Introduction to Statistical Learning

Instructor: Nhat Ho

The University of Texas, Austin

Outline

- Definition of statistical learning
- Parametric versus nonparametric methods
- Assessing model accuracy
- Train/ test set
- Bias and variance trade-off

Definition of Statistical Learning

- Y: quantitative response
    - Example: Income, house's price, etc.
- $X_{.1}, \ldots, X_{.p}$: p different predictors
    - Example: Education, age, temperature, etc.
- Assume that Y and $X = (X_{.1}, \ldots, X_{.p})$ are related through the following equation:

$$Y = f(X) + \varepsilon$$

- f is some fixed but **unknown** function
- ε is random error term and independent of X

Definition of Statistical Learning

- $Y = f(X) + \varepsilon$
- f represents the information that X provides about Y

- **Definition**: Statistical Learning refers to a set of techniques/ approaches to estimate the function f

- **Questions**: Why should we estimate f and what are the techniques to estimate f ?

Real Estate Example

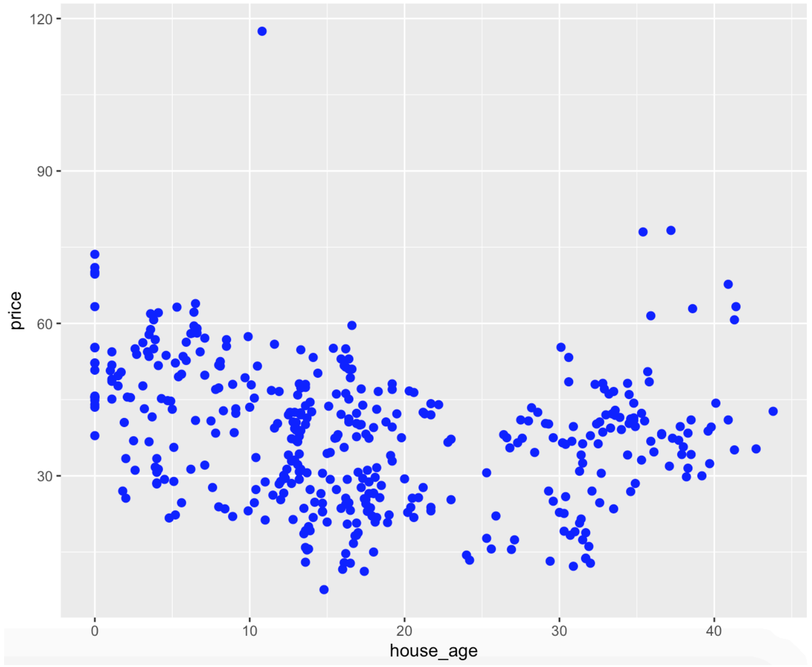
- Open the data “Real_estate.csv” (Can be downloaded here: https://drive.google.com/drive/u/0/folders/1JY1yE_9oujj9fgpTlquLIOlTTvrQQQ3z)
- Data source: Kaggle (https://www.kaggle.com/quantbruce/real-estate-price-prediction)
- The data is about predicting real estate price (response variable) based on some predictive variables (predictors), such as house age, distance to the nearest station, number of convenience stores

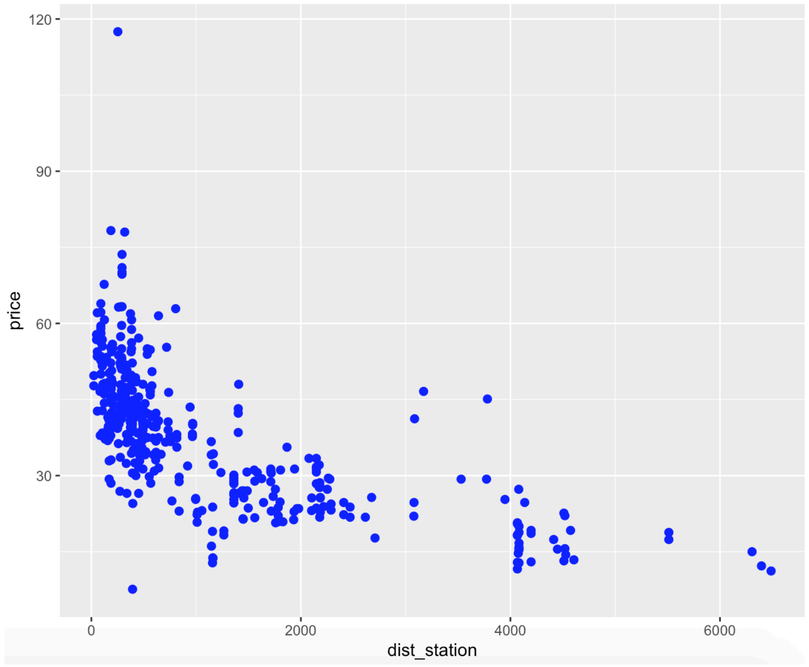
	No	transact_date	house_age	dist_station	number_stores	latitude	longitude	price
1	1	2012.917	32.0	84.87882	10	24.98298	121.5402	37.9
2	2	2012.917	19.5	306.59470	9	24.98034	121.5395	42.2
3	3	2013.583	13.3	561.98450	5	24.98746	121.5439	47.3
4	4	2013.500	13.3	561.98450	5	24.98746	121.5439	54.8
5	5	2012.833	5.0	390.56840	5	24.97937	121.5425	43.1
6	6	2012.667	7.1	2175.03000	3	24.96305	121.5125	32.1

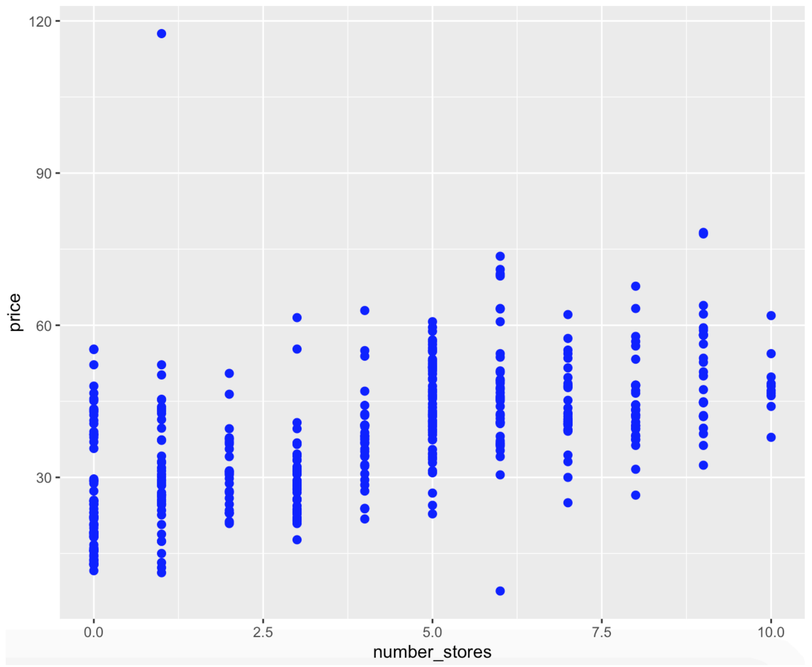
Real Estate Example

$X_{.1}$ $X_{.2}$ $X_{.3}$

- Assume that Y = real estate price, X = (house age, distance to station, number of stores)
- We would like to estimate f such that $Y = f(X) + \varepsilon$







Prediction versus Inference

- Assume that the estimation of f is $\hat{f}$
- Obtaining $\hat{f}$ is good for two purposes:
    - **Prediction:** Given house age, distance to station, and number of stores, we predict the real estate price using $\hat{Y} = \hat{f}(X)$ ($\hat{f}$ can be treated as a black-box)
    - **Inference:** Understand how the real estate price changes as a function of house age, distance to station, and number of stores (require explicit form of $\hat{f}$)
        - Which predictor is most important?
        - Which predictor generates the biggest price?

Estimate f : Parametric Methods

- We assume parametric form of f:
    - Linear form: $f(X) = aX + b$
    - Quadratic form: $f(X) = aX^2 + bX + c$
    - Polynomial form: $f(X) = a_1X^m + a_2X^{m-1} + \ldots + a_mX + a_{m+1}$

How to Estimate f ?

