FLIP ROBO

Housing Project

-Kundan

Introduction

Houses are one of the necessary need of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world's economy. It is a very large market and there are various companies working in the domain. Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases. Predictive modeling, Market mix modeling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. Our problem is related to one of such company.

Problem Statement

This is our US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. The data is provided in the CSV file below. The company is looking at prospective properties to buy houses to enter the market. We are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know:

- Which variables are important to predict the price of variable?
- How do these variables describe the price of the house?

Business Goal

You are required to model the price of houses with the available independent variables. This model will then be used by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand the pricing dynamics of a new market.

Conceptual background of domain problem

There are 1168 rows and 81 columns, The first things is that we have did was analyzing whole dataset and fetched those columns which are significantly correlated to sales price column which is our target columns and the second thing we did was building the best model by filling the null values with suitable values and label encoding it .

Review of literatures

The data for this project is been provided from the client side, apart from that to accomplish this project in better way went through several websites like geeksforgeeks, towards datascience and many more for effective visualization and also took the help of datatrained institude. Read books like data visualization using python, and did some research over stack overflow subject on the same topic.

Motivation for problem undertaken

The particular project is undertaken to advice our client regarding the factors that affect most for the sales price, Adding to that we would always like to see our client at top of chart in this reckless competition.

Analytical Problem Framing

Analytical modeling of the problem

Since all most all columns are of type categorical we label encoded them to make the machine understand, In this case the response variable was sales price so we label encoded all the variable after cleaning the dataset and we took the correlation heatmap of that and noted down which are the variables which highly affects the Sales Price.

Data Sources and their formats

The dataset what we are provided with consist of 81 columns which consist of both categorical and non categorical data types.

Software and tools used

We carried out this project in jupiter notebook of Anaconda Navigator For better visualization and for data processing we used libraries like pandas, matplotlib, seaborn and sklearn.For better visualization we used Tableau software.

Project Flow

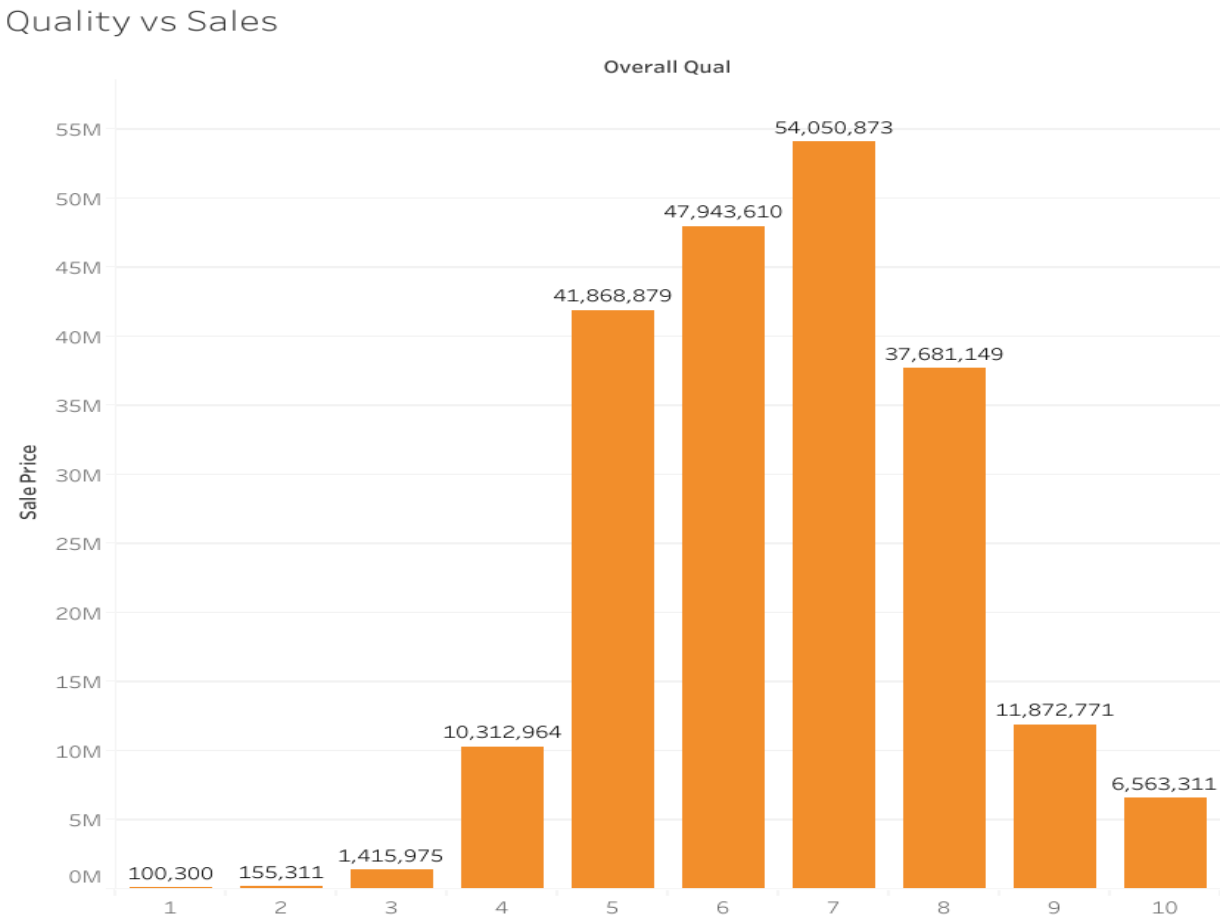
1. Importing the dataset
2. Data Processing
3. Exploratory Data Analysis
    a. Univariate Analysis
    b. Bi variate analysis
    c. Multivariate analysis

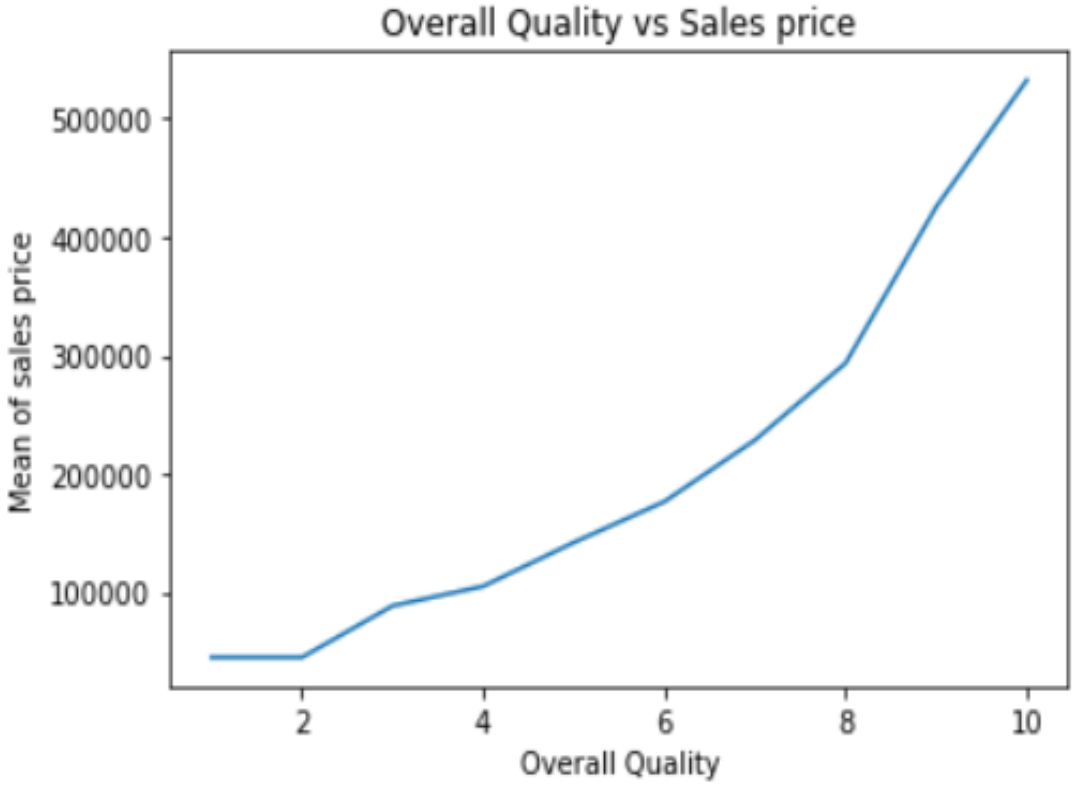
4. Data Cleaning
5. Model building

Information Derived from Exploratory Data Analysis

In this given dataset there are 81 columns or parameters, We mainly focused on what are the parameters which decides the sales price, There are 35 parameters out of all which has got a significant effect over deciding the sales price.

