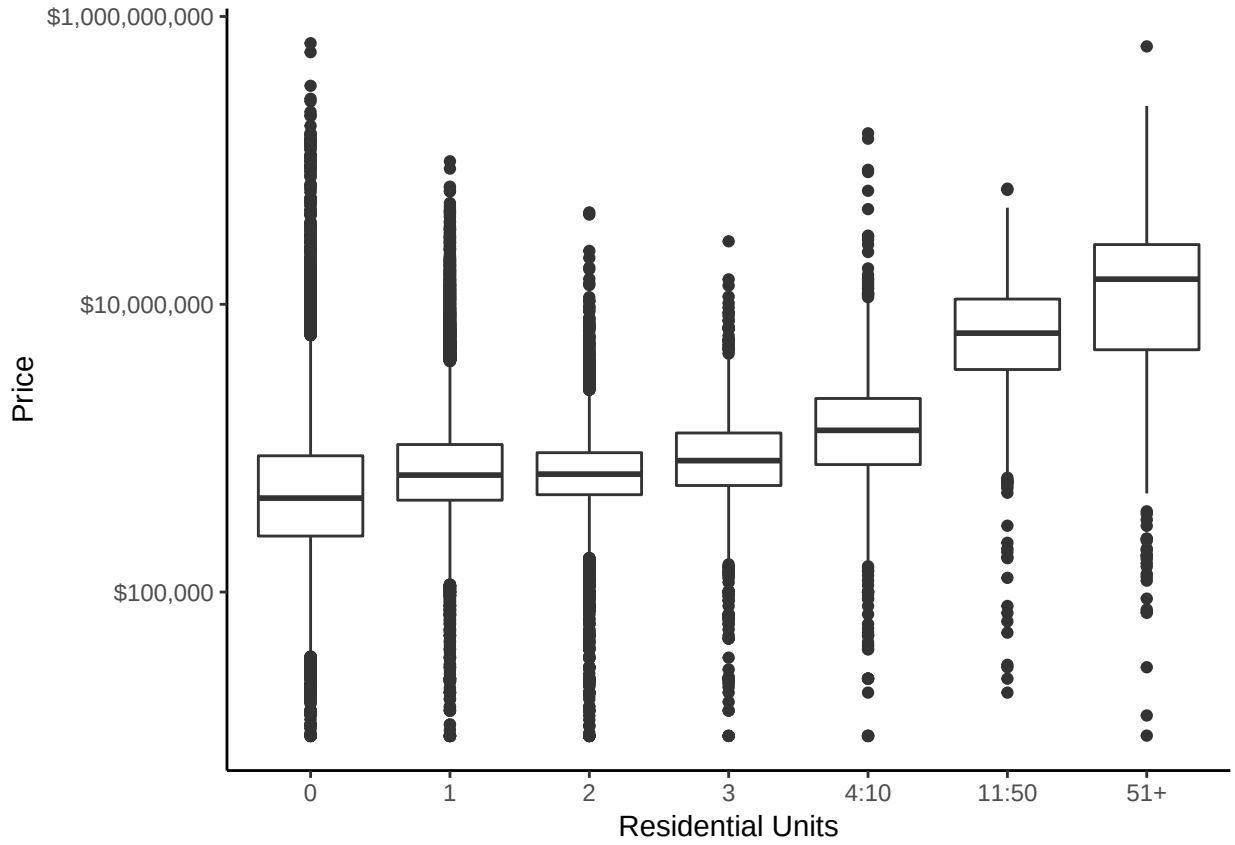
Despite attempts to standardize sale price data, extreme values still drove the distribution. Price was log10 transformed and the distribution was again examined.

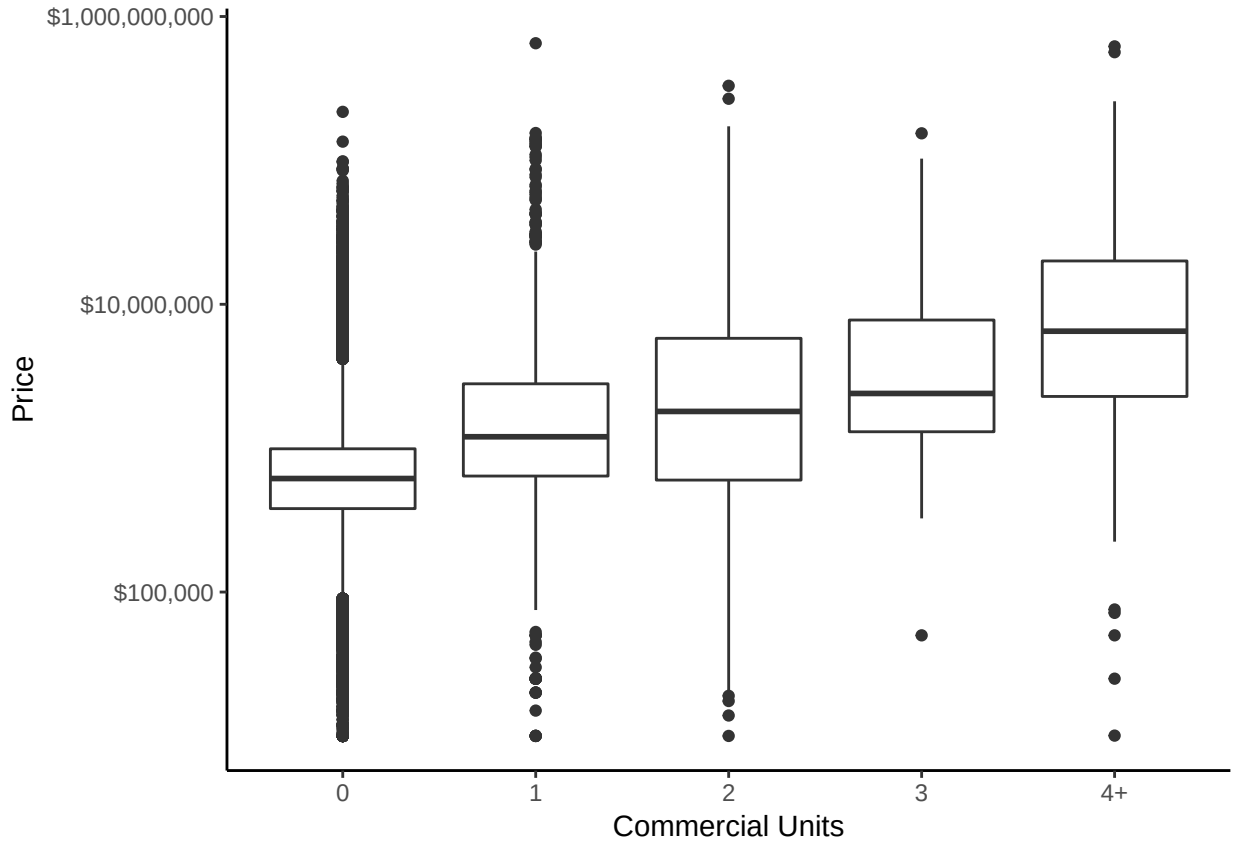
```
nycdat=nycdat%>%
  mutate(logprice=log10(price))
nycdat%>%
  ggplot(aes(x=logprice))+
  geom_histogram(bins=100)+
  scale_x_continuous(labels=dollar)+
  labs(x="Log10 Price $USD",y="Count")+
  theme_classic()
```