- The training data looks like this

$$(Y_1, X_1), \ldots, (Y_n, X_n)$$

- **Step 1**: We choose the form of f, e.g., $f(X) = aX + b$

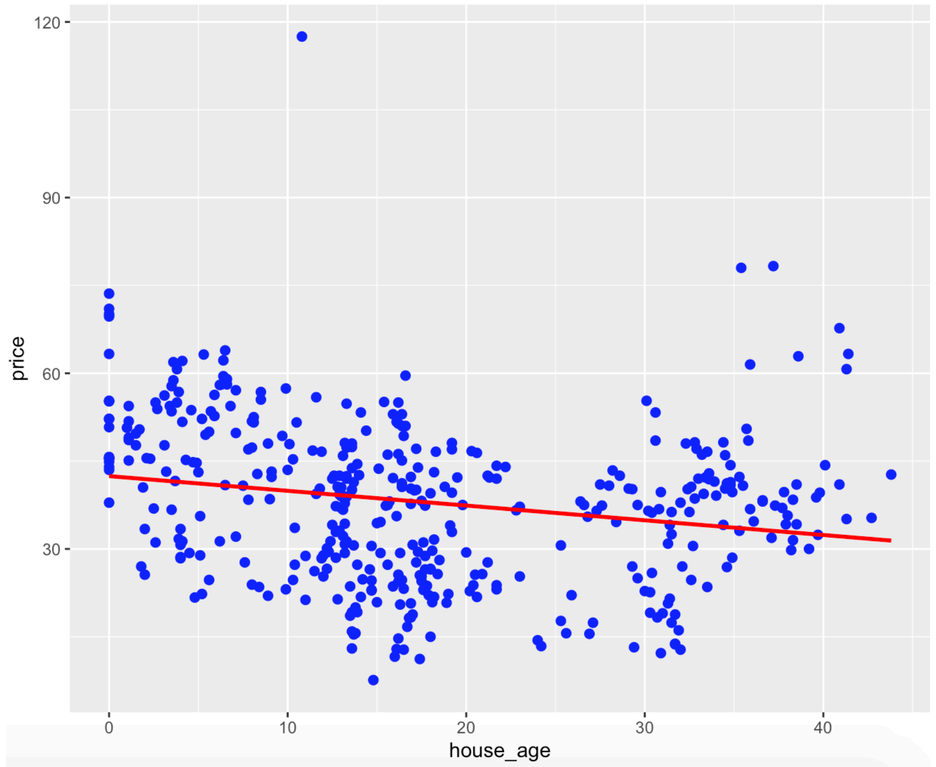
- **Step 2**: We select a loss function that measures the difference between $f(X)$ and Y. An example is squared loss (a.k.a. least squares):

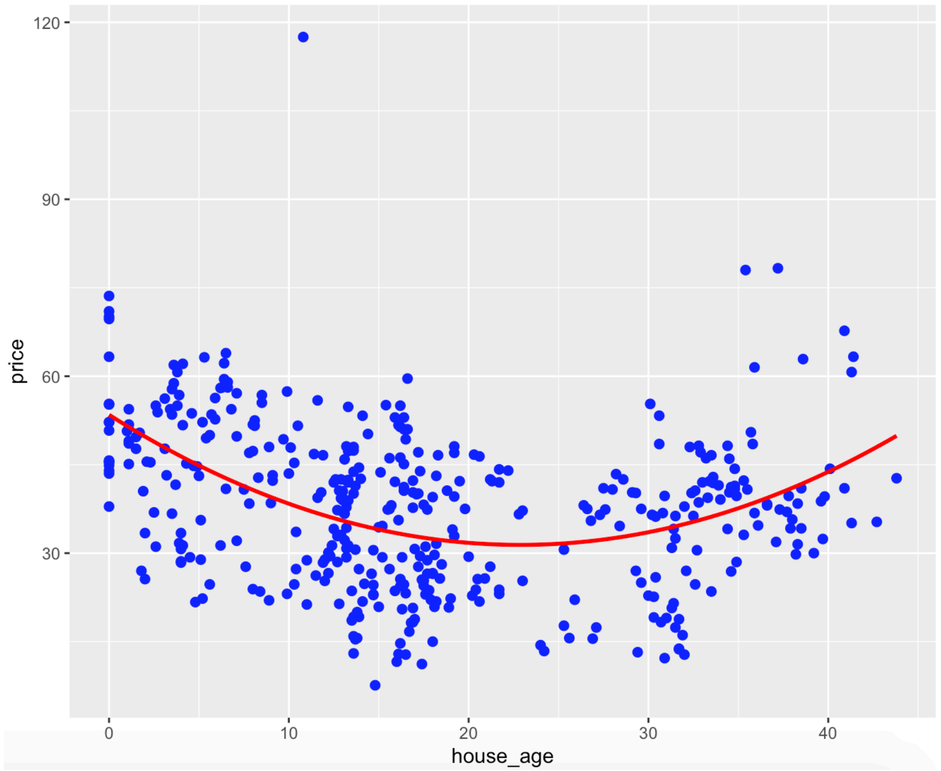
$$L(f) = \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$

- **Step 3**: We choose the parameters to minimize $L(f)$

Estimate f: Parametric Methods

Y = real estate price, X = house age





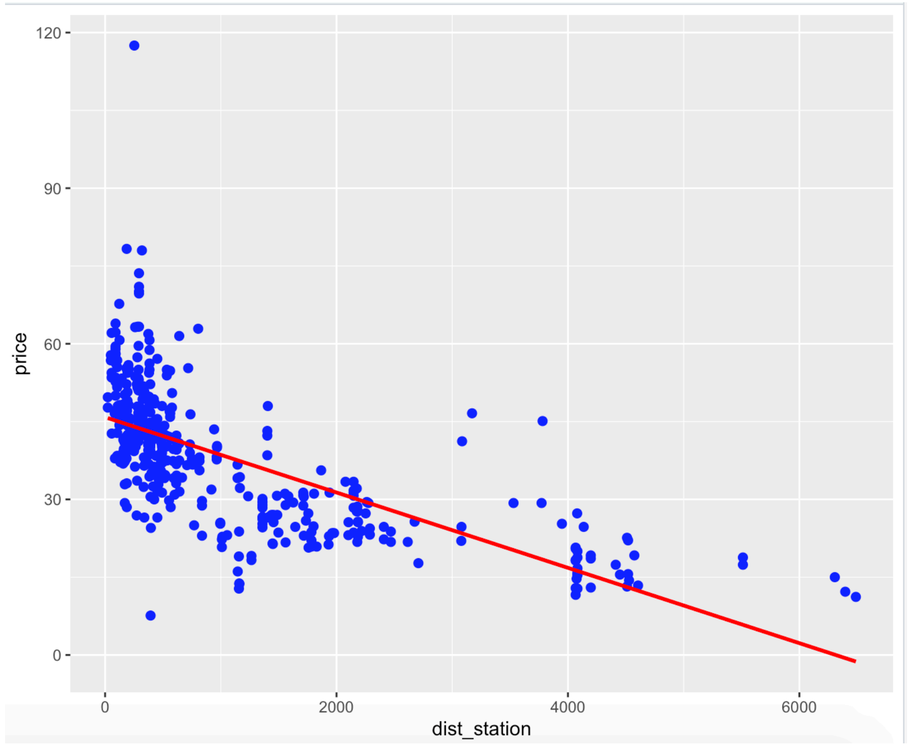
$$f(X) = aX + b$$

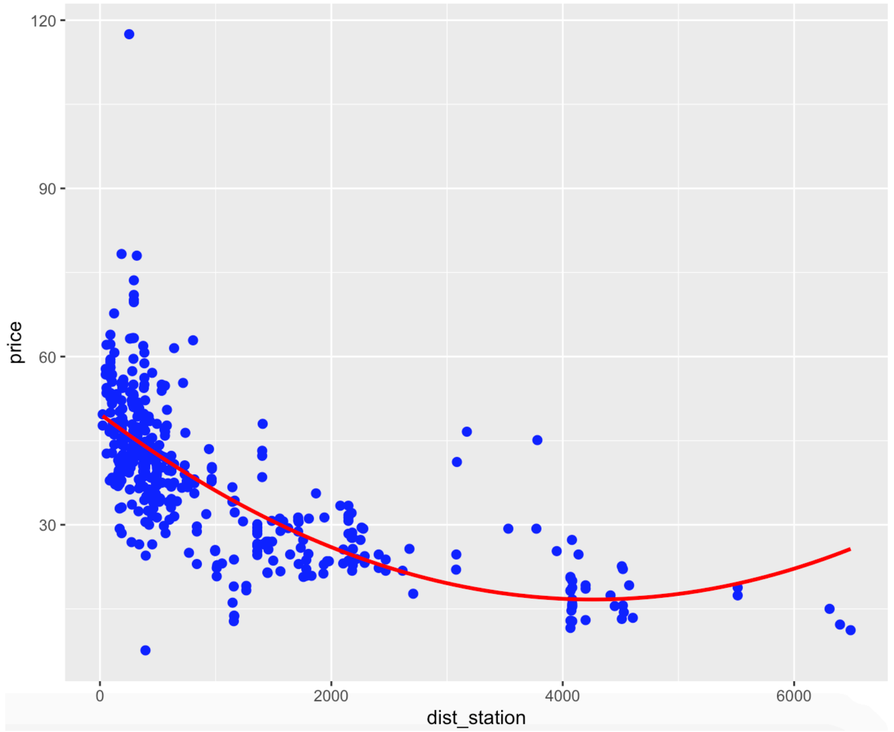
$$f(X) = aX^2 + bX + c$$

Question: Which form will we choose? Why?