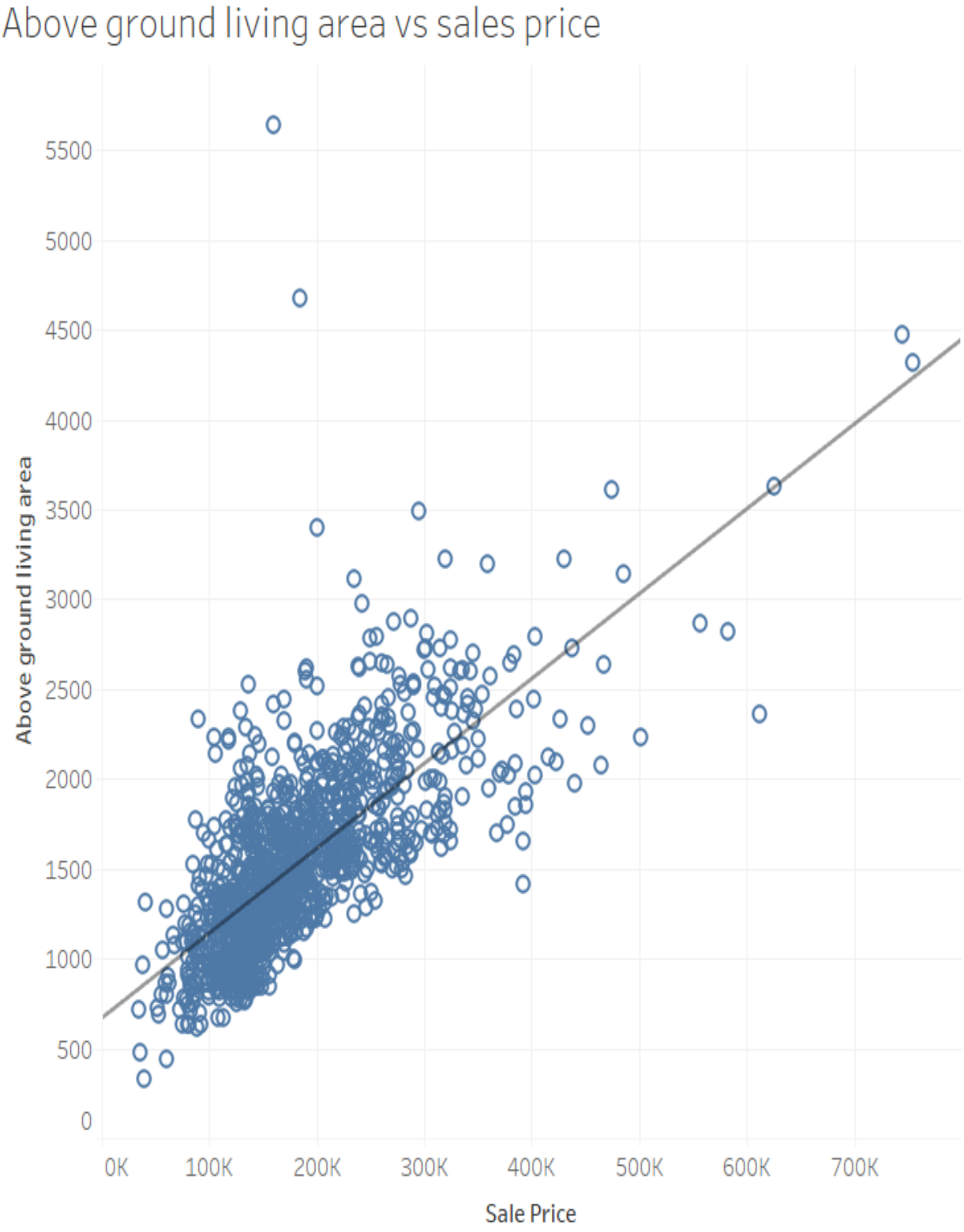
1. Sales Price vs Overall Quality





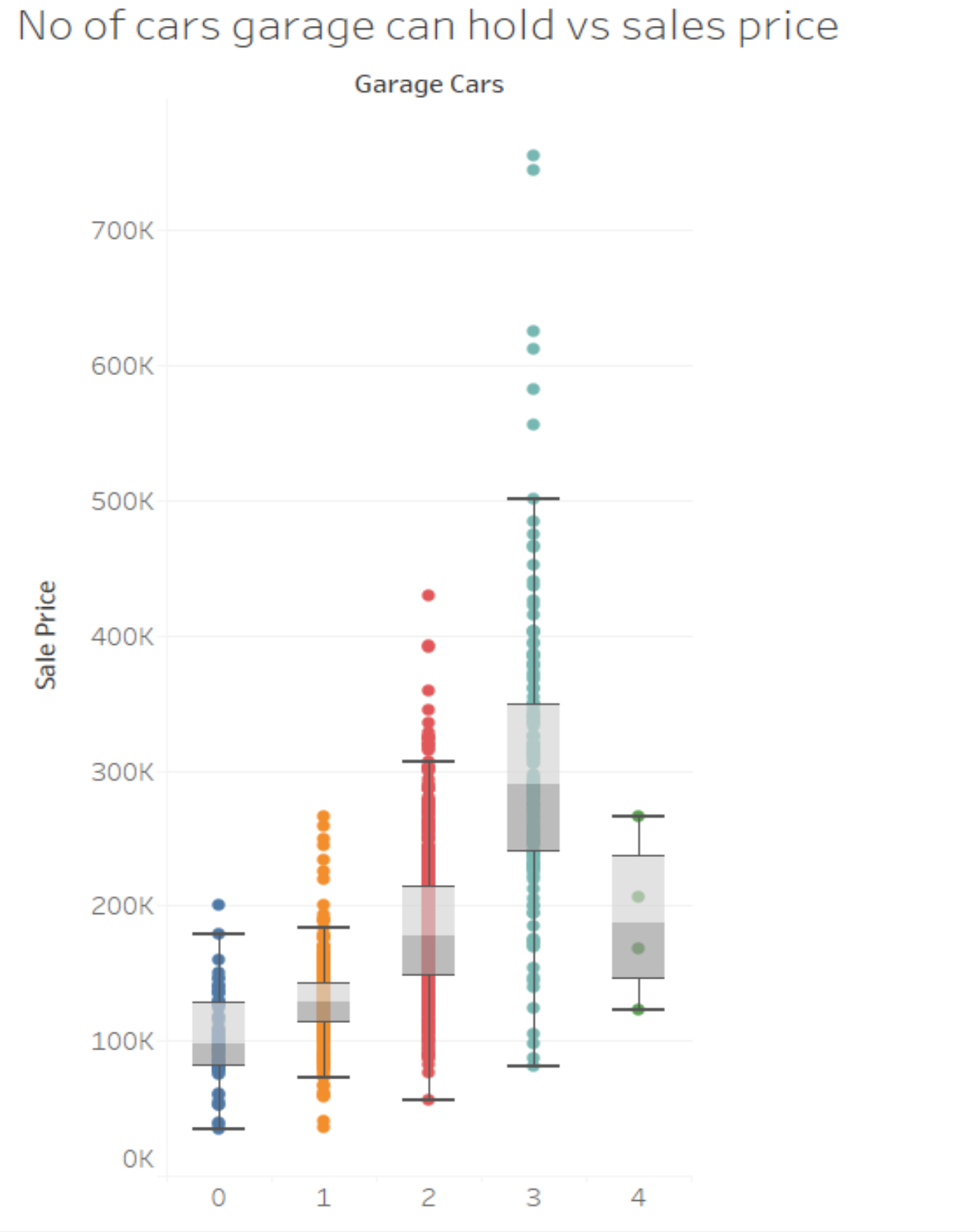
From the above graphs we can say that there is a linear correlation between the sales price and overall quality.