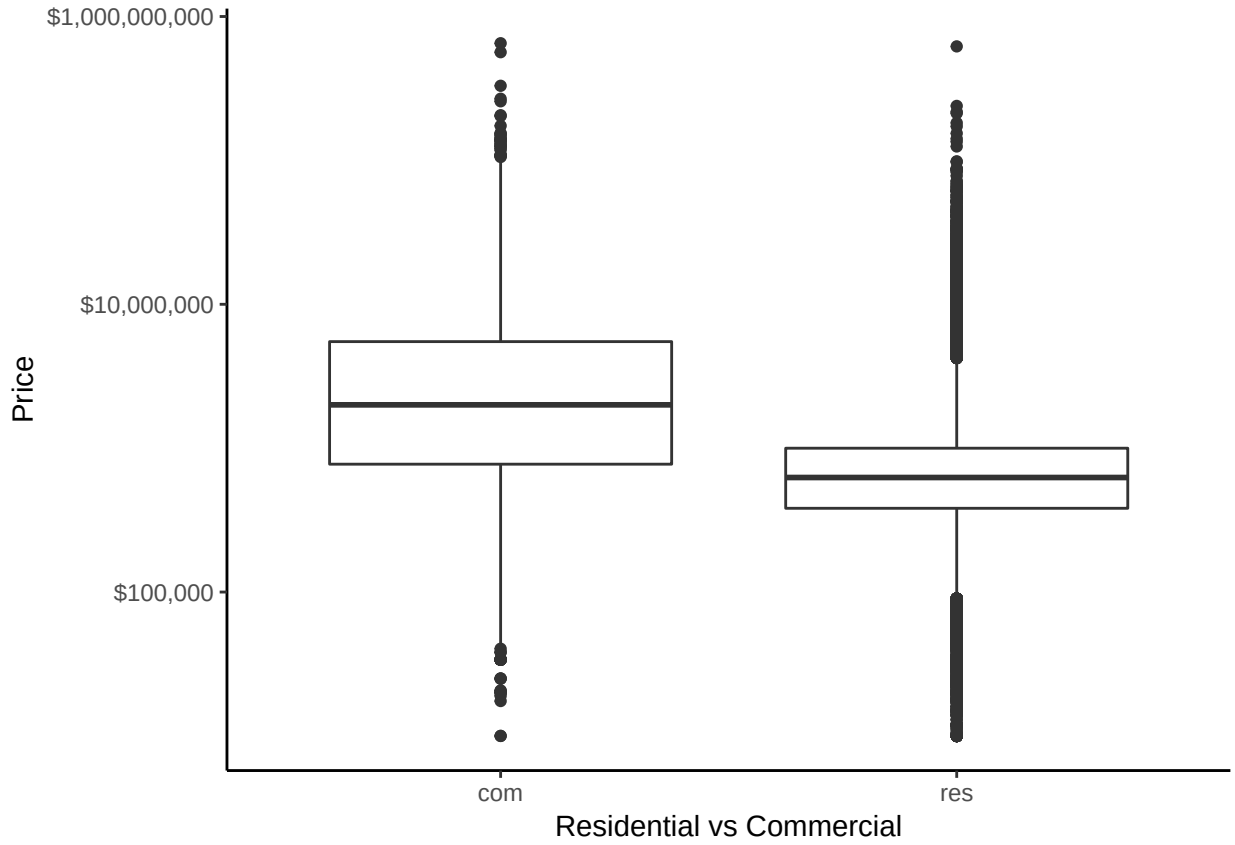


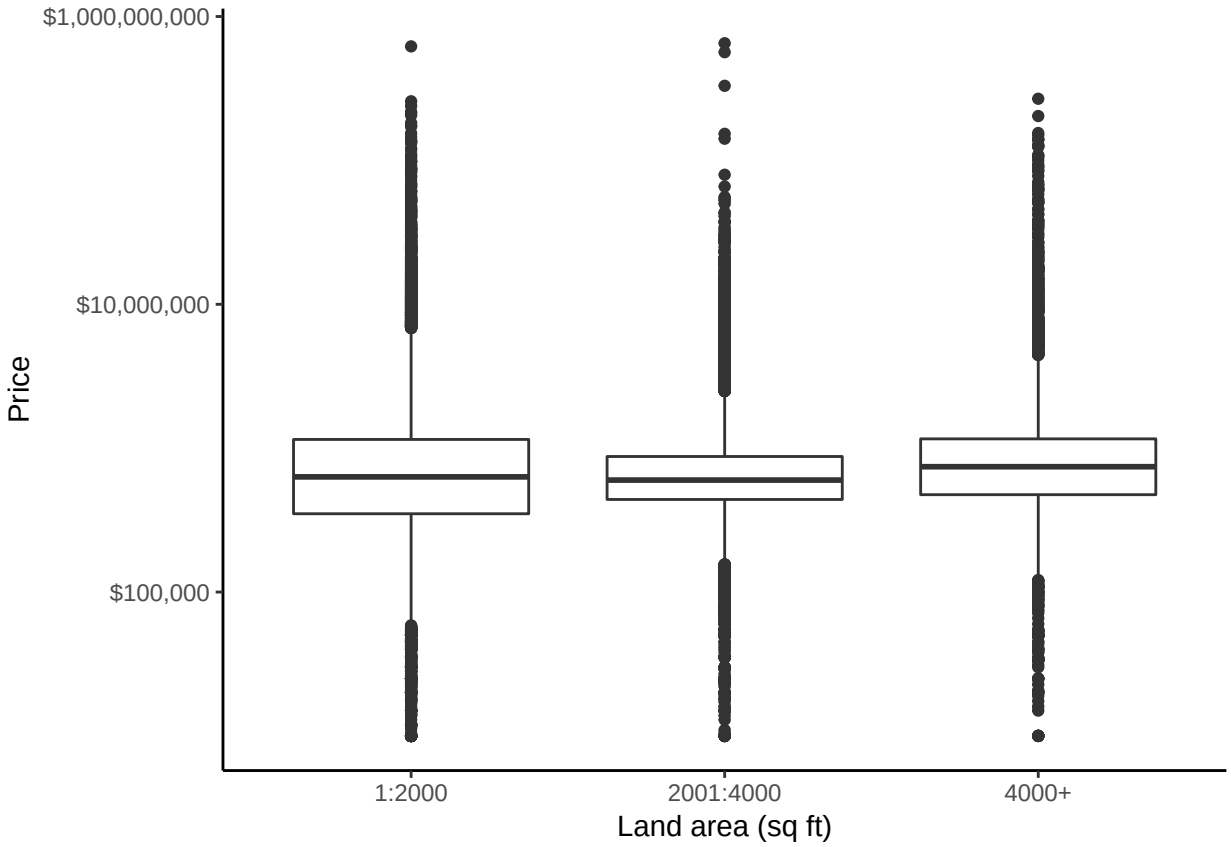
Following data cleansing and price transformation, predictor variables were examined against property price to examine for cursory strong relationships.