Estimate f : Parametric Methods

Y = real estate price, X = distance to stations





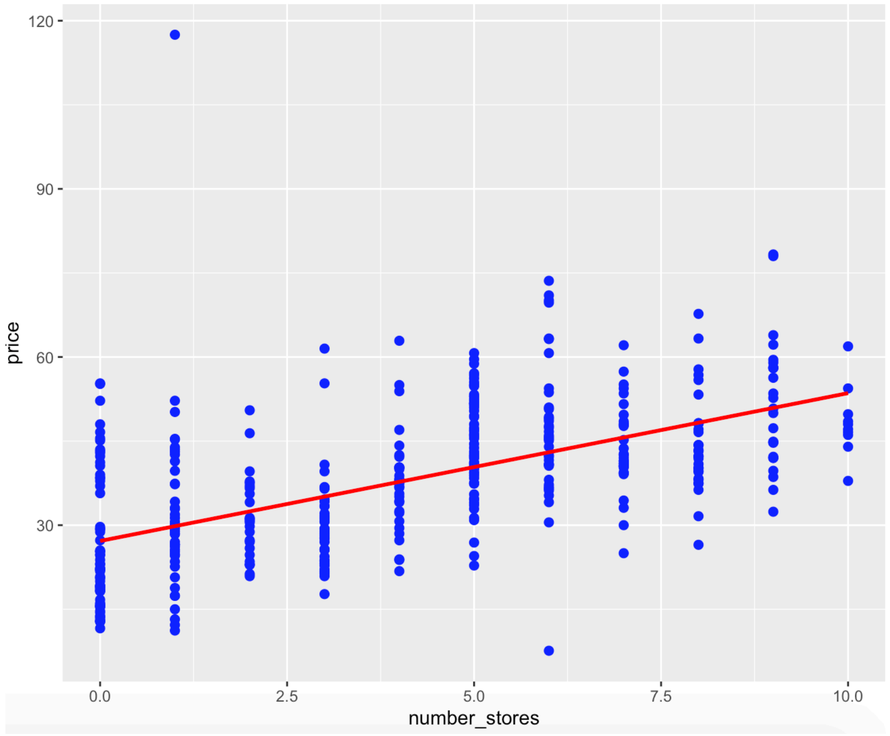
$$f(X) = aX + b$$

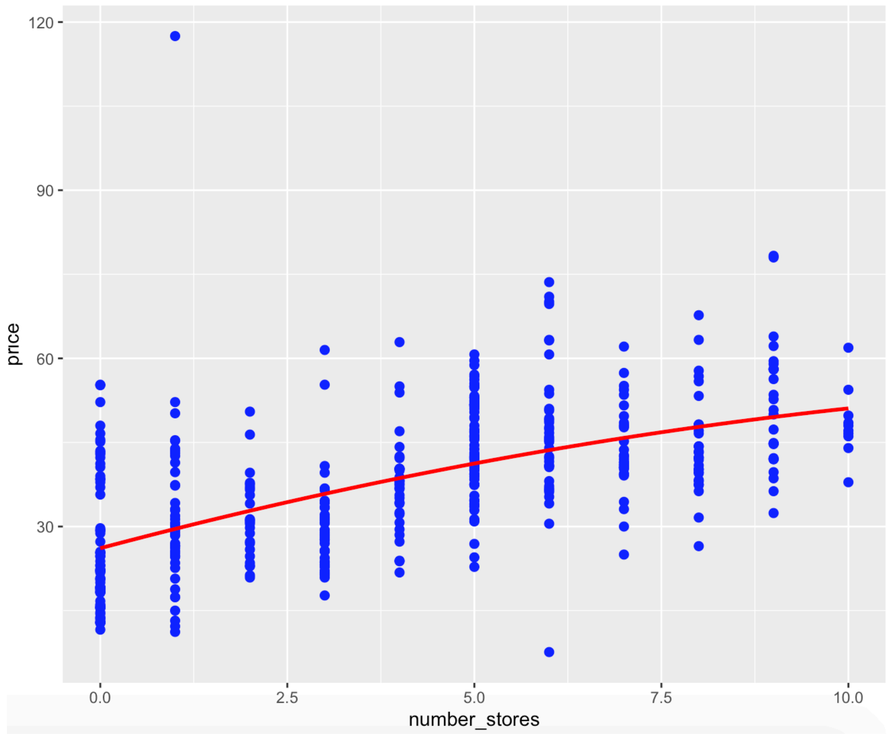
$$f(X) = aX^2 + bX + c$$

Question: Which form will we choose? Why?

Estimate f : Parametric Methods

Y = real estate price, X = number of stores





$$f(X) = aX + b$$

Question: Which form will we choose? Why?

$$f(X) = aX^2 + bX + c$$

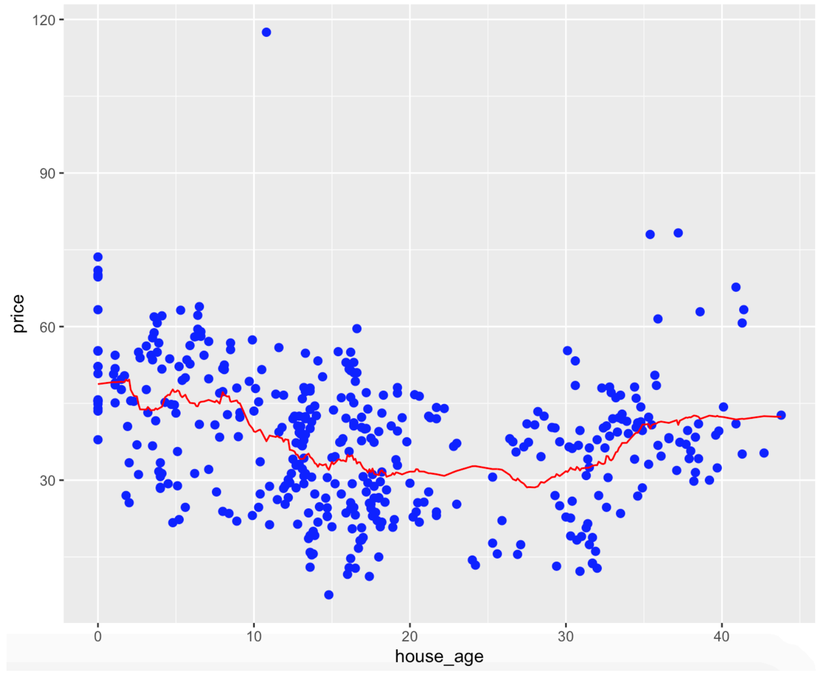
Estimate f : Nonparametric Methods

- We **do not** have any assumption on the form of f
- The model is much **richer yet harder** to obtain an estimate of f (There is no parameter to estimate like the parametric cases)
- Popular nonparametric methods:
    - K-nearest neighbors
    - Kernel regression

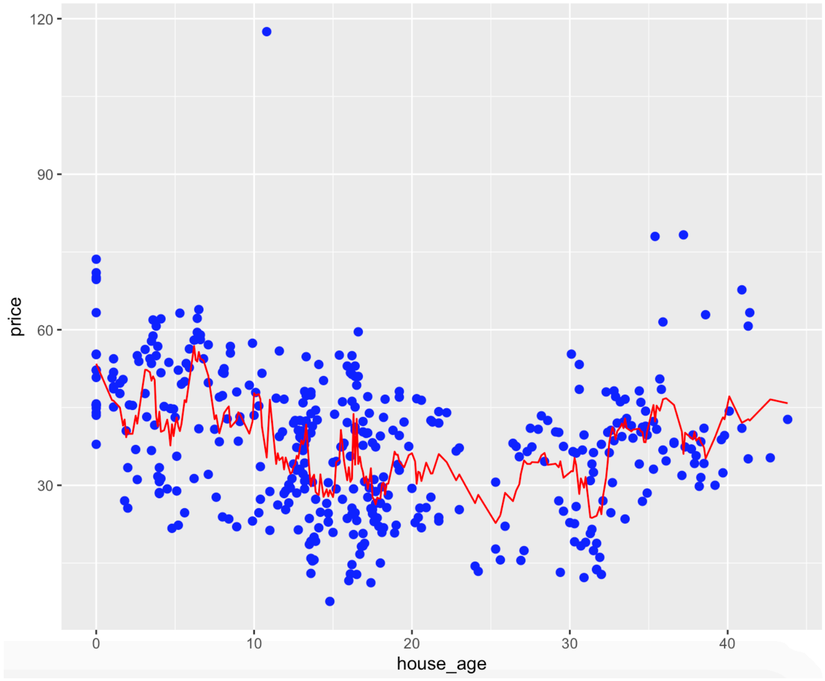
Nonparametric Method: K-nearest Neighbors