2.Above grade living area square feet vs Sales Price



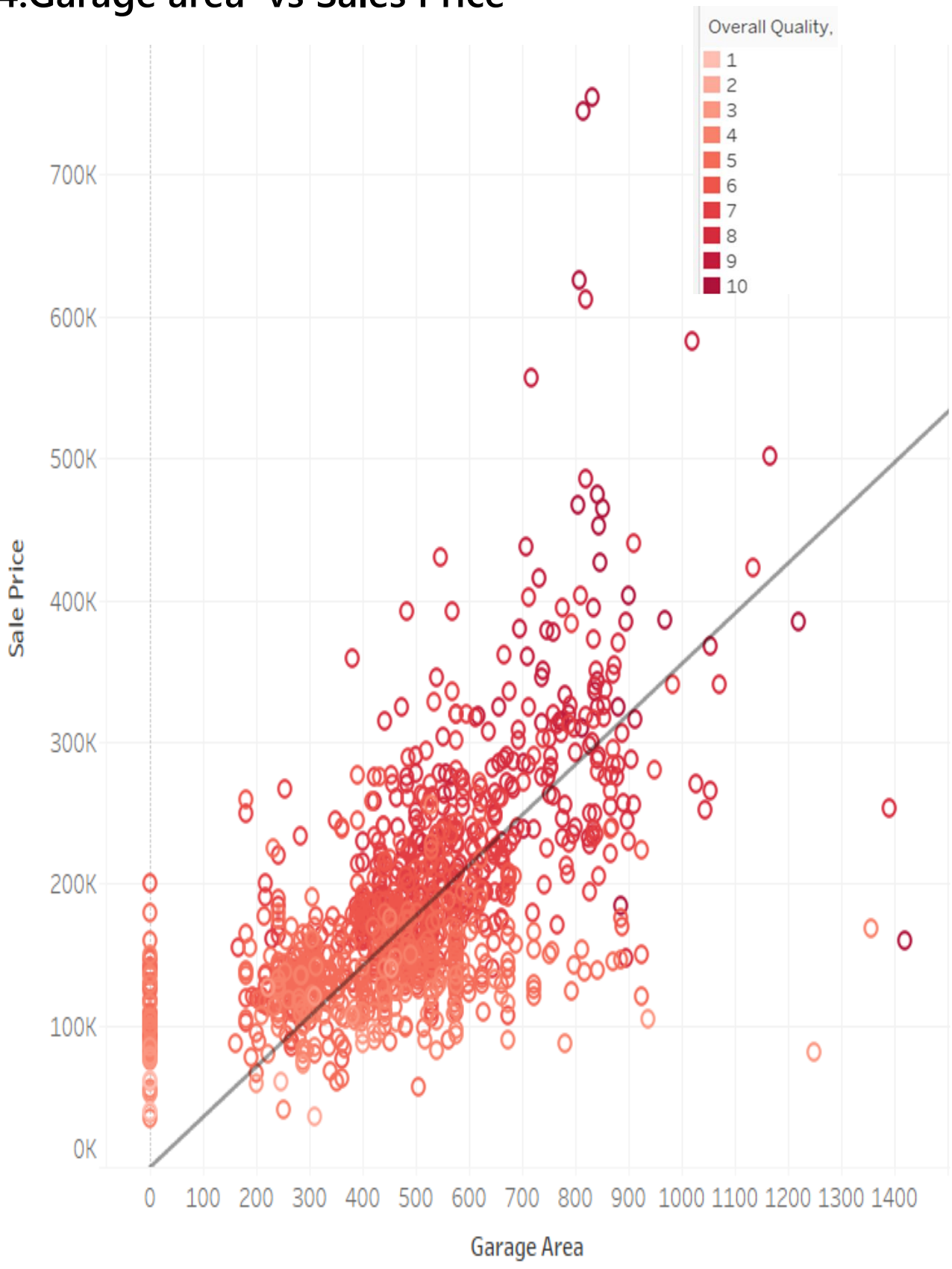
- The above ground living area and sales price variables vary linearly with each other.

3.No of cars that garage can hold vs Sales Price



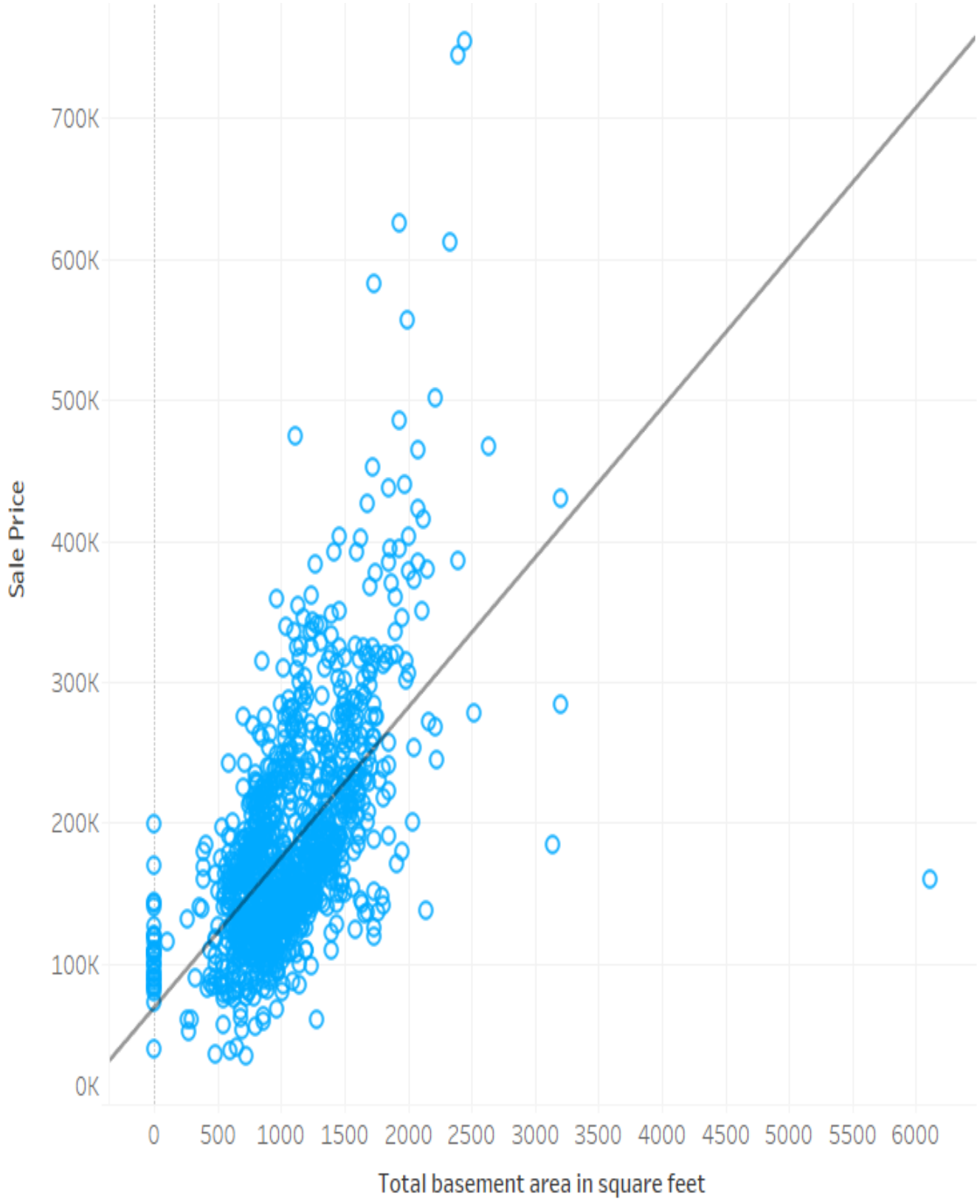
- ❑ As the garage size increases the sales price will automatically get increased.