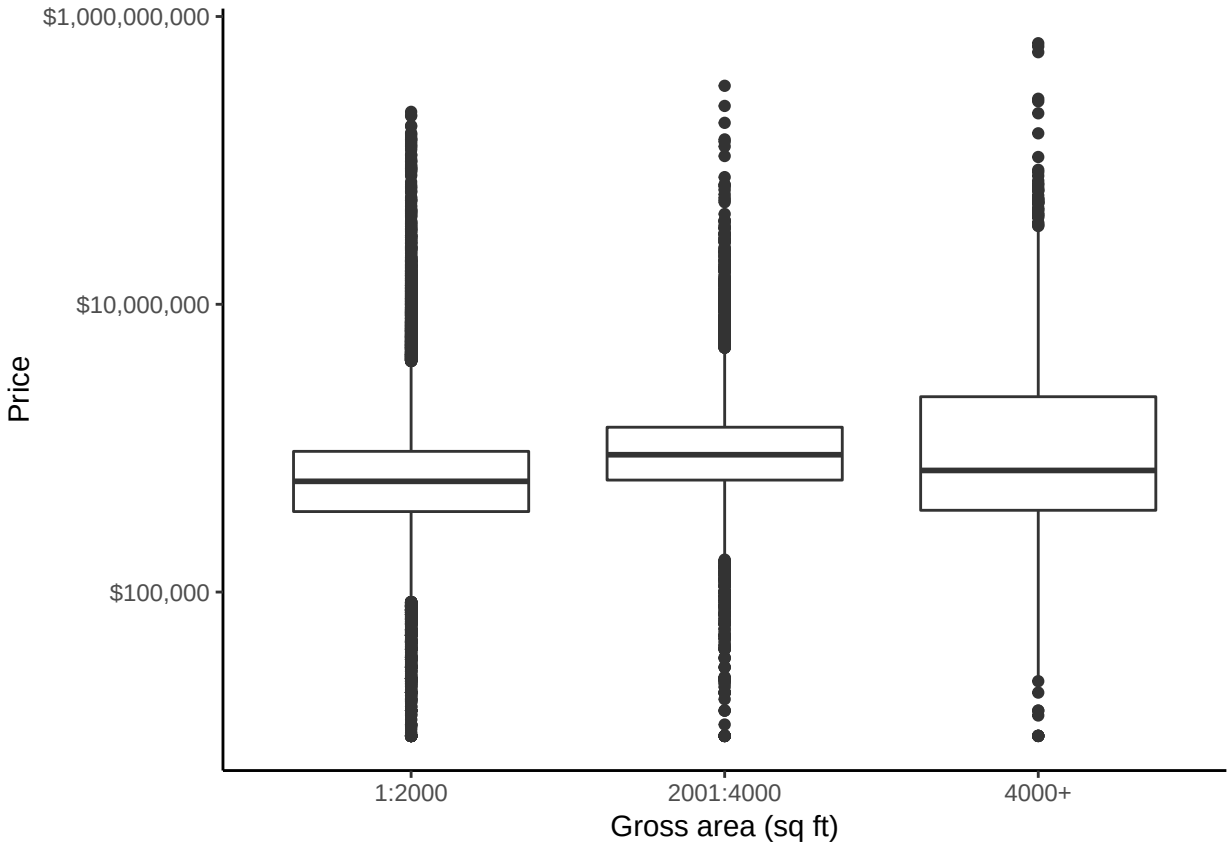

$1,000,000,000
$10,000,000
$100,000
Price
1
2
3
4
5
Borough

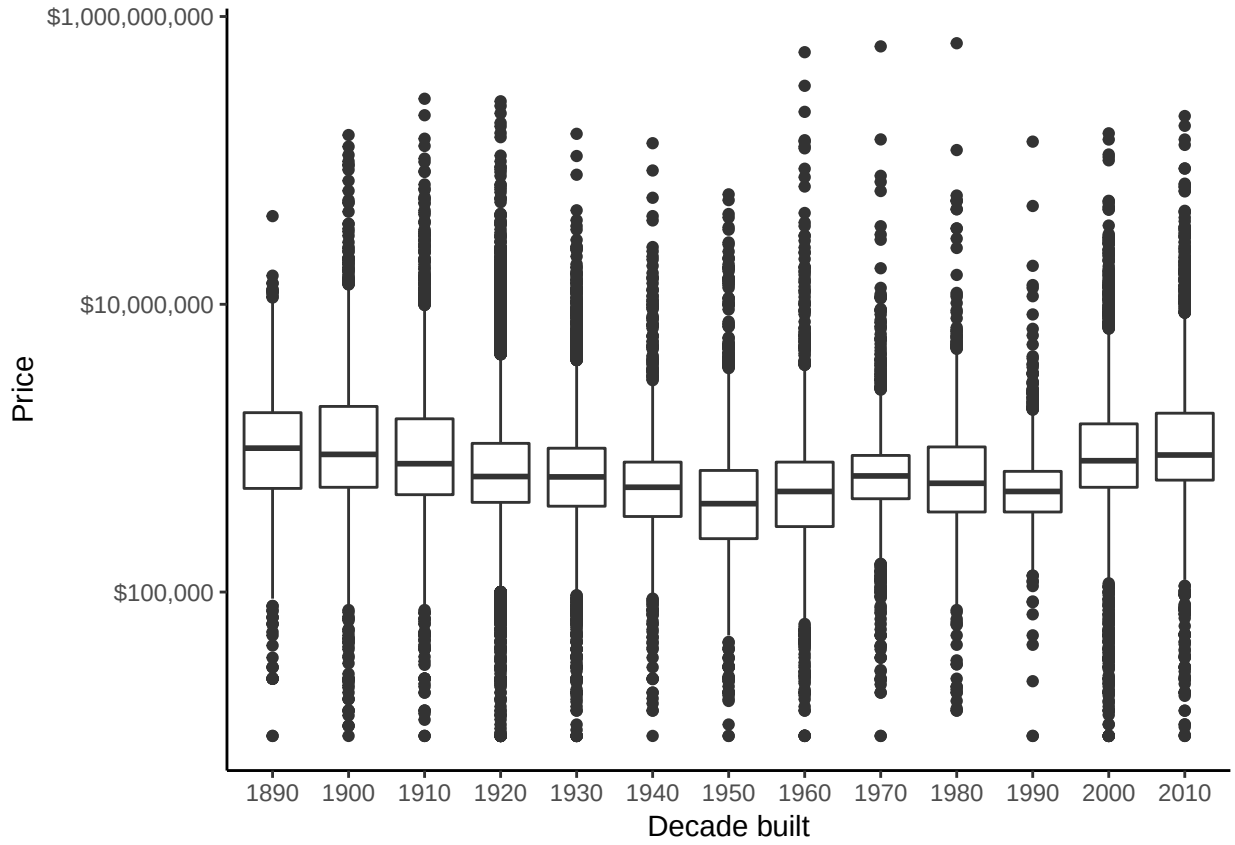

$1,000,000,000
$10,000,000
$100,000
Price
0
1
2
3
4:10
11:50
51+
Residential Units