- The training data: $(Y_1, X_1), \ldots, (Y_n, X_n)$
- Assume that we have new data point X_* and we want to predict the value of Y_* at this new point
- **K-nearest neighbors**:
    - **Step 1**: We choose K points in the set $\{X_1, \ldots, X_n\}$ that are closest to X_*
    - **Step 2**: We average the values of Y_i at these points to obtain the Y_*

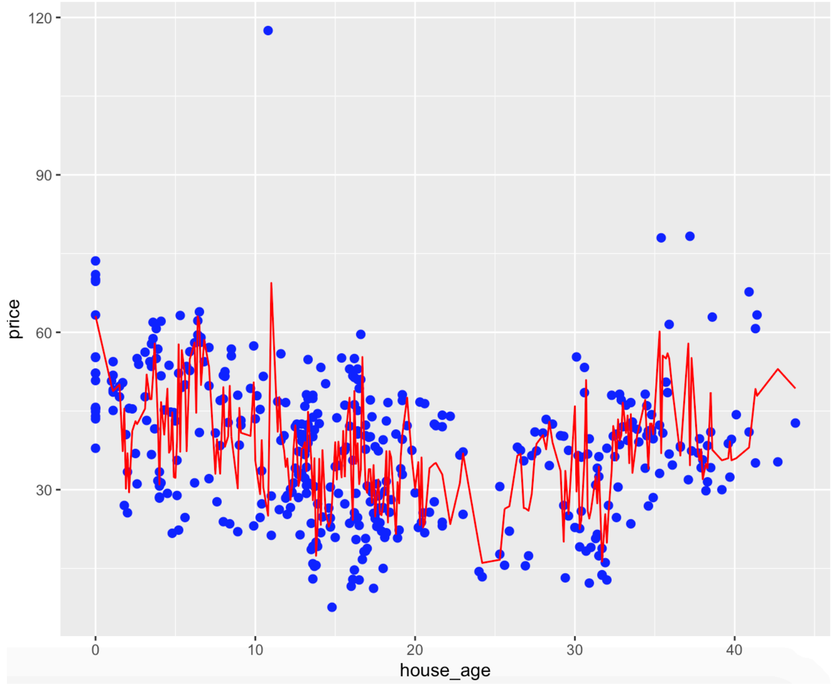
Nonparametric Method: K-nearest Neighbors



K = 50



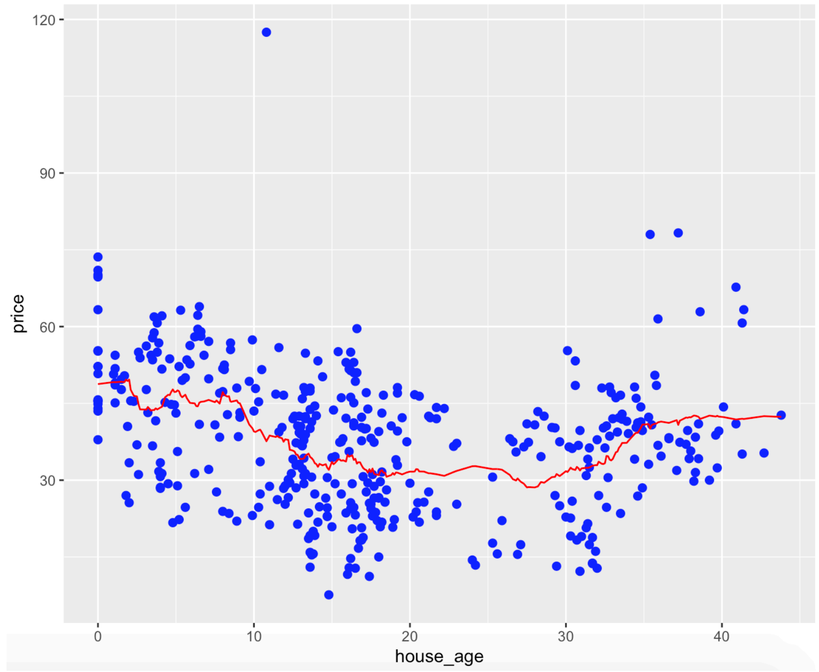
K = 10



K = 2

Question: Which value of K looks reasonable to us? Why?

Nonparametric Method: K-nearest Neighbors



K = 50

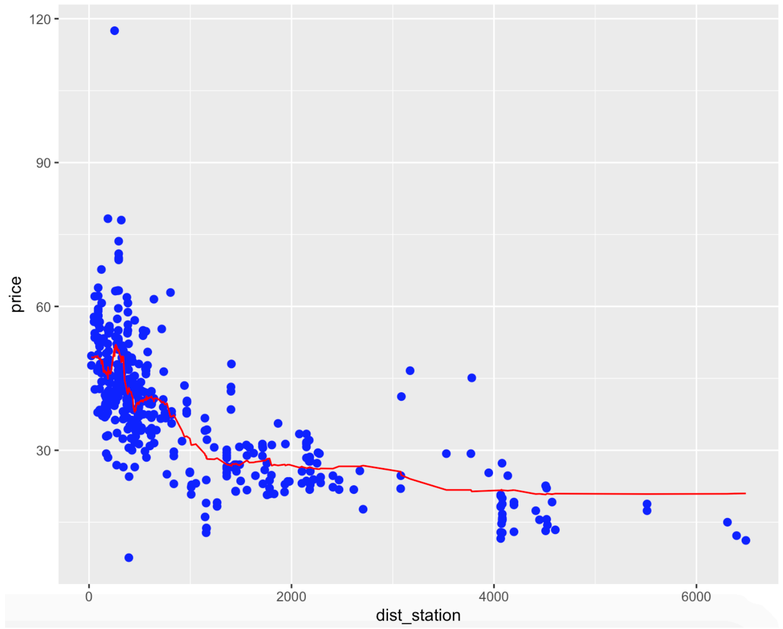
K = 10

K = 2

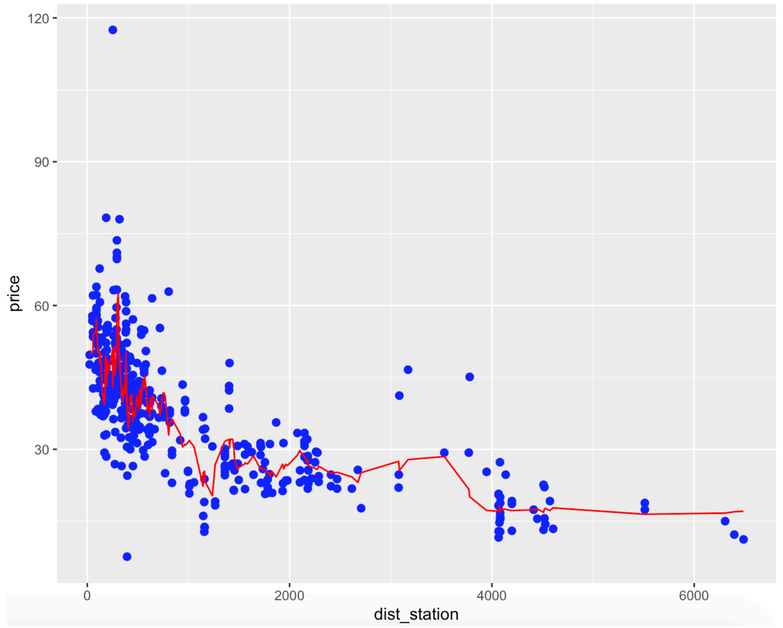
- Small K: complex model, very wiggly (small bias, high variance)
- Large K: smooth but simple model (large bias, small variance)

→ Bias-variance tradeoff

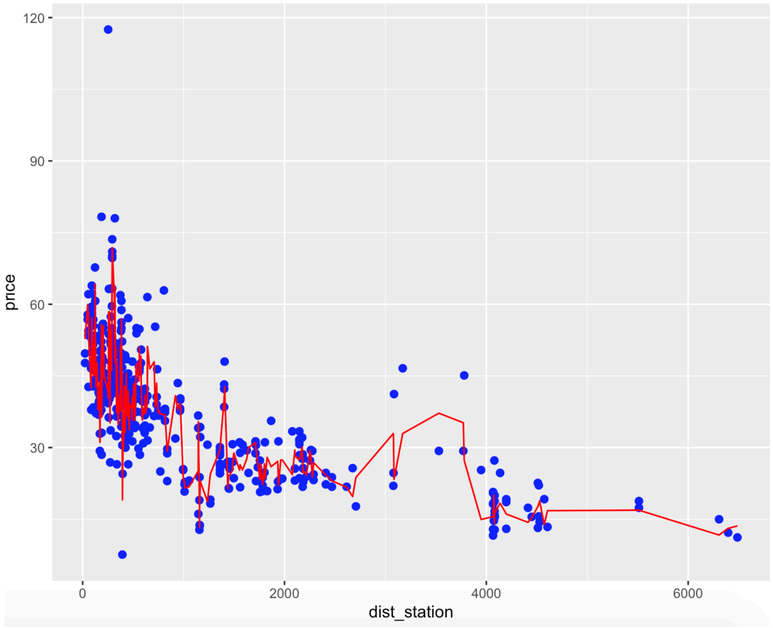
Nonparametric Methods: K-nearest Neighbors