4.Garage area vs Sales Price



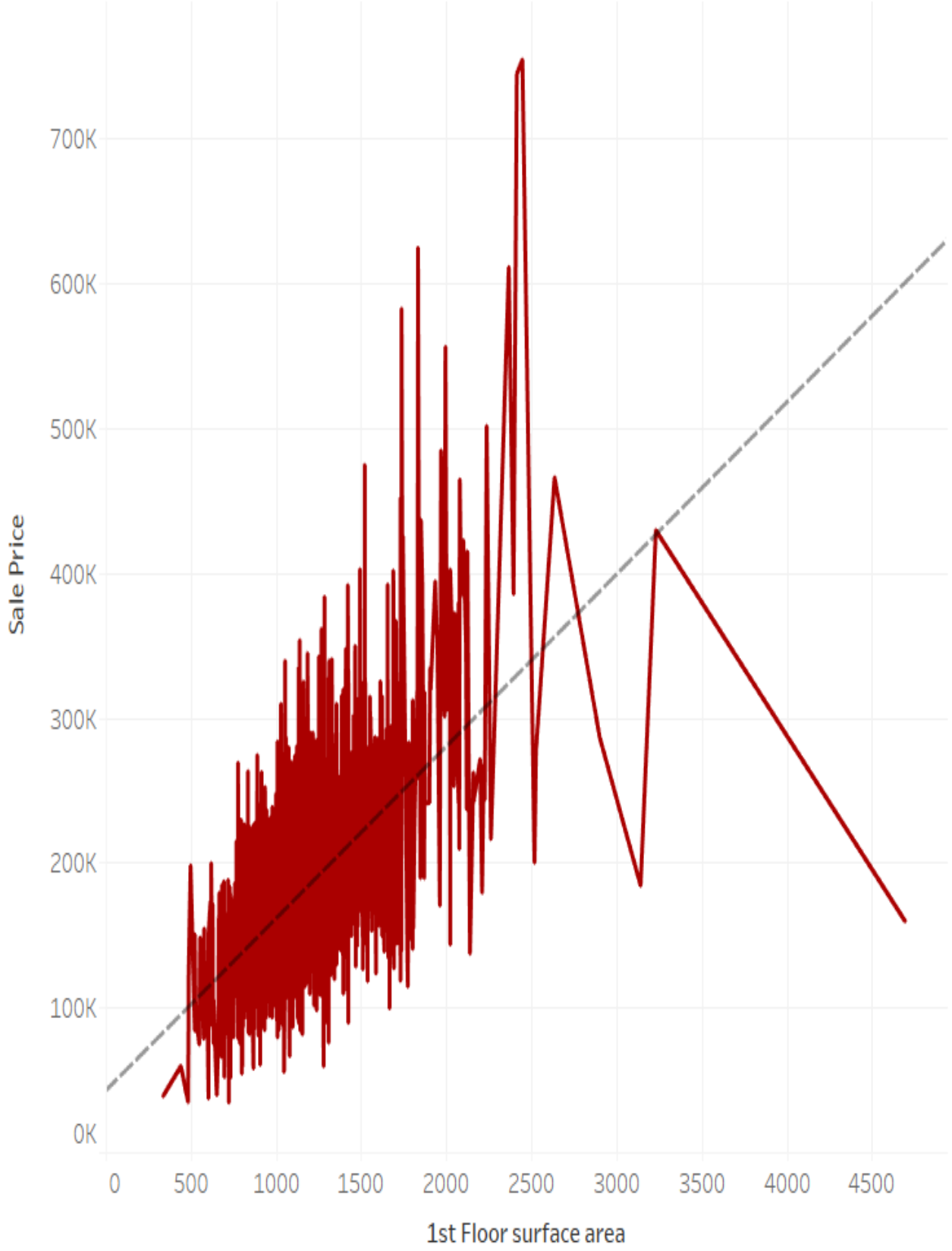
- The garage and sales price are linearly related.

6.Total basement area in sq ft vs Sales Price



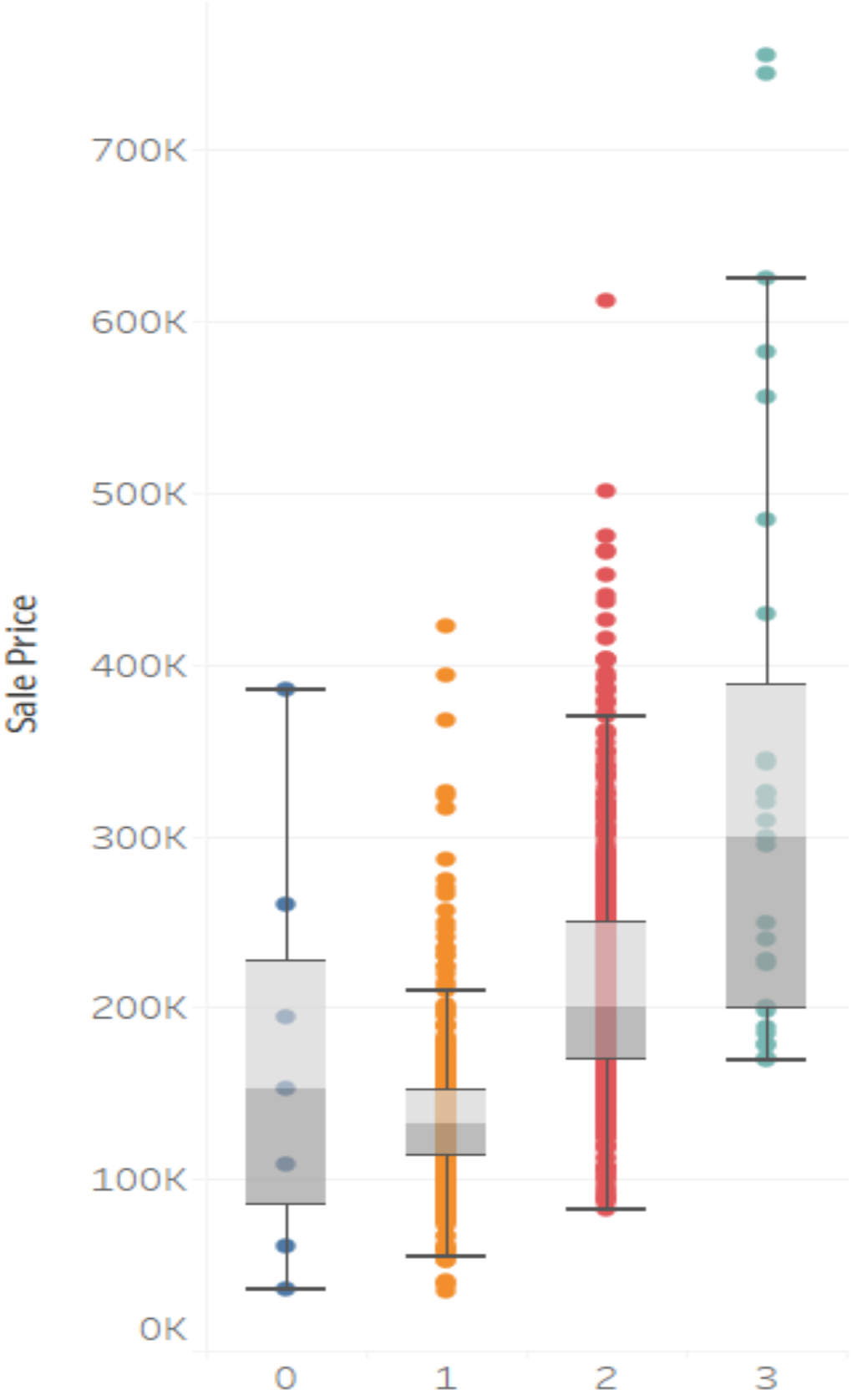
- The total basement area in sq ft and sales price are linearly related.