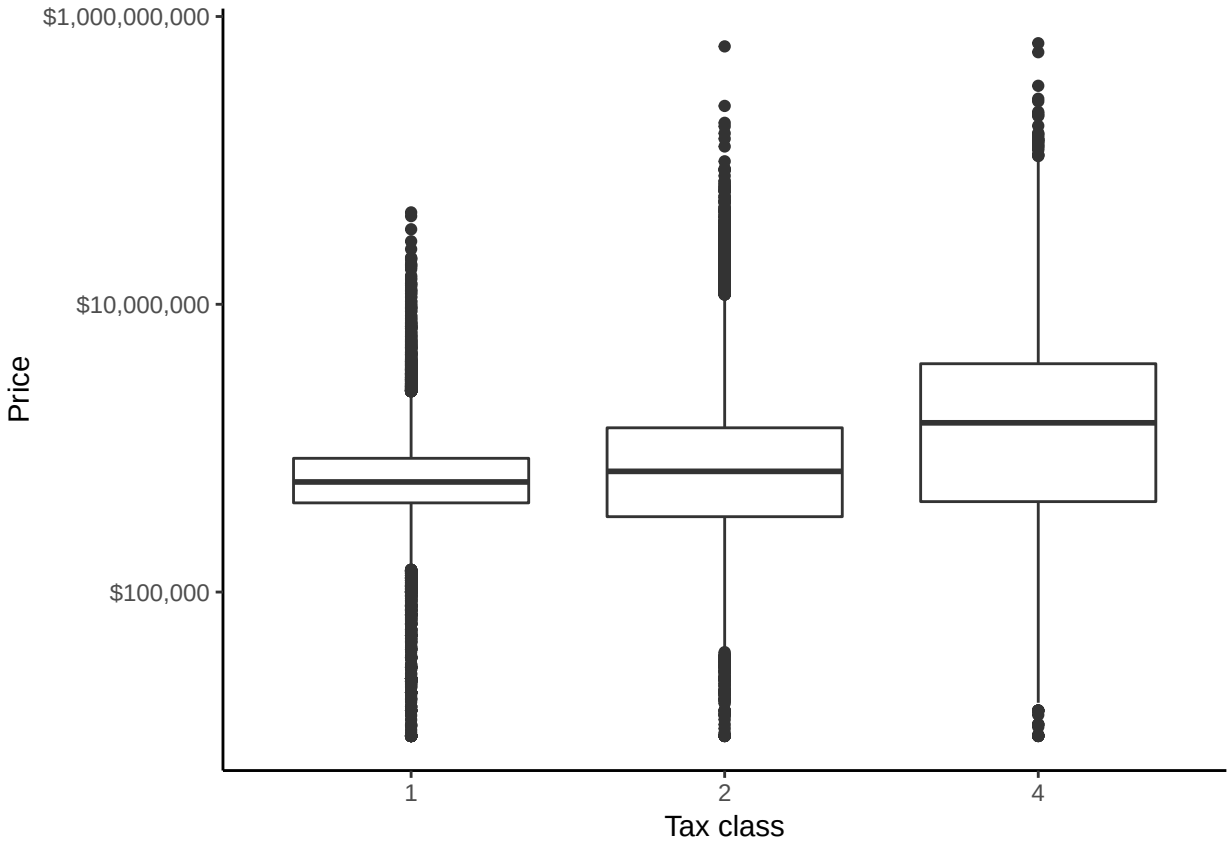

$1,000,000,000
$10,000,000
$100,000
Price
0
1
2
3
4+
Commercial Units

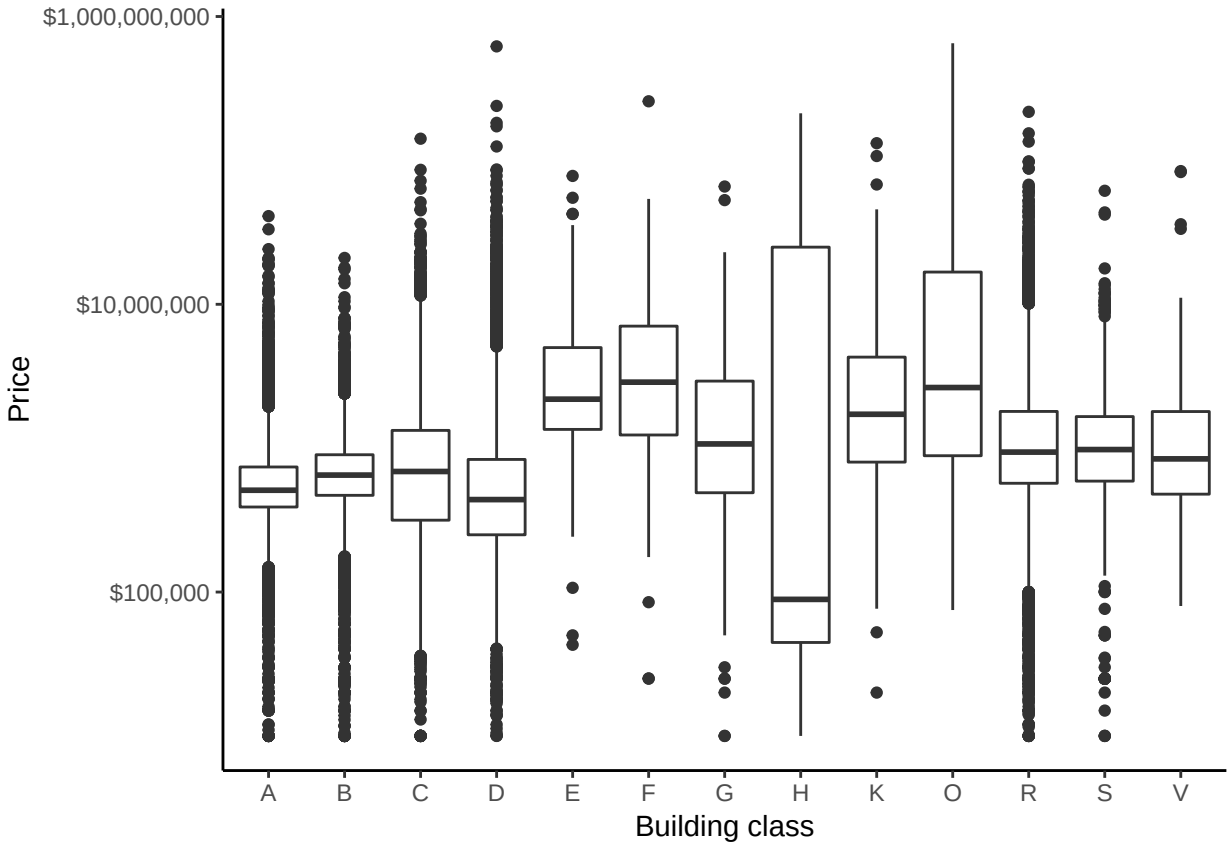