K = 50

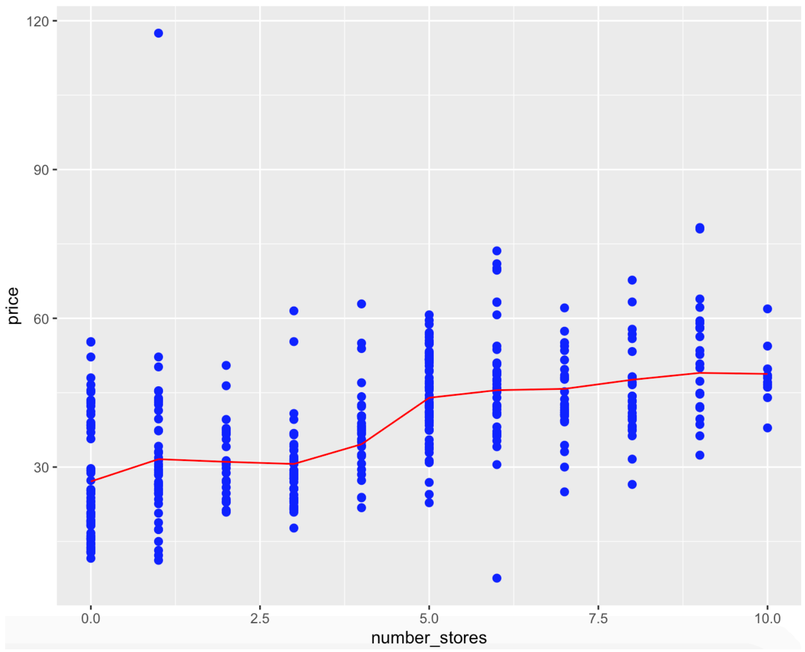


K = 10

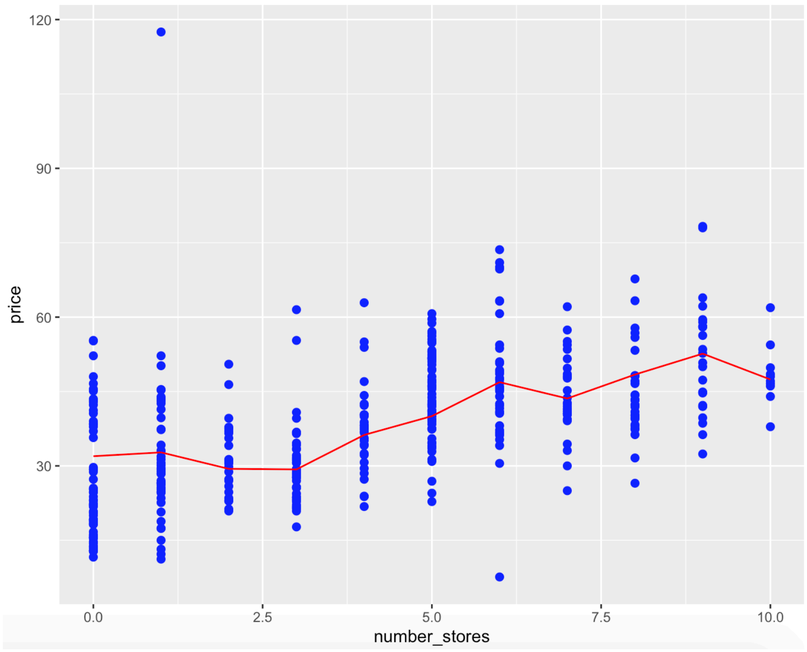


K = 2

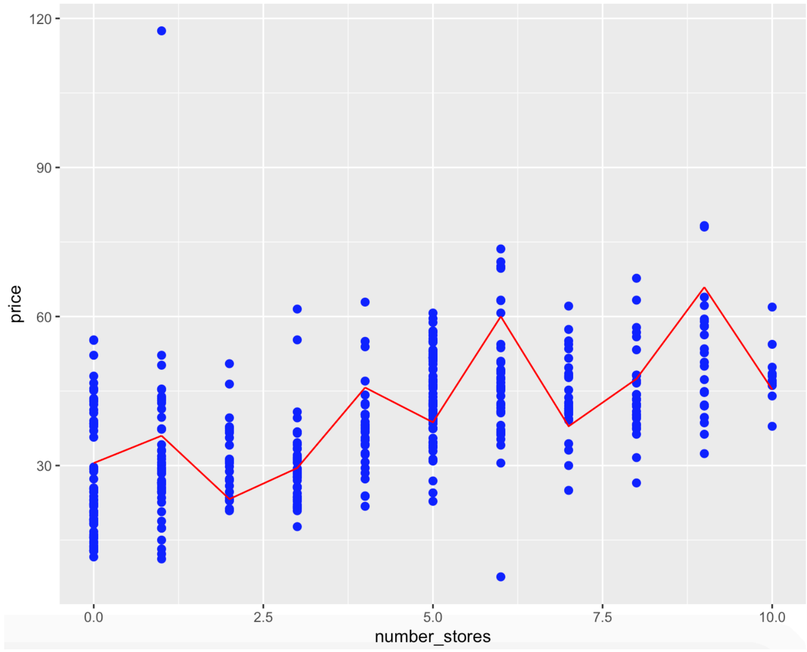
Nonparametric Methods: K-nearest Neighbors



K = 50



K = 10



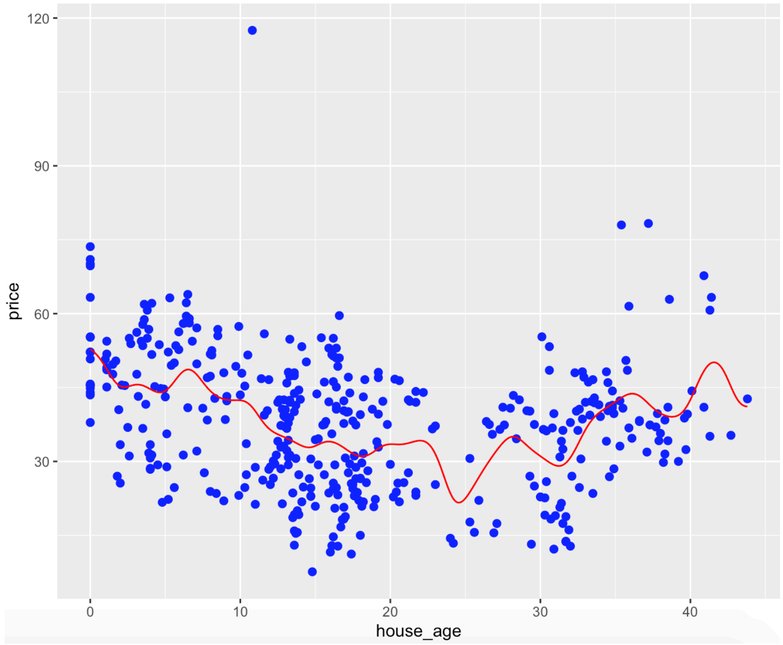
K = 2

Nonparametric Method: Kernel Regression

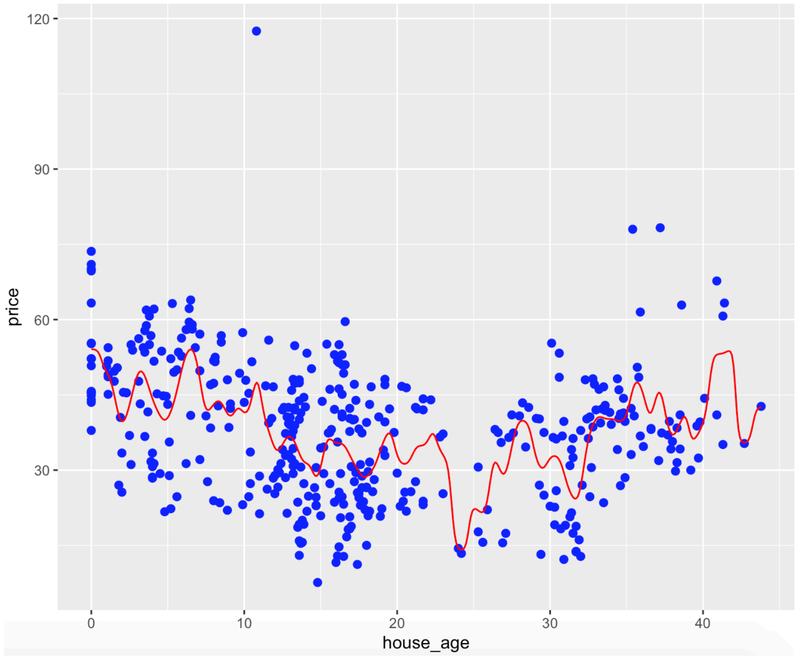
- K-nearest neighbors: equal weights for all Y_i (can be undesirable if some Y_i are more important than the others)