6. 1st Floor surface area vs Sales Price



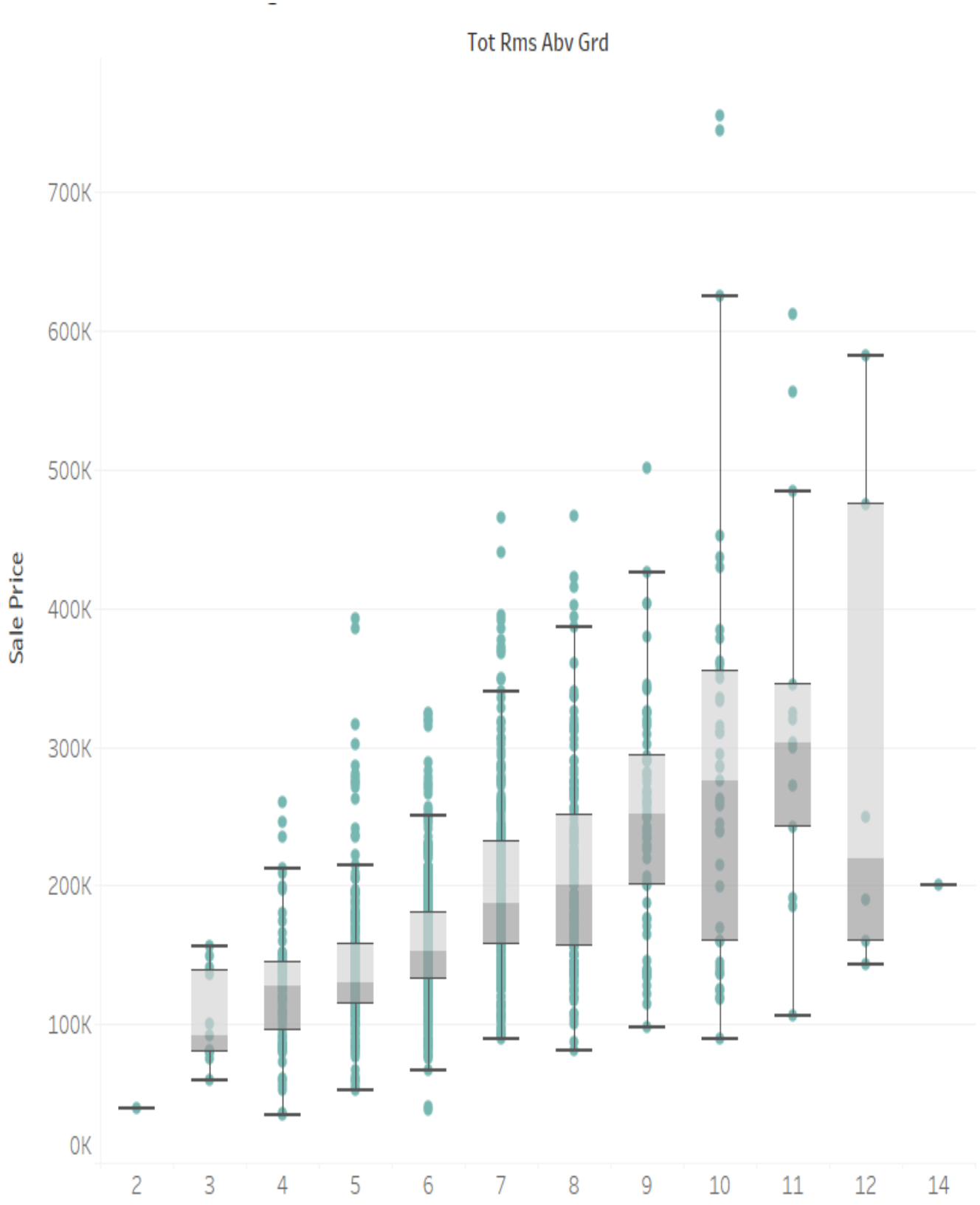
- There is a positive linear correlation between the 1st floor surface area and sales price.

7.Full bathrooms above ground vs Sales Price



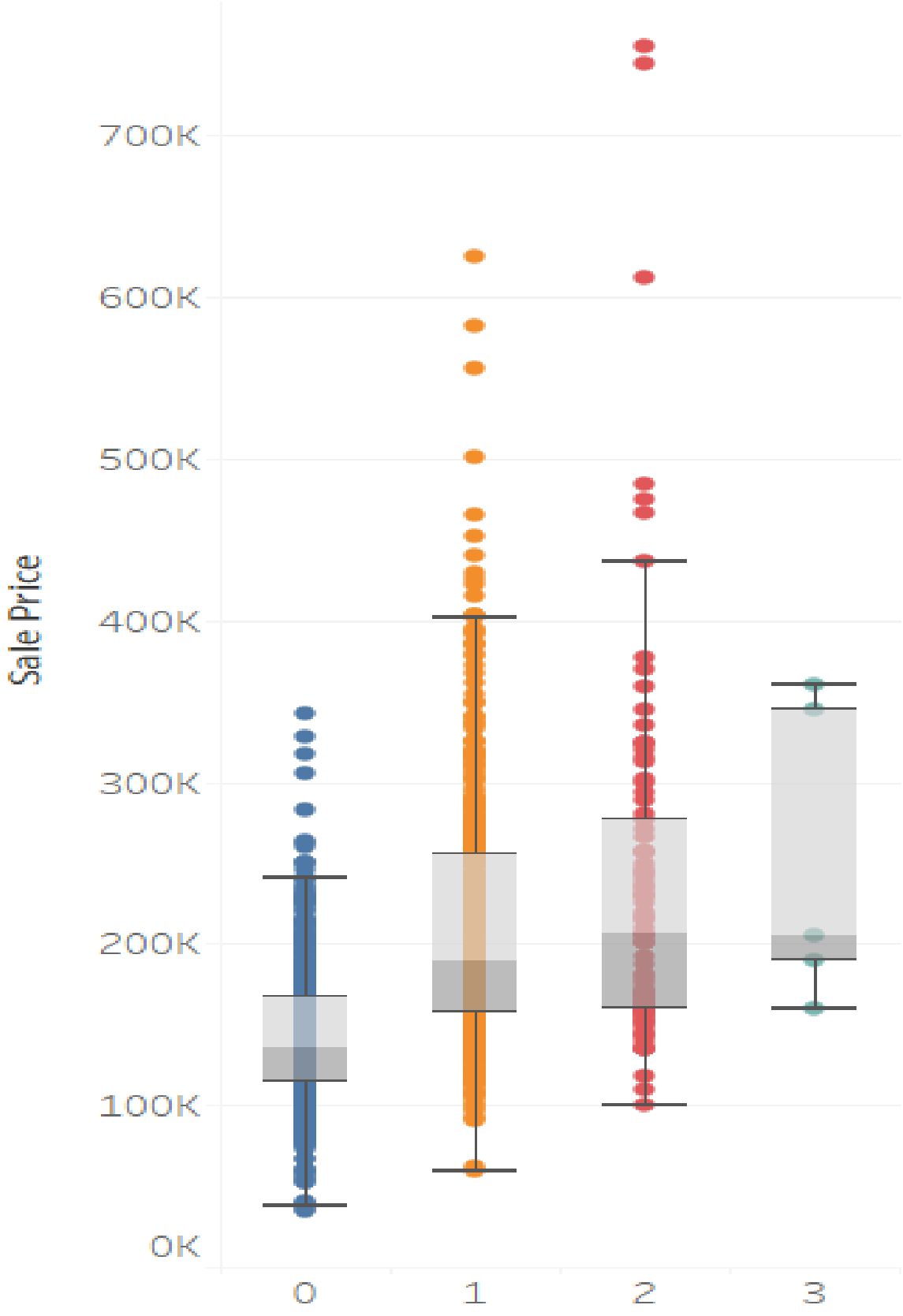
- Generally as number of full bathroom increases the sales price also going to be increased.

8. Total rooms above ground vs Sales Price



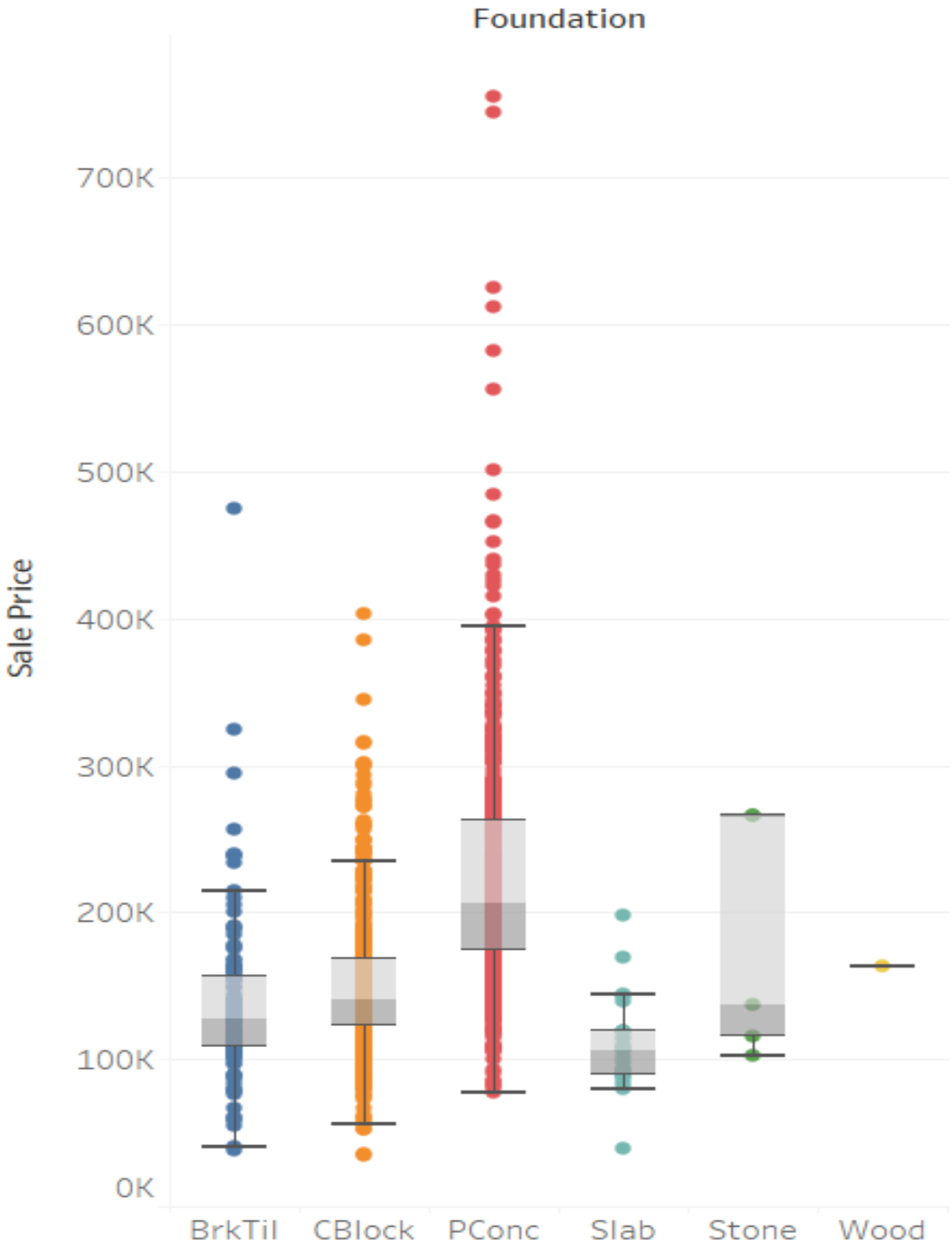
- As total rooms above the ground increases sales prices also going to increase.

