STAT 37400 Final Project

Lijing Wang

Description of the data

The dataset used in this report is obtained from Kaggle. It contains house sale prices for King County, and includes homes sold between May 2014 and May 2015.

The full dataset includes 21613 observations. The input and output variables are defined as follows:

Inputs:
· `bedrooms`: number of bedrooms
· `bathrooms`: number of bathrooms
· `sqft_living`: the total house square footage of the house
· `sqft_loft`: lot size of the house
· `floors`: number of floors
· `waterfront`: whether the house is located near waterfront
· `view`: view of the house
· `condition`: condition of the house
· `grade`: construction quality of the house
· `sqft_basement`: size of the basement
· `sqft_above`: `sqft_living - sqft_basement`
· `yr_built`: time of building the house
· `yr_renovated`: time of renovating the house
· `zipcode`: zipcode of the area the house belongs to
· `lat`: latitude of the house
· `long`: longitude of the house
· `sqft_living15`: the average house square footage of the 15 closest houses
· `sqft_lot15`: the average lot square footage of the 15 closest houses

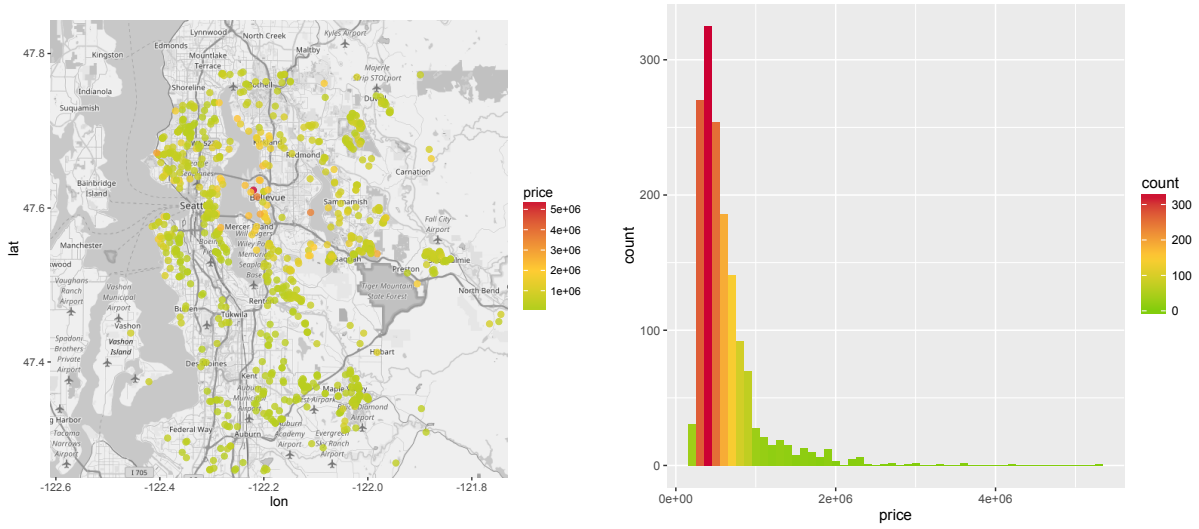
Output:
· `price`: sale price of the house

In the following part we'll mainly focus on houses that are built after 2000 and sold in 2015. There're 1542 houses satisfying the conditions above.

The objective of our analysis is to find a robust estimation of the housing price. First, we assume all the inputs are the potential explantory variables for estimating the price. Then we'll implement parametric analysis (simple linear regression) and nonparametric (local linear regression/local polynomial regression). And we'll compare the results using these three methods.

First we label the location and price of the house on a map to get some intuitive idea about housing price pattern. As we can see, houses near the lake generally have a higher price than houses in other places. This is especially obvious around Lake Washington area.

Then we look at the scale of the response variable and see if we need to do any kind of transformation. The histogram of housing price is given as below:



The price distribution is right-skewed. Most of the data points are less than 1,000,000, However, some can be as large as 5,000,000. In order to make them on the same scale, consider taking log transformation to the response variable.

In the following part, randomly divide the data into two equally-sized parts. The first part is the training dataset, we'll build the model using training dataset and test our model on the remaining part, which is our test dataset.