$1,000,000,000
$10,000,000
$100,000
Price
com
res
Residential vs Commercial

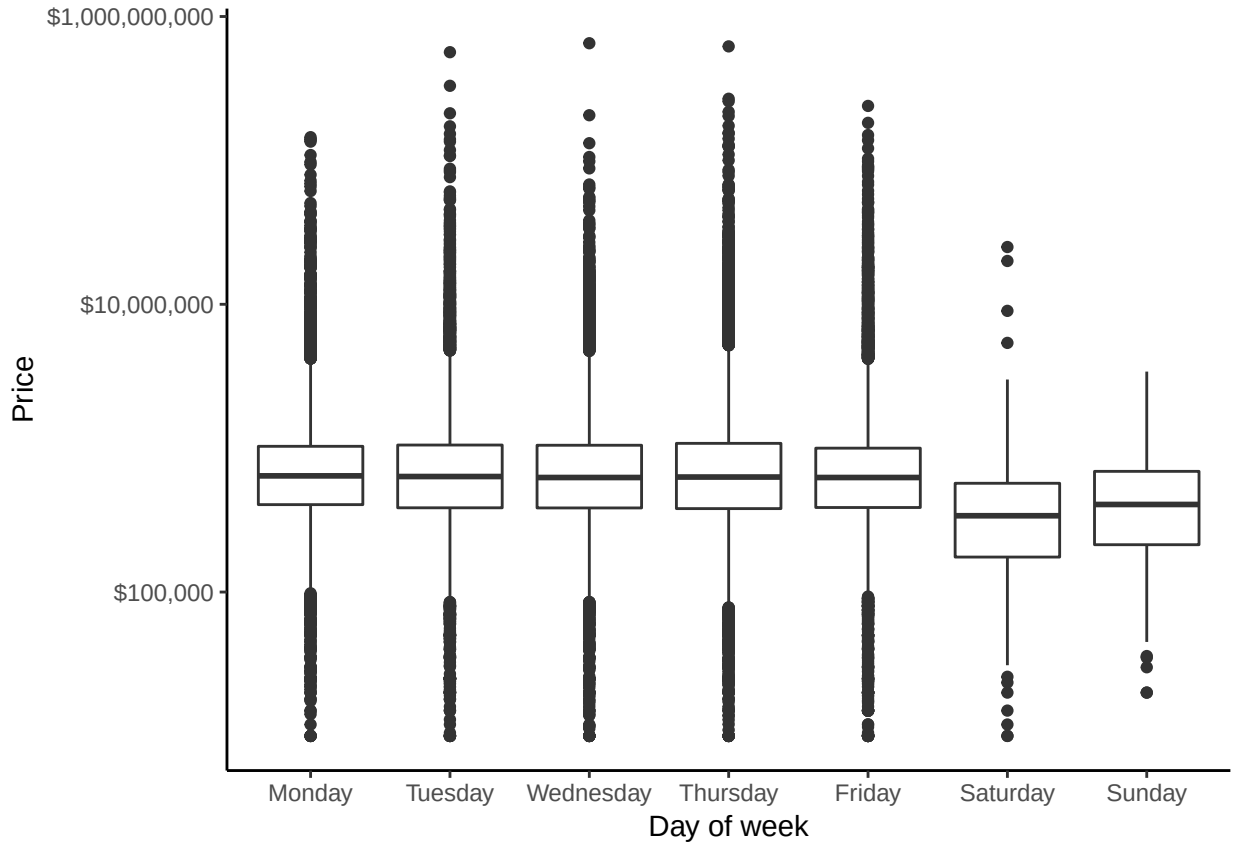

$1,000,000,000
$10,000,000
$100,000
Price
1:2000