9.Number of fire places vs Sales Price



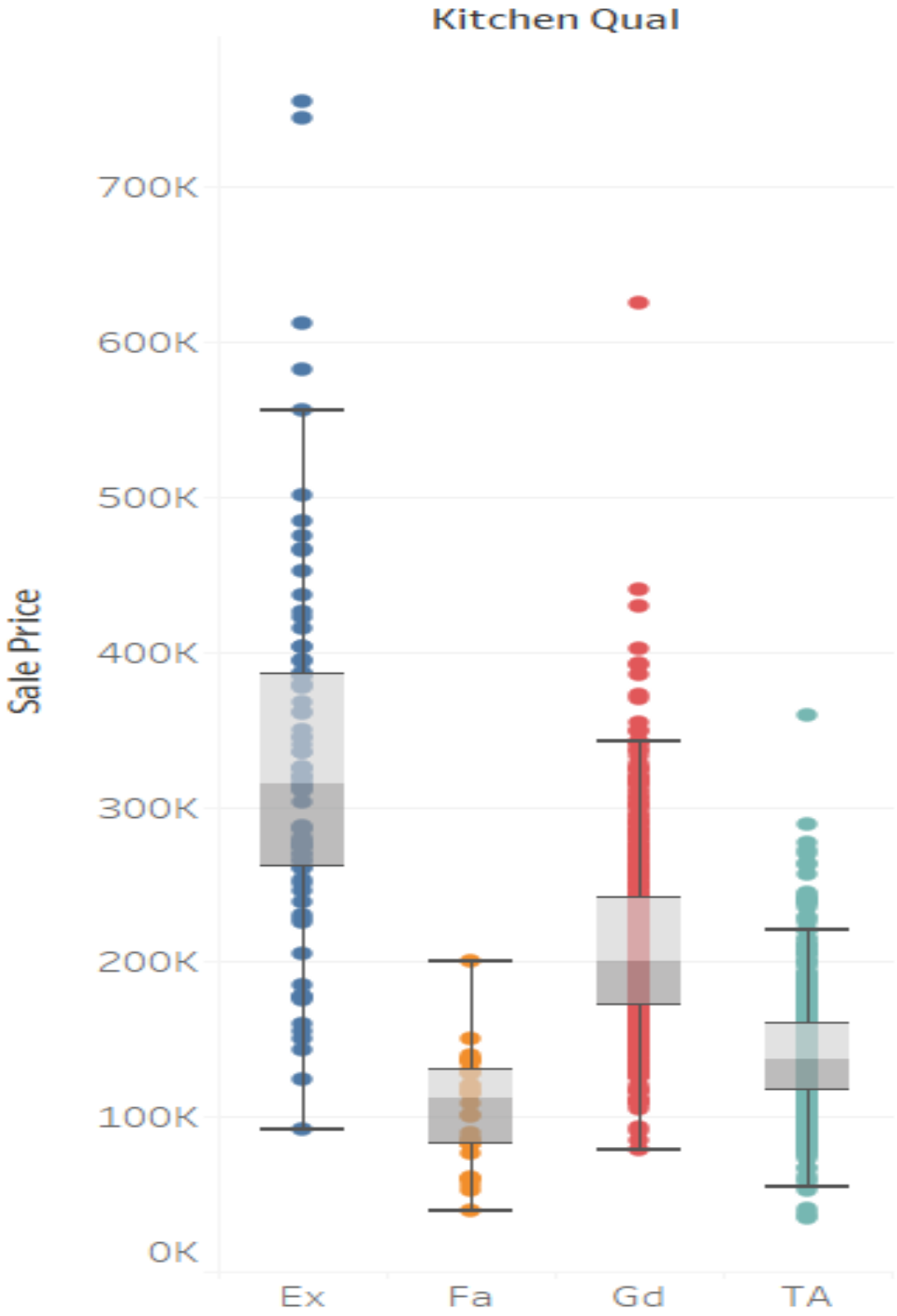
- Generally as number of fire places increases the sales price also going to be increased.

10. Type of foundation vs Sales Price



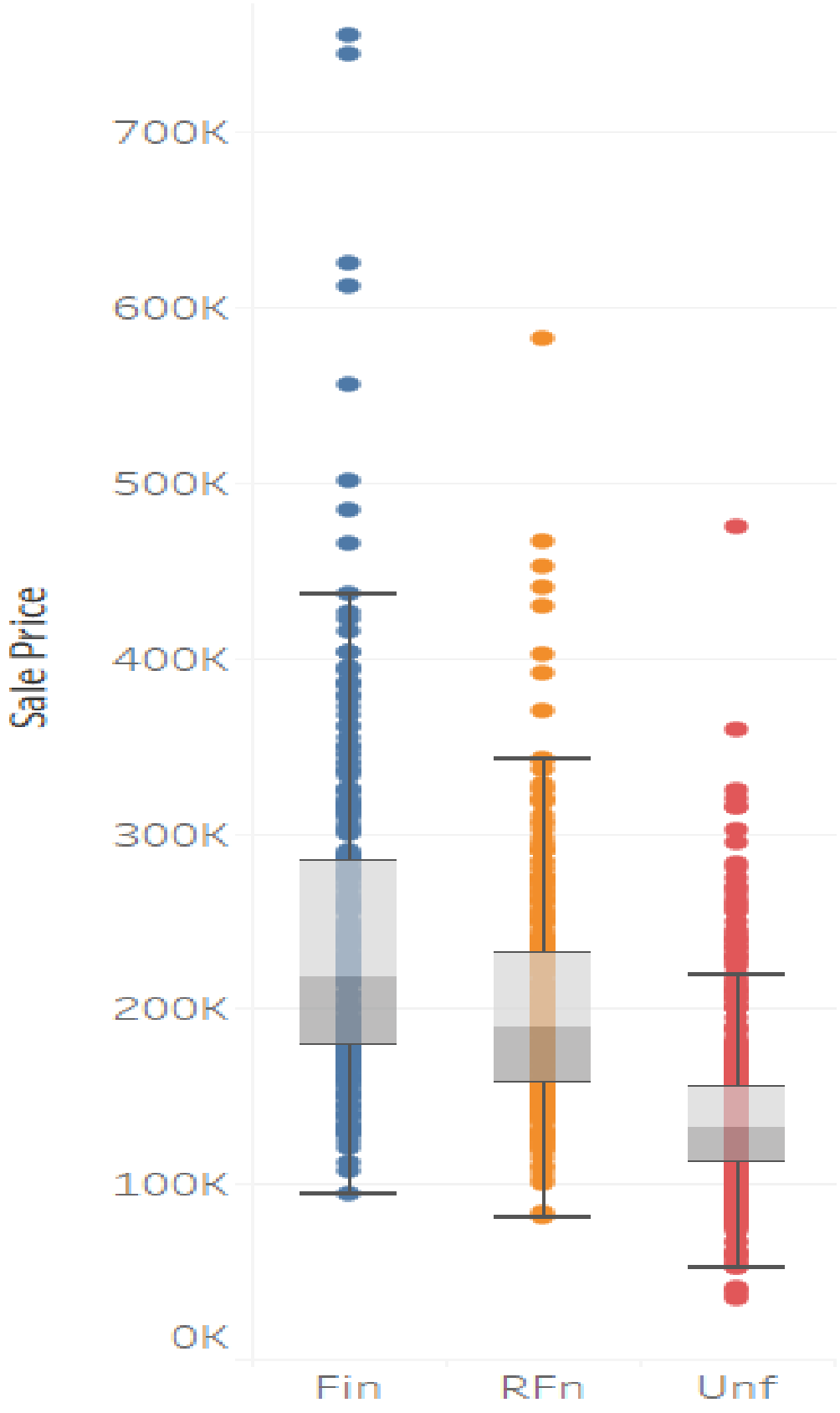
- If it's poured concrete foundation then its sales price will be higher than any other type of foundation followed by Cinder block, stone, brick and tail and wood.