- **Kernel regression**: Assigning weights to Y_i based on certain distributions, such as Gaussian distribution

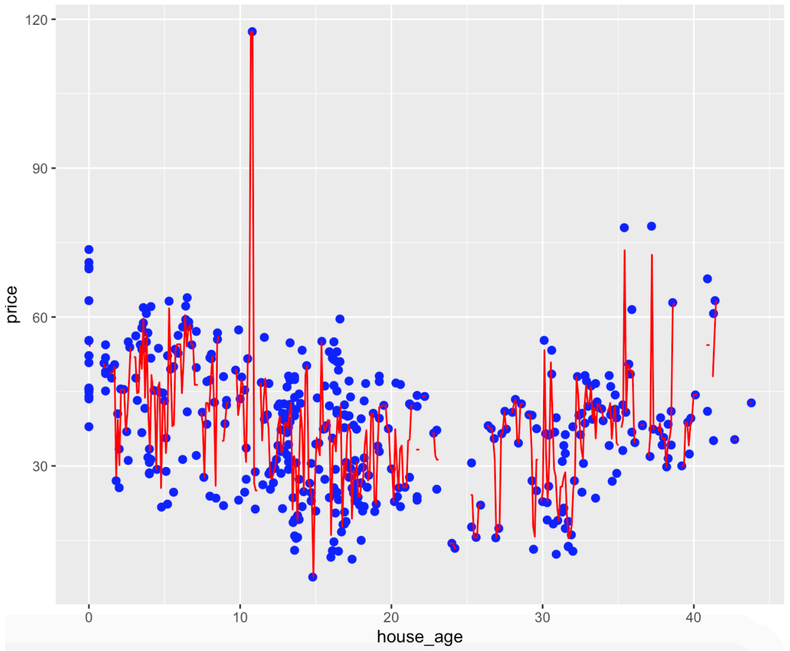
Nonparametric Method: Kernel Regression



bandwidth = 2.5

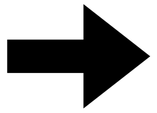


bandwidth = 1



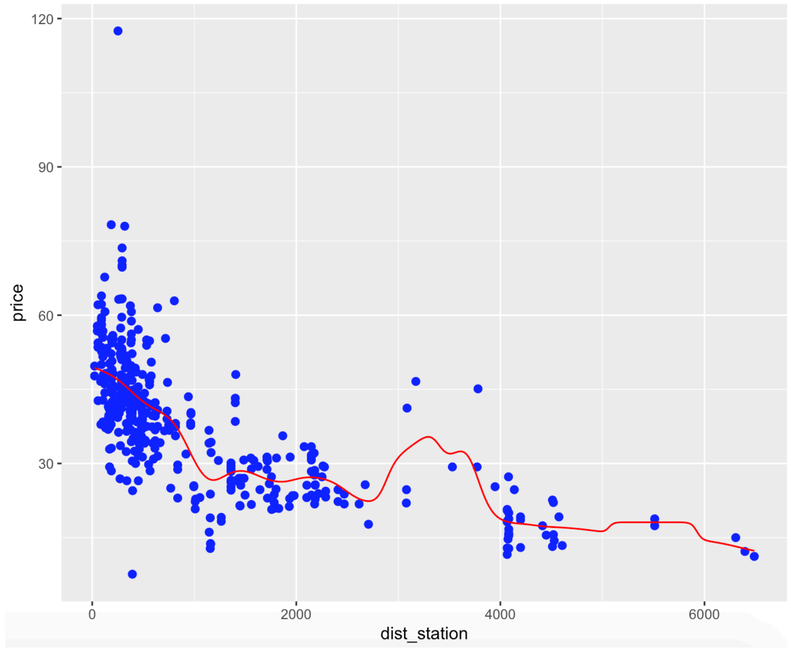
bandwidth = 0.1

- Small bandwidth leads to good bias but high variance
- Large bandwidth leads to poor bias but low variance

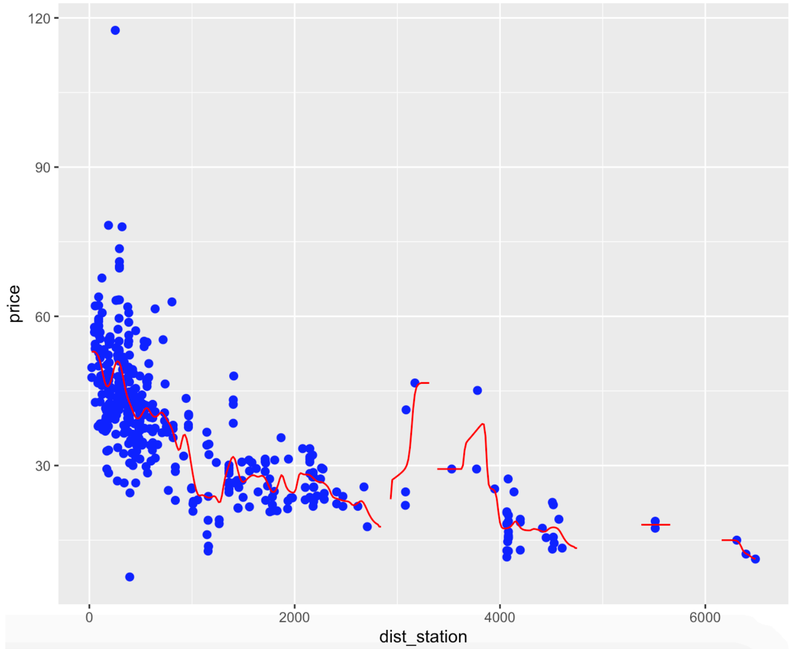


Question: What is “best” bandwidth?