Parametric Model

Fit the data using simple linear regression. For the linear model, we have the following assumption:

· The design matrix $X_{n\times p}$ has full column rank.
· The error terms ε_i's have mean zero
· The error terms ε_i's have constant variance
· The error terms ε_i's are independent
· Assume no interactions between predictors

The model is given as below:

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

where $\varepsilon_i \sim (0, \sigma^2)$.

In this case, since `sqft_above` is defined to be the difference between `sqft_living` and `sqft_basement`. We can just include two of them in our design matrix. (Here I choose `sqft_above` and `sqft_living`) We also eliminated `zipcode` from the design matrix because `long` and `lat` can give us a more detailed information about the geographical location of the house.

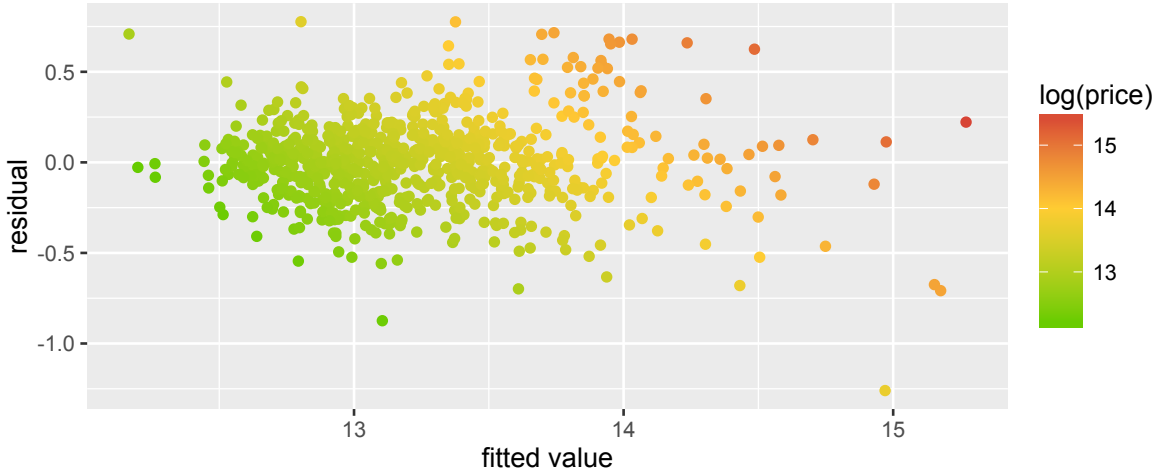
Regress log(`price`) on all the predictors first and use backward selection to remove predictors that are not significant. (p-value less than 0.05)

The model trained by the data contains 6 variables. The coefficients are given in the following table:

	Estimate	Std. Error	t value	p-value
Intercept	-65.24	5.026	-12.981	$< 2 \times 10^{-16}$
`sqft_living`	0.00021	0.000012	17.505	$< 2 \times 10^{-16}$
`waterfront`	0.7540	0.1429	5.278	1.7×10^{-7}
`view`	0.0451	0.01299	3.472	0.000545
`grade`	0.1693	0.01141	14.839	$< 2 \times 10^{-16}$
`yr_built`	0.0074	0.00196	3.762	0.000181
`lat`	1.298	0.06462	20.087	$< 2 \times 10^{-16}$

The adjusted R^2 for the model is 0.8004. The estimated $\hat{\sigma}$ is 0.2332.

Plot the fitted value against residual to see if the residuals are of constant variance. As the plot shows, the residuals spread evenly around x-axis. No obvious sign of non-constant variance is detected in the plot.



Therefore, the final model used to predict the housing price is

$$\begin{aligned}\log(\hat{price}) = &-65.24 + 0.00021\texttt{sqft\_living} + 0.754\texttt{waterfront} + 0.00451\texttt{view} + 0.169\texttt{grade} \\ &+ 0.00737\texttt{yr\_built} + 1.298\texttt{lat}\end{aligned}$$

The training error for the model is calculated by $\frac{1}{n}\sum_{i=1}^{n}(y-\hat{y})^2$. In this case, 0.05388 ($\log(\texttt{price})$ as response). The test error for the remaining 771 observations is 0.04414.