11.Kitchen quality vs Sales Price



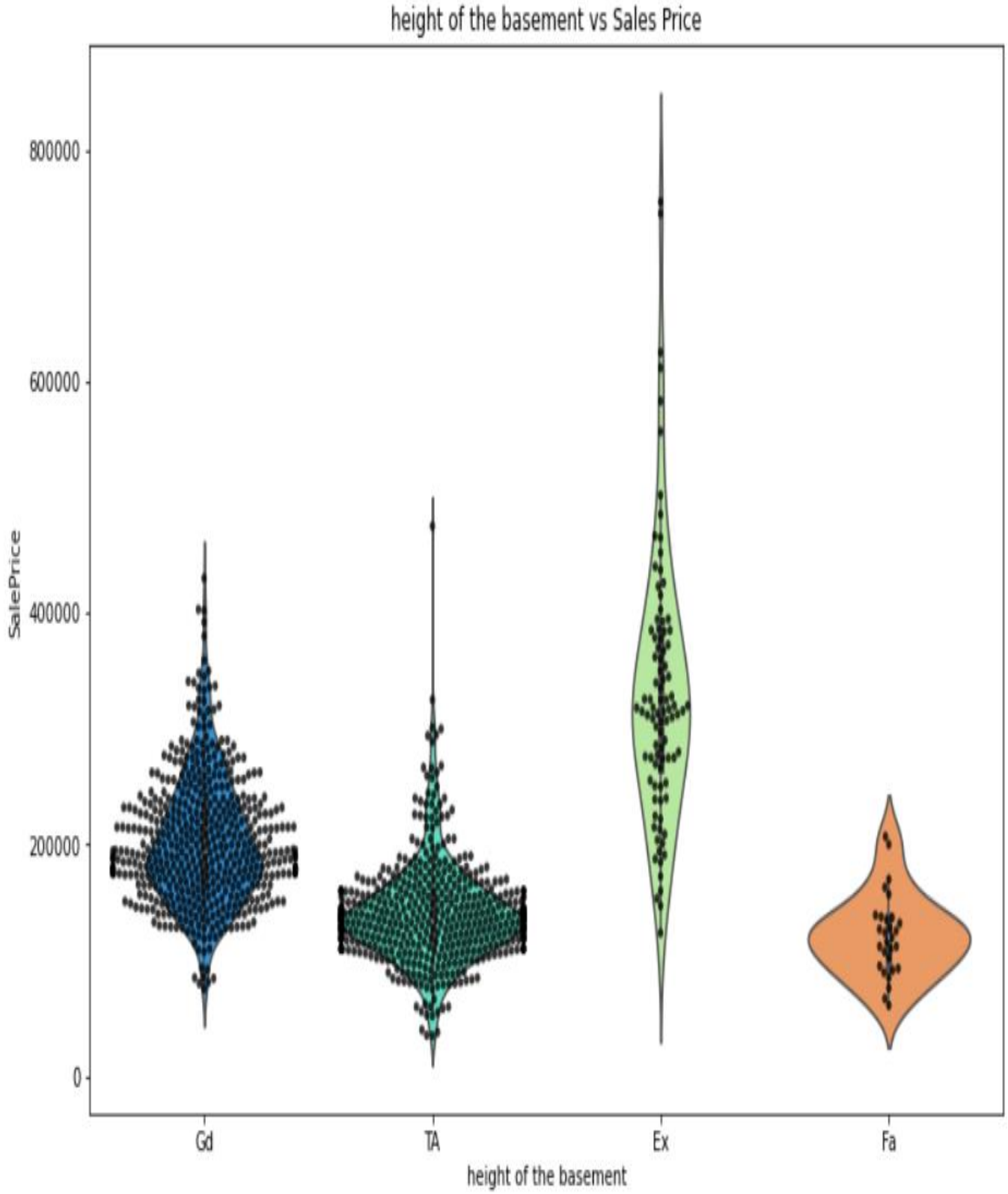
- The excellent kitchen quality will get sold for higher price, followed by good, average/typical, fair and poor.

12. Interior finish of the garage vs Sales Price



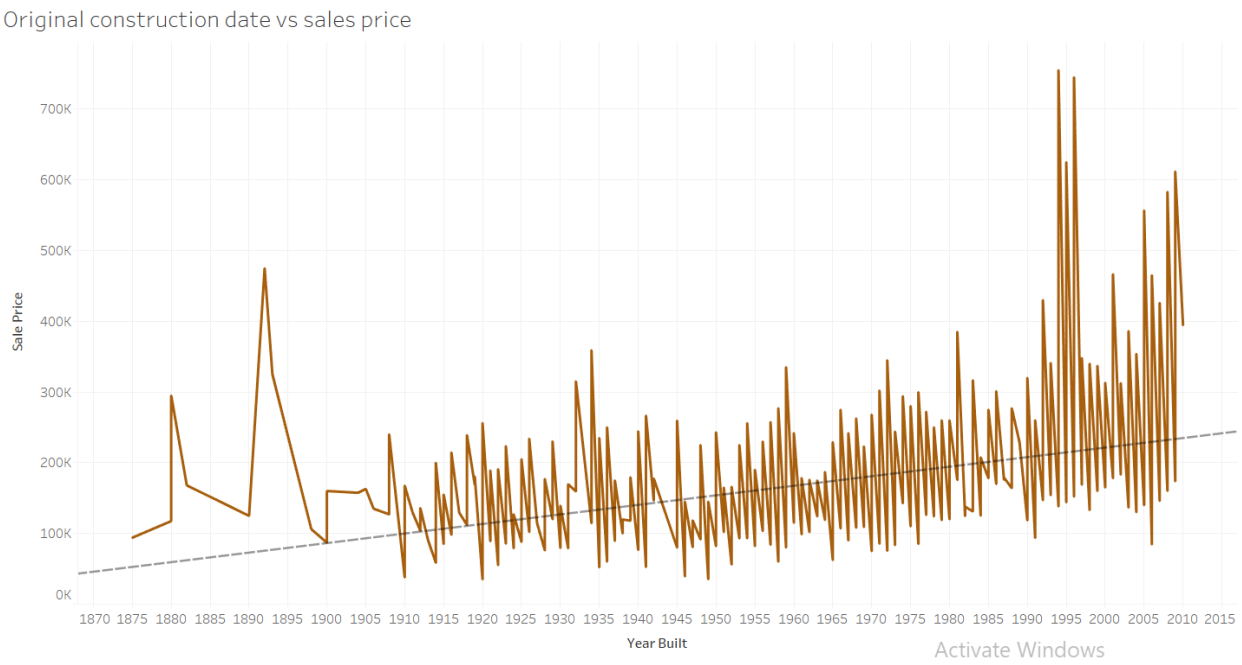
- If interior of garage is finished then it will have higher sales price followed by rough finish and unfinished.