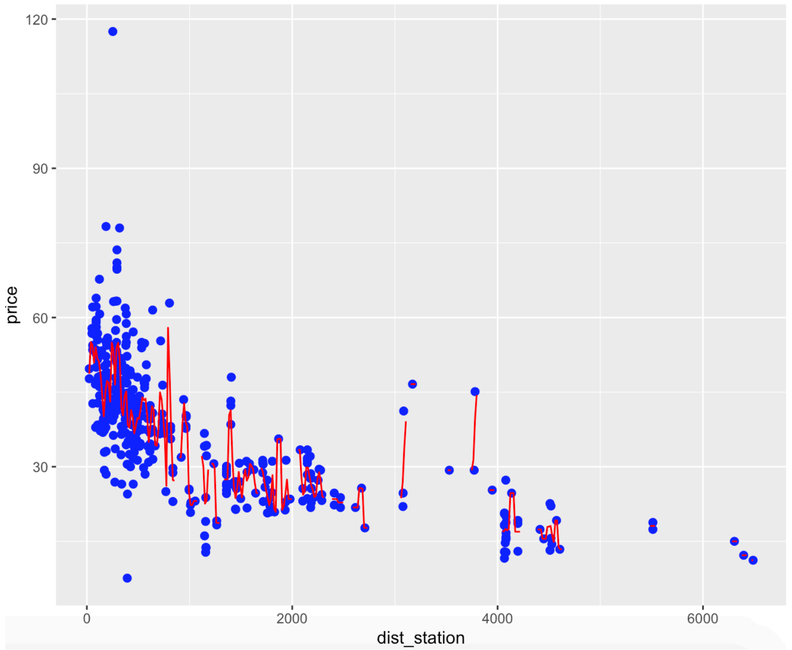
Nonparametric Method: Kernel Regression



bandwidth = 400



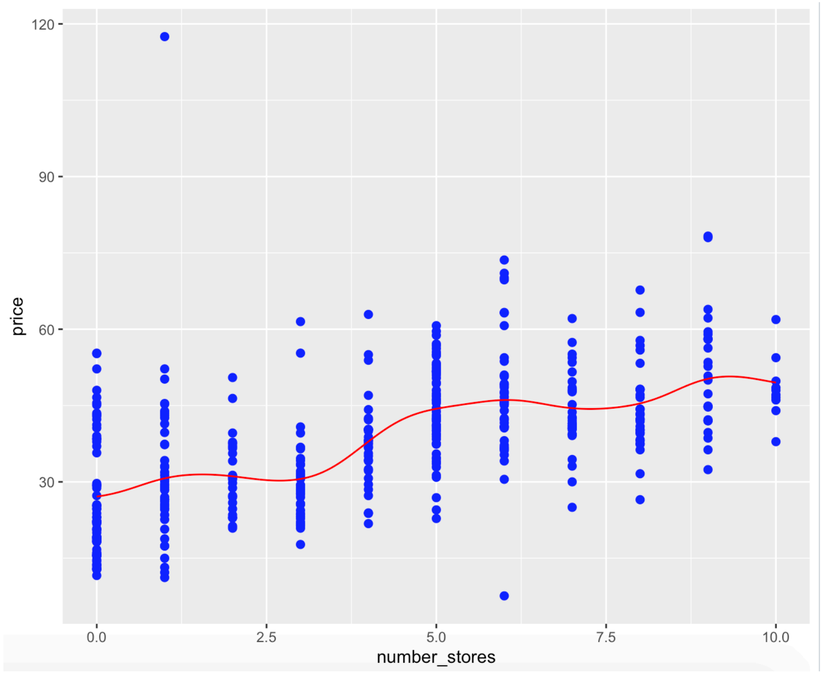
bandwidth = 100



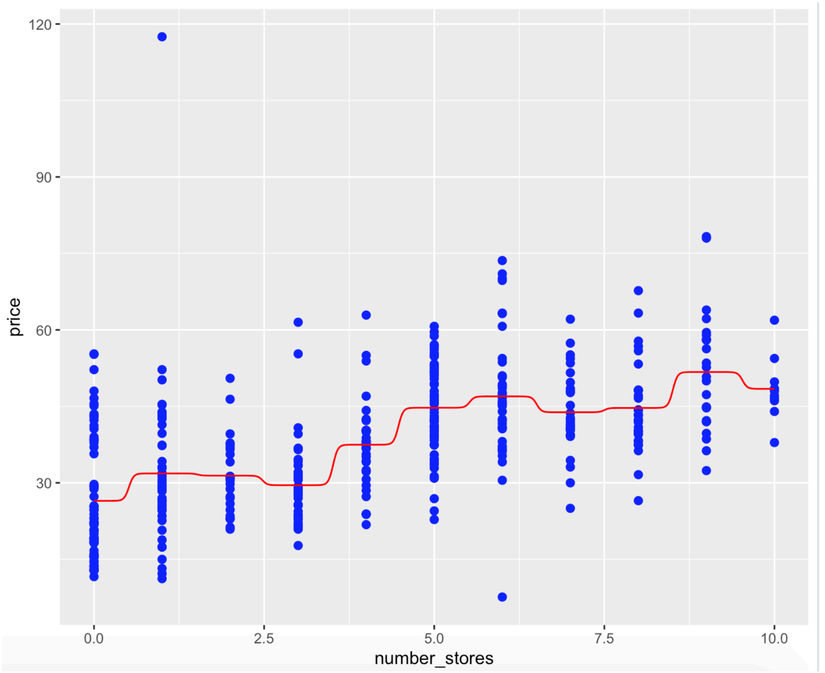
bandwidth = 20

Rule of Thumb: The bandwidth should be chosen based on the scale of data

Nonparametric Method: Kernel Regression



bandwidth = 1.5



bandwidth = 0.5

Assessing Model Accuracy

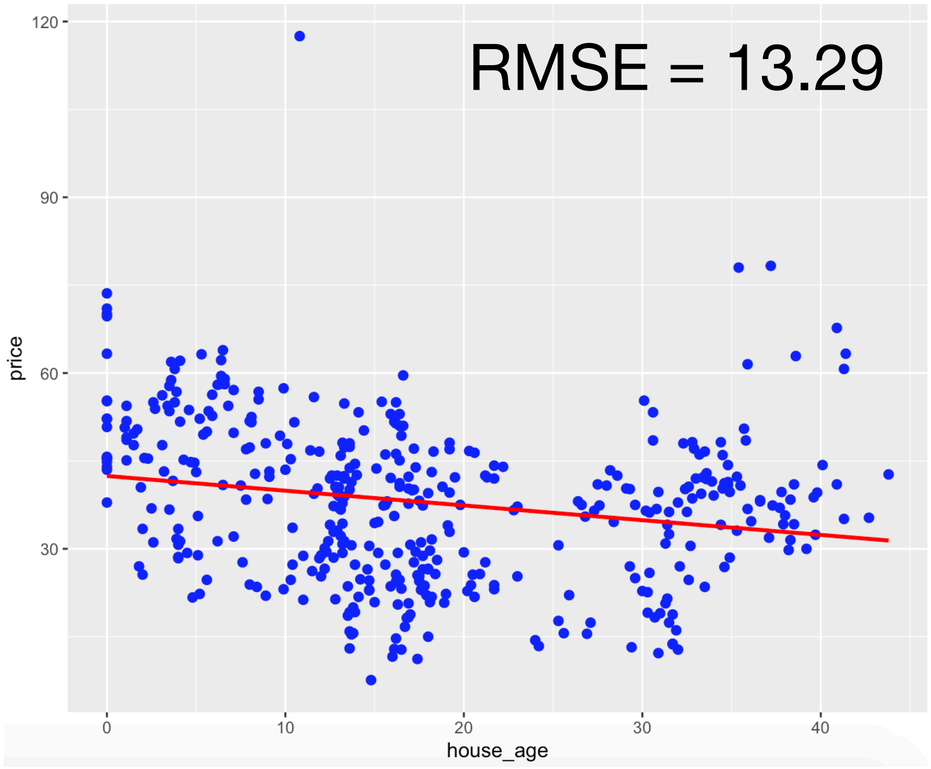
- In both kernel and K-nearest neighbor methods, how to choose good K or bandwidth?
- We first need to define a way to assess model accuracy
- **Mean square error (MSE)**:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}(X_i))^2$$

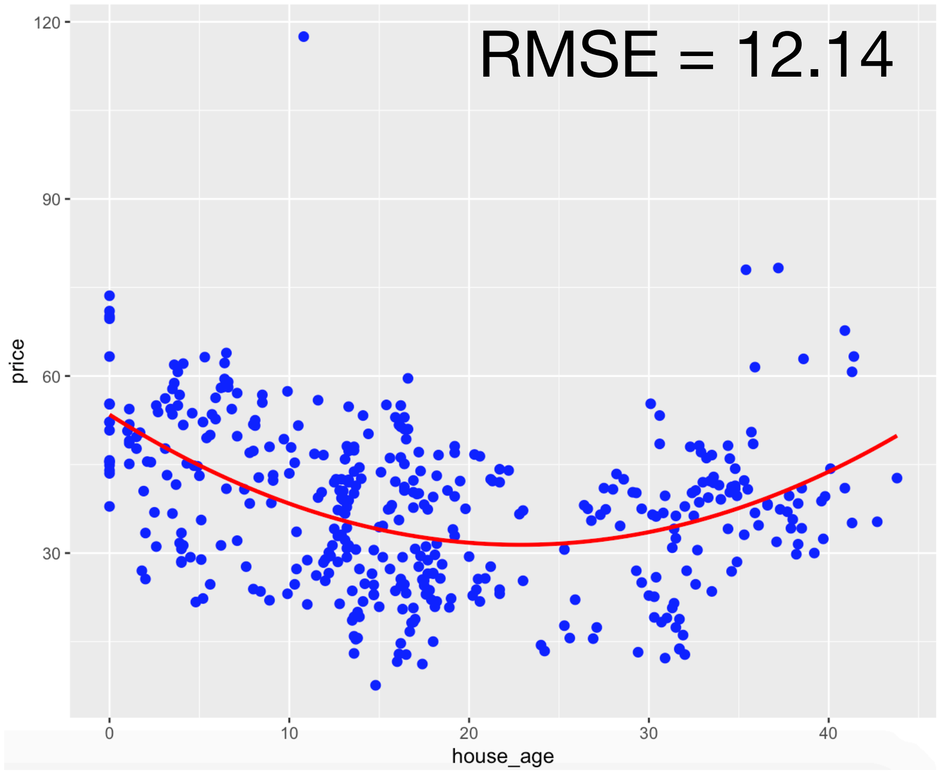
where $\hat{f}(X_i)$ is the prediction at data point X_i

- MSE is important to understand the trade-off between bias and variance
- In practice, we can use RMSE = $\sqrt{MSE}$ to evaluate the model accuracy