2001:4000
4000+
Land area (sq ft)

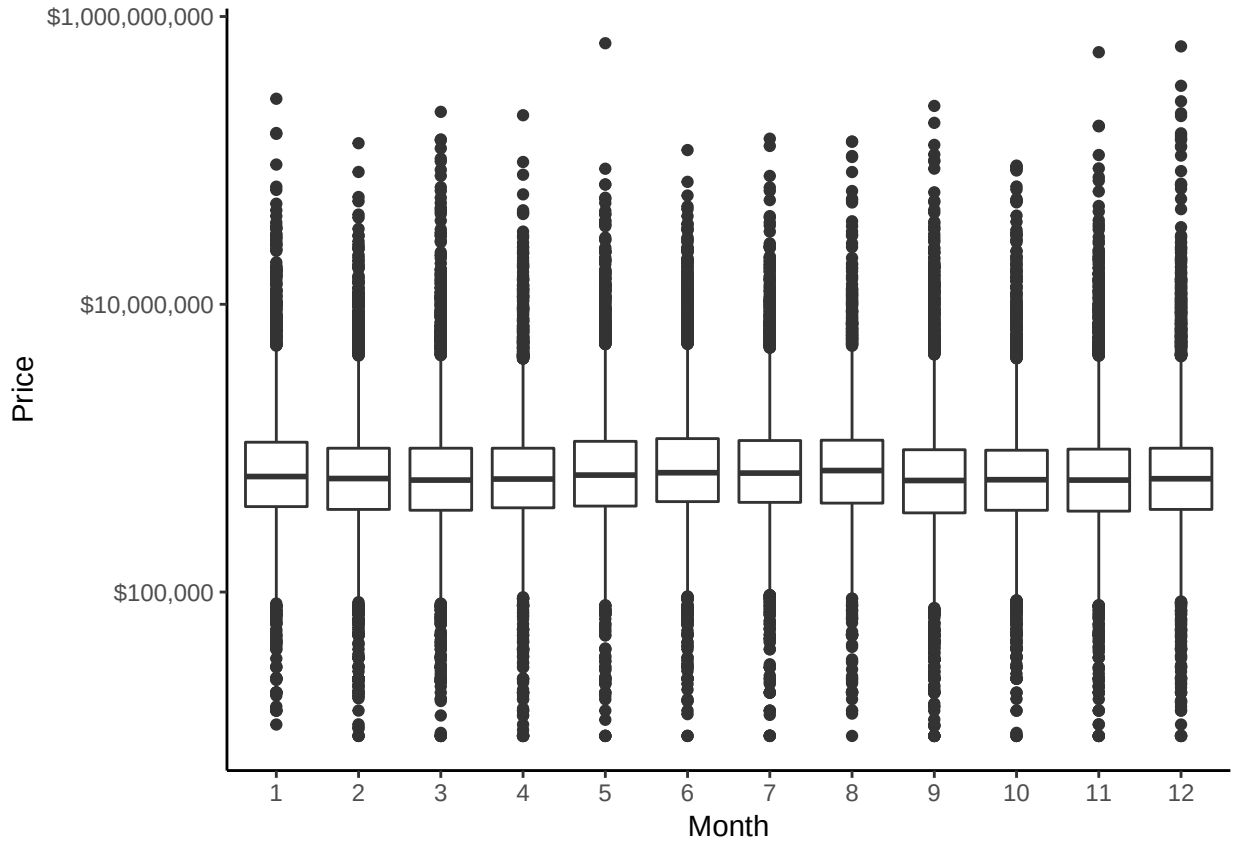

$1,000,000,000
$10,000,000
$100,000
Price
1:2000
2001:4000
4000+
Gross area (sq ft)