13. Evaluates the height of the basement vs Sales Price



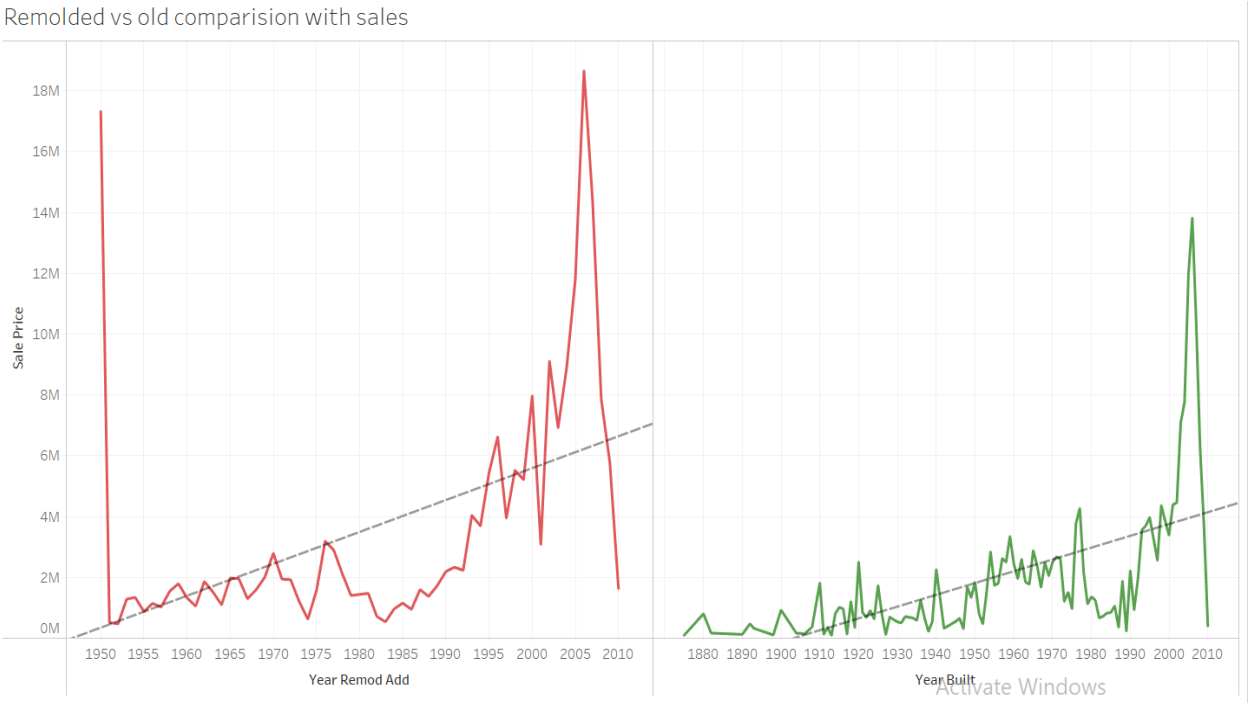
- The Excellent (100+ inches) basement will get sold for significantly high price followed by Gd-Good (90-99 inches),TA-Typical (80-89 inches)

14. Year in which house was built vs Sales Price



- ❑ We can see the trend that if the house was built in recent years then it will have high sales price.

15. Comparing the remolded and non remolded houses against sales price



- ❑ The remolded house will get sold for better price than non remolded house

Preparation of dataset for model building

- The dataset what we are provided with contains 81 columns and most of the columns are of categorical type and there are null values in the dataset.
- We have to do majorly two task to prepare the dataset for model, First we have to fill the null values and after that we have to label encode it
- The null values can be in categorical columns or in integer or float columns, If the null values in any columns exceeds by 63% then its better to remove entire column, If there are any null values in float columns just replace it with the median of the column and if there are any null values in the categorical column replace it with the mode of the column.
- After cleaning the null values label encode all the categorical columns.

Building the model

- In order to fetch the best suited model for this dataset we need to evaluate all major parameters regarding the linear regression models, we have to find the difference between cross validation score and accuracy , the least of the difference is considered as best model and we had hyper parameter tuned that.
- The accuracy ,cross validation score and their difference are follows.

Linear Model

```
#linear model
ln=LinearRegression()
ln.fit(x_train,y_train)
predln=ln.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predln)),3))
print('RMSE:',round(np.sqrt(mean_squared_error(y_test,predln)),3))
print('mean absolute error:',round((mean_absolute_error(y_test,predln)),3))
```

```
r2 score is : 0.896
RMSE: 23402.494
mean absolute error: 17891.43
```

Lasso Model

```
#lasso model
ls=Lasso(alpha=2.5)
ls.fit(x_train,y_train)
predls=ls.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predls)),3))
print('RMSE:',round(np.sqrt(mean_squared_error(y_test,predls)),3))
print('mean absolute error:',round((mean_absolute_error(y_test,predls)),3))
```

```
r2 score is : 0.896
RMSE: 23386.629
mean absolute error: 17881.855
```

Ridge Model

```
#Ridge model
rd=Ridge(alpha=2.5)
rd.fit(x_train,y_train)
predrd=rd.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predrd)),3))
print('RMSE:',round(np.sqrt(mean_squared_error(y_test,predrd)),3))
print('mean absolute error:',round((mean_absolute_error(y_test,predrd)),3))
```

```
r2 score is : 0.897
RMSE: 23219.907
mean absolute error: 17763.64
```

Elastic Net

```
#ElasticNet model
enr=ElasticNet(alpha=0.1)
enr.fit(x_train,y_train)
predenr=enr.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predenr)),3))
print('RMSE:',round(np.sqrt(mean_squared_error(y_test,predenr)),3))
print('mean absolute error:',round((mean_absolute_error(y_test,predenr)),3))
```

```
r2 score is : 0.898
RMSE: 23192.433
mean absolute error: 17479.941
```

RansacRegressor

```
ran = RANSACRegressor(base_estimator=LinearRegression(), max_trials=100)
ran.fit(x_train, y_train)
predran=ran.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predran)),3))
print('RMSE:',round(np.sqrt(mean_squared_error(y_test,predran)),3))
print('mean absolute error:',round((mean_absolute_error(y_test,predran)),3))
```

```
r2 score is : 0.362
RMSE: 57889.26
mean absolute error: 31495.336
```

Random Forest Regressor

```
rf = RandomForestRegressor(n_estimators=100)
rf.fit(x_train, y_train)
predrf=rf.predict(x_test)
print('r2 score is :',round((r2_score(y_test,predrf)),3))
print('RMSE:',round(np.sqrt(mean_squared_error(y_test,predrf)),3))
print('mean absolute error:',round((mean_absolute_error(y_test,predrf)),3))
```

```
r2 score is : 0.899
RMSE: 22985.224
mean absolute error: 16174.083
```

Cross Validation

```
models=[ln,ls,rd,enr,ran,rf]
for m in models:

    score=cross_val_score(m,x2,y2,cv=5)
    print(m,'score is:')
    print(round((score.mean()),3))
    print('\n')
```

```
LinearRegression() score is:
0.833


Lasso(alpha=2.5) score is:
0.834


Ridge(alpha=2.5) score is:
0.838


ElasticNet(alpha=0.1) score is:
0.845


RANSACRegressor(base_estimator=LinearRegression()) score is:
0.578


RandomForestRegressor() score is:
0.854
```

Sl.No	Model Name	Difference
1.	Linear Regression	0.063
2.	Lasso regression	0.056
3.	Ridge Regression	0.059
4.	Elastic Net	0.053
5.	RANSACRegressor	-0.216
6.	Random Forest	0.045

The best model among all the above is Random Forest regressor which has got the least difference of 0.045

Hyper Parameter Tuning

Hyper parameter tuning

```
rf=RandomForestRegressor()
grid_param={
    'criterion':['mse','mae'],

    'max_depth':[10,20,30,40,50],
    'max_features':['auto', 'sqrt', 'log2'],
    'min_samples_split':[2,5,10,15,20],
    'bootstrap':[True,False]
}

gd_sr=GridSearchCV(estimator=rf,
                   param_grid=grid_param,
                   scoring='r2',
                   cv=5)

gd_sr.fit(x2,y2)

best_parameters=gd_sr.best_params_
print(best_parameters)
best_result=gd_sr.best_score_
print(best_result)
```

```
{'bootstrap': False, 'criterion': 'mse', 'max_depth': 50, 'max_features': 'sqrt', 'min_samples_split': 2}
0.8643391899273511
```

```
rf1=RandomForestRegressor(n_estimators=100,criterion='mse',max_depth=50,max_features='sqrt',min_samples_split=2,bootstrap=False)
```

Conclusion

- **Key findings and the conclusion of the study**
  To excel in this competitive world the company should see that the overall quality of the house is excellent, followed by 1st floor and second floor s area, and all the points above mentioned are so cruicial.
- **Learning outcomes from the project**
  If we have too many variables then no need to look at every corner for irrelevant information, We should always keep in mind what we are doing and the aim of the project.
- **The best model what we have got is Random Forest regressor with an accuracy of 90%**
- **To enter into new market the above mentioned points are so cruicial and it has to be given first priority since they influence sales price in an significant manner compared to all other variables.**

Thank you