Nonparametric Model

A. Local Linear Regession

In this part we use local linear regression to fit the data. The model is described as below:

Let x be some fixed point at which we want to estimate $r(x)$. For value u which belongs to the neighborhood of x, define:

$$P_x(u, a) = a_0 + a_1(u - x)$$

Then we have

$$r(u) \approx P_x(u, a)$$

We estimate $a = (a_0, a_1)^\intercal$ by choosing $\hat{a} = (\hat{a}_0, \hat{a}_1)^\intercal$ to minimize the locally weighted sum of square

$$\sum_{i=1}^{n} w_i(x)(y_i - P_x(x_i, \hat{a}))^2$$

where $w_i(x)$ is determined by Gaussian kernel.

Due to computation limit, we'll use the 6 variables selected by simple linear regression to run the local linear regression.

The fitted degrees of freedom is 15.62. The estimated $\hat{\sigma}$ is 0.228.

The training error for the model is calculated by $\frac{1}{n}\sum_{i=1}^{n}(y-\hat{y})^2$. In this case, 0.0505, slightly smaller than that of simple linear regression . The test error for the remaining 771 observations is 0.043998, which is also slightly smaller than that of simple linear regression.

B. Local Polynomial Regession

Similar to local linear regression, let x be some fixed point at which we want to estimate $r(x)$. For value u which belongs to the neighborhood of x, define:

$$P_x(u, a) = a_0 + a_1(u - x) + \frac{a_2}{2!}(u - x)^2 + \cdots + \frac{a_p}{p!}(u - x)^p$$

Estimate $a = (a_0, a_1, \cdots, a_)^\intercal$ by choosing $\hat{a} = (\hat{a}_0, \hat{a}_1, \cdots, \hat{a}_p)^\intercal$ to minimize the locally weighted sum of square

$$\sum_{i=1}^{n} w_i(x)(y_i - P_x(x_i, \hat{a}))^2$$

where $w_i(x)$ is determined by Gaussian kernel.

Use the 6 variables selected by simple linear regression to run the local polynomial regression. The fitted degrees of freedom is 43.079. The estimated $\hat{\sigma}$ is 0.211.

The training error for local polynomial regression is calculated by $\frac{1}{n}\sum_{i=1}^{n}(y-\hat{y})^2$. In this case, 0.04148, which is significantly smaller than that of simple linear regression and local linear regression. The test error for the remaining 771 observations is 0.03993, which is also smaller than that of the previous two models.

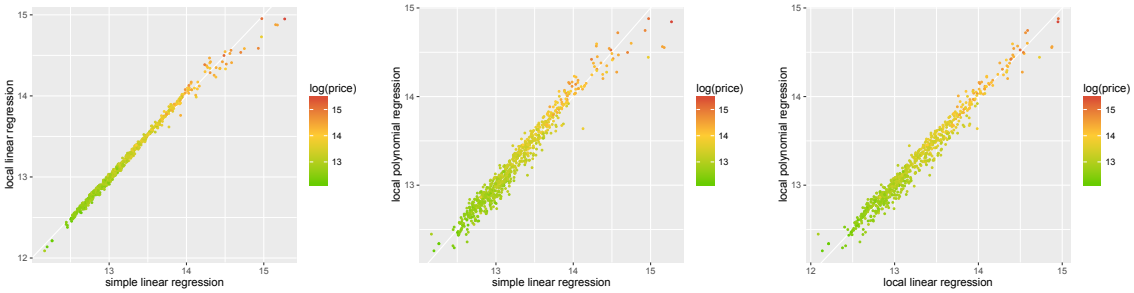
Model Comparison

The training errors and testing errors are given as below:

model	simple linear regression	local linear regression	local polynomial regression
training error	0.05388	0.05051	0.04148
testing error	0.04414	0.04400	0.03993

As we can see, nonparametric methods tend to have smaller risk comparing to parametric methods.

The following plots shows the difference between fitted values from different models



The fitted results of simple linear regression and local linear regression are very similar. And simple linear regression tends to have a larger fitted value than local linear/polynomial regression when the fitted value itself is relatively large.

Generally speaking, simple linear regression are simple and easy to compute. But it has too many assumptions and these assumptions might be too strict for data from real world. Nonparametric models require less assumptions, but it is more complicated and requires more time to compute.