$1,000,000,000
$10,000,000
$100,000
Price
1890
1900
1910
1920
1930
1940
1950
1960
1970
1980
1990
2000
2010
Decade built


$1,000,000,000
$10,000,000
$100,000
Price
1
2
4
Tax class


$1,000,000,000
$10,000,000
$100,000
Price
A
B
C
D
E
F
G
H
K
O
R
S
V
Building class


$1,000,000,000
$10,000,000
$100,000
Price
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Day of week



Although several factors seemed to have greater influence on purchase price, all were analyzed to maximize explained variablity. The tidy data set to be analyzed had the following form.

```
## # A tibble: 6 x 13
##   borough  price resunits comunits rescom landareacat grossareacat decade
##   <fct>    <dbl> <fct>    <fct>    <fct>  <fct>       <fct>        <fct>
## 1 1       6.62e6 4:10     0        res    1:2000      4000+        1900
## 2 1       3.94e6 4:10     0        res    1:2000      4000+        1910
## 3 1       8.00e6 4:10     0        res    2001:4000   4000+        1900
## 4 1       3.19e6 4:10     0        res    1:2000      2001:4000    1920
## 5 1       1.62e7 11:50    0        res    4000+       1:2000       1920
## 6 1       1.03e7 4:10     0        res    2001:4000   1:2000       2000
## # ... with 5 more variables: tax.class <fct>, bldg.class <fct>, day <fct>,
## #   month <fct>, logprice <dbl>
```

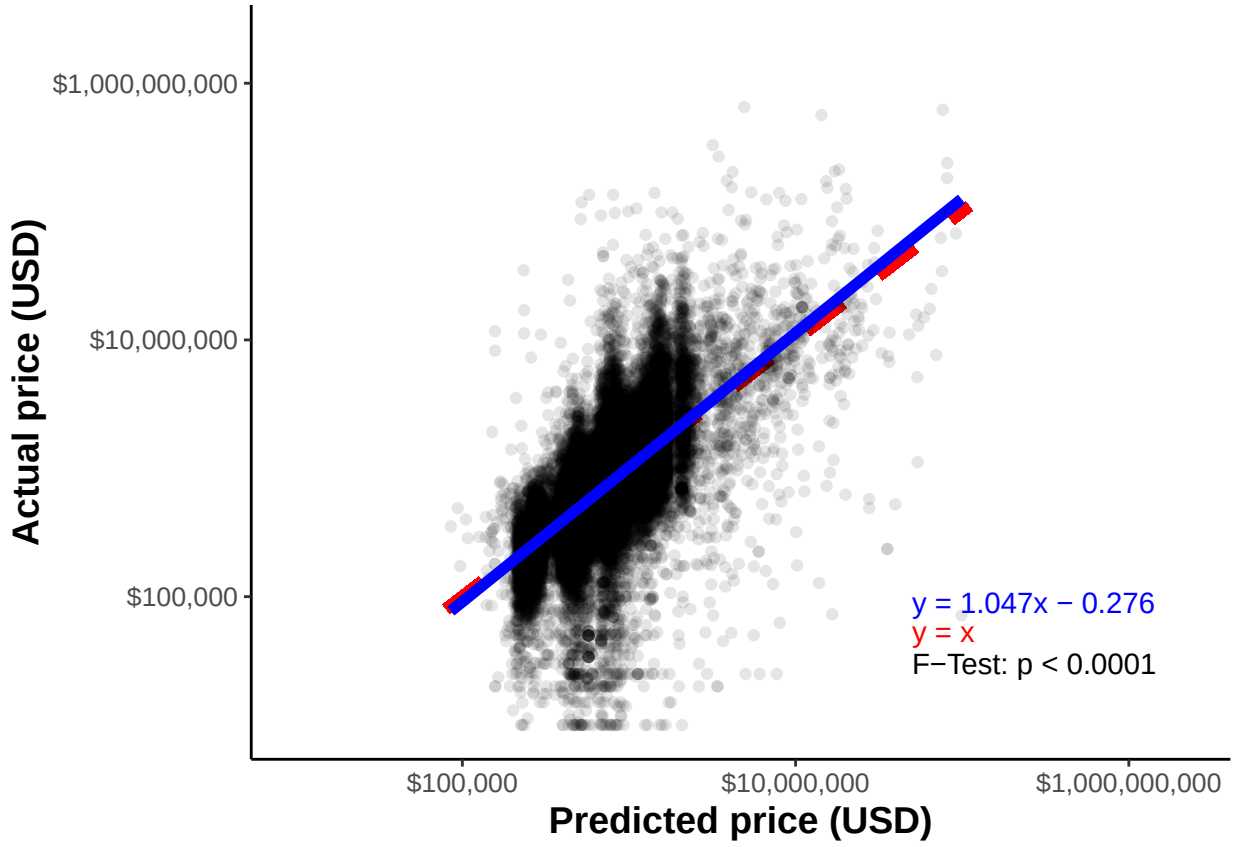
A training data set was created by randomly extracting 25% of records (n = 13,506) and the remaining 75% of records (n = 40,513) was used as a validation data set for the predictive algorithm. Finally, a function to calculate RMSE was developed to assess algorithm precision. The intital predictive model simply used mean of all property prices as the lone predictor variable. Effects of borough, number of residential units, number of commercial units, property type, land area, gross area, decade built, tax class, building class, day of week the property sold, and month the property sold were iteratively added and RMSE was calculated for each model to measure fit. The best fit model was applied to the validation data set to predict sale price. Accuracy was assessed by calculating the absolute value of the difference between the true sale price and the predicted sale price. Accuracy was measured by the proportion of predictions that were within 1 SE and 2 SE of the mean property price.

RESULTS

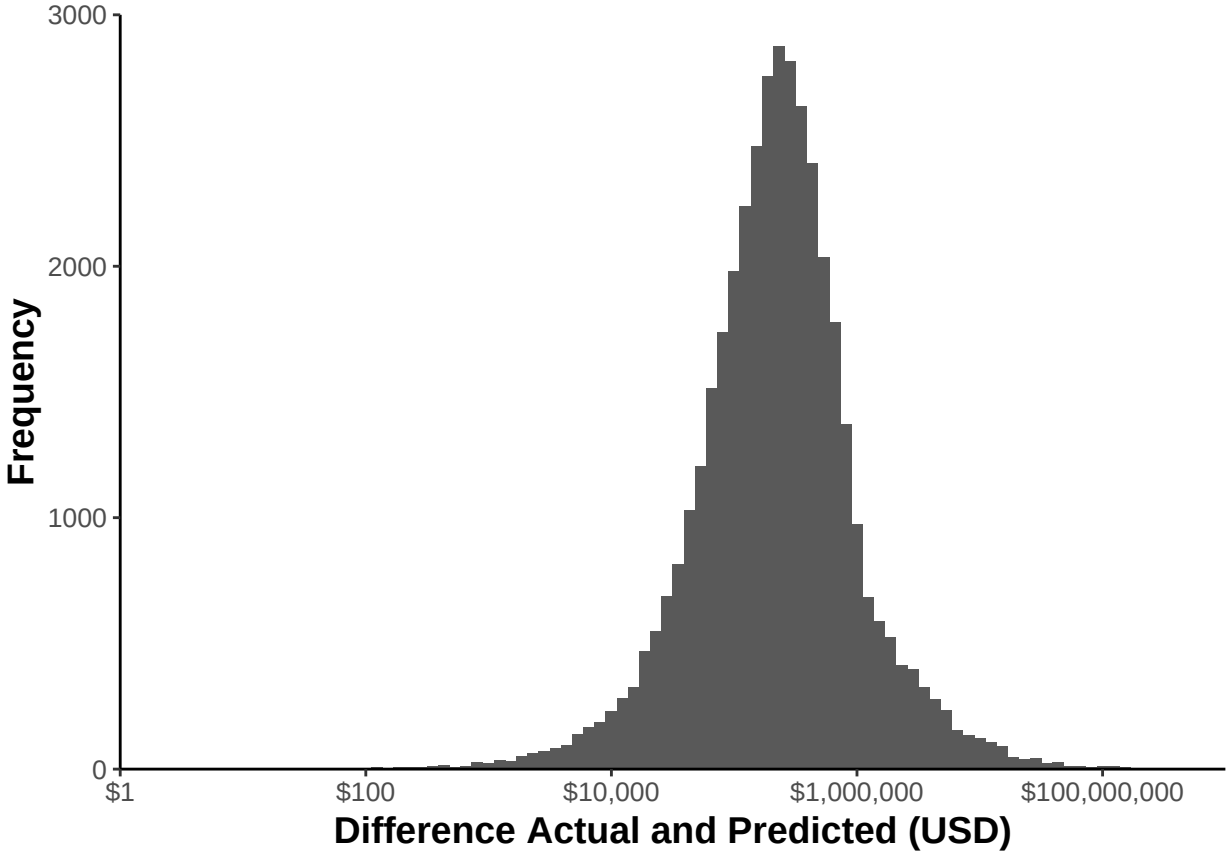
The model that accounted for effects of all variables resulted in the lowest RMSE (0.331). However, RMSE was primarily affected by borough and number of residential units. Additional factors added little predictive capability once borough and number of residential units were considered. Given the objective of this exercise is to minimize RMSE, the model containing all explanatory variables should be used to predict property sales price.

Model	RMSE
Average model	0.4308744
Borough Effect Model	0.3917485
B + Residential Units Effects Model	0.3578881
B + RU + Commercial Units Effects Model	0.3448101
B + RU + CU + Property Type Effects Model	0.3422820
B + RU + CU + PT + Land Area Effects Model	0.3421341
B + RU + CU + PT + LA + Gross Area Effects Model	0.3417773
B + RU + CU + PT + LA + GA + Decade Effects Model	0.3369585
B + RU + CU + PT + LA + GA + D + Tax Class Effects Model	0.3367216
B + RU + CU + PT + LA + GA + D + TC + Building Class Effects Model	0.3326964
B + RU + CU + PT + LA + GA + D + TC + BC + Day Sold Effects Model	0.3318301
B + RU + CU + PT + LA + GA + D + TC + BC + DS + Month Sold Effects Model	0.3311692

Accuracy of predicted sales price within 1 SE and 2 SE of mean property sale price were 10.5% and 20.6%, respectively. A linear regression model addressing predicted price relative to true price suggested that slopes differed and predicted price using this algorithm was not a suitable predictor for true sales price.



Further, a histogram of difference between actual and predicted price indicated a mode near $1,000,000 USD lending further evidence that the developed algorithm did not suitably predict true property sales price.



CONCLUSIONS

Data science skills including data organization, data cleansing, algorithm development, and data visualization were used in this exercise to predict property sales price in New York City between 1 September 2016 and 31 August 2017 from a large, publically available data set. Root mean squared error values decreased with inclusion of more explanatory variables. However, simply using borough and residential effects to predict property sale price may have been sufficient. Accuracy with the algorithm presented herein was lower than expected. This could be attributed to many factors, but may be associated with limited predictive power of available data. For example, factors related to subway proximity, available parking, or crime rate, may allow more accurate predictions of property sale value. Although the algorithm presented herein was not effective at prediction of property value, it demonstrates an approach that can be used to assess coarse property value categories. Additional parameters would need to be considered if accurately predicting property sale value was the primary objective.