Assessing Model Accuracy: Parametric Method



$$f(X) = aX + b$$

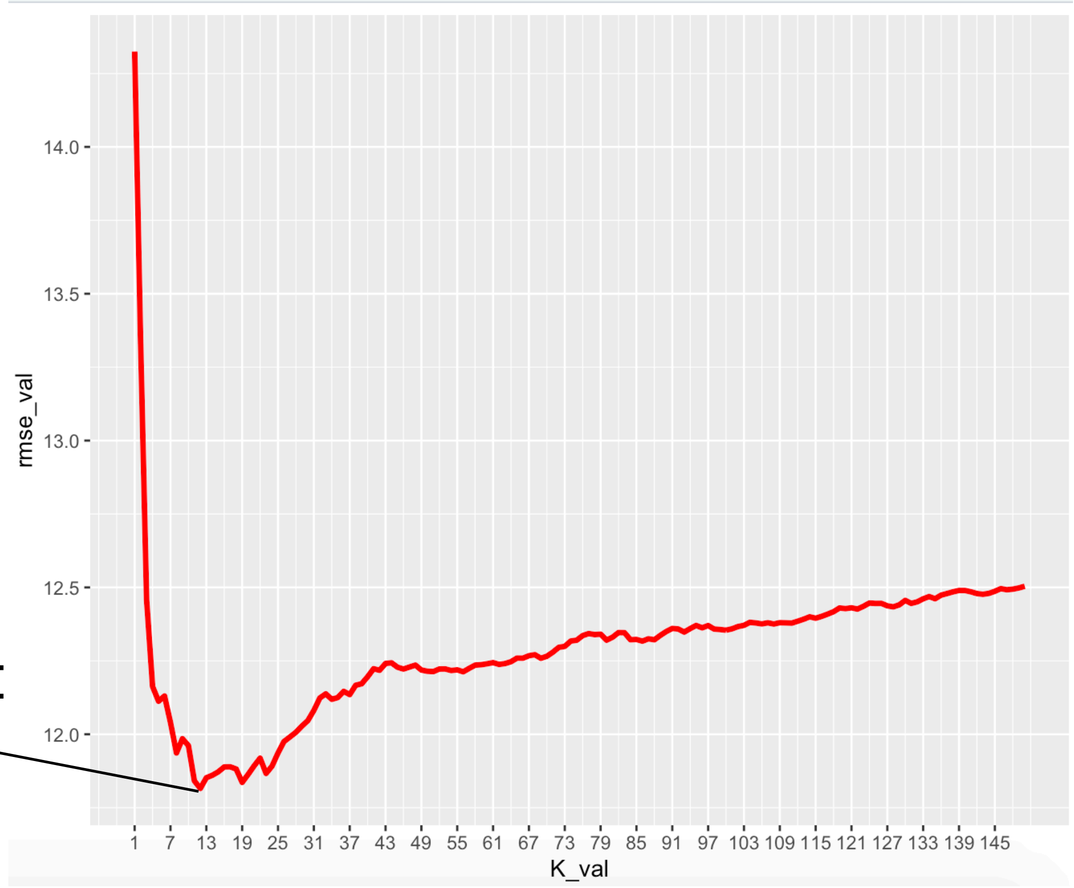


$$f(X) = aX^2 + bX + c$$

Question: Should we prefer quadratic model to linear model?

Assessing Model Accuracy: Nonparametric Method

- We use Y = real estate price, X = house age
- K-nearest neighbors method is used for estimating f



Smallest RMSE at K = 12

Should we choose K = 12 as the best value of K?

Train/ Test Set

- All the previous RMSE are calculated via all the training data
- The method that obtains best accuracy in training data does not necessarily imply good performance in the test/ new data
- The famous "**over-fitting**" phenomenon refers to when we have almost perfect performance of the method in training data but have poor performance on test/ new data

Can we guarantee that this model still performs well for new/ test data?

K-nearest neighbors with K = 12

Train/ Test Set

- Training data: $(Y_1, X_1), \ldots, (Y_n, X_n)$
- Test data: $(Y'_1, X'_1), \ldots, (Y'_m, X'_m)$
- **MSE for test data**:

$$MSE_{out} = \frac{1}{m} \sum_{i=1}^{m} (Y'_i - \hat{f}(X'_i))^2$$

where $\hat{f}$ is the estimate we obtain from training data

- $RMSE_{out} = \sqrt{MSE_{out}}$

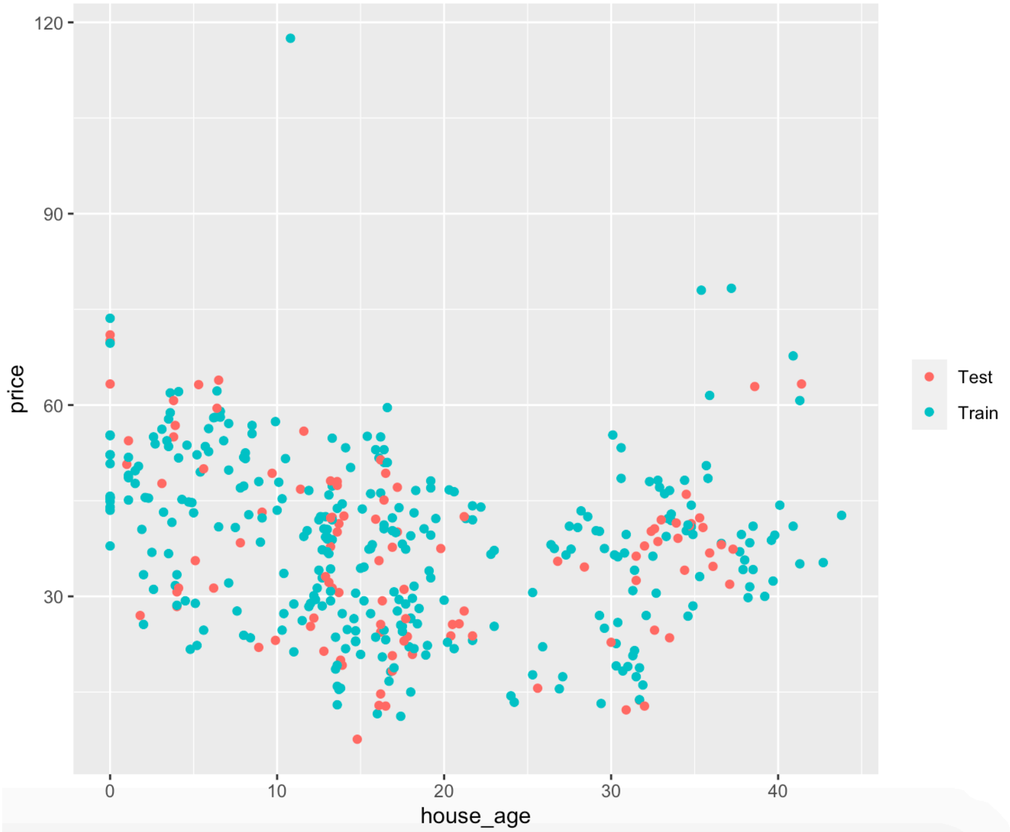
Train/ Test Set

- In practice, **test set is not available** and we only have training set
- A popular approach to create a test set is to split the whole training set into new training set and test set
- $(Y_1, X_1), \ldots, (Y_n, X_n)$: whole training set
- $(Y_1, X_1), \ldots, (Y_m, X_m)$: new training set (m < n)
- $(Y_{m+1}, X_{m+1}), \ldots, (Y_n, X_n)$: test set

- **Rule of Thumb**: We should not have overlapped data between training set and test set

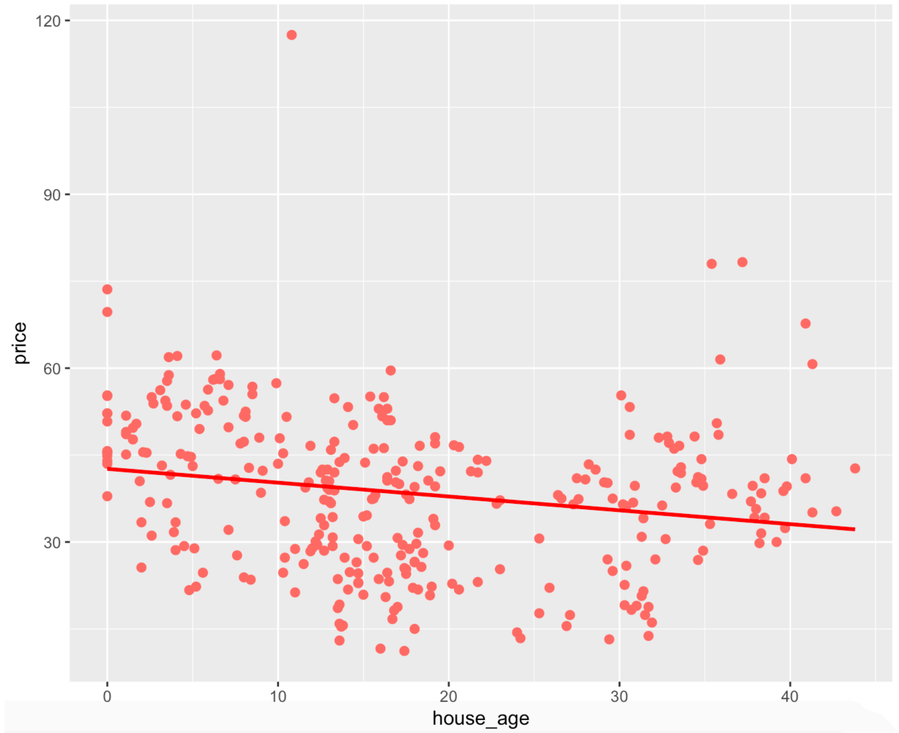
Train/ Test Set: Parametric Method

- We consider Y = real estate price, X = house age
- We split the whole training data into new training set and test set with the ratio 3:1 (75% training and 25% testing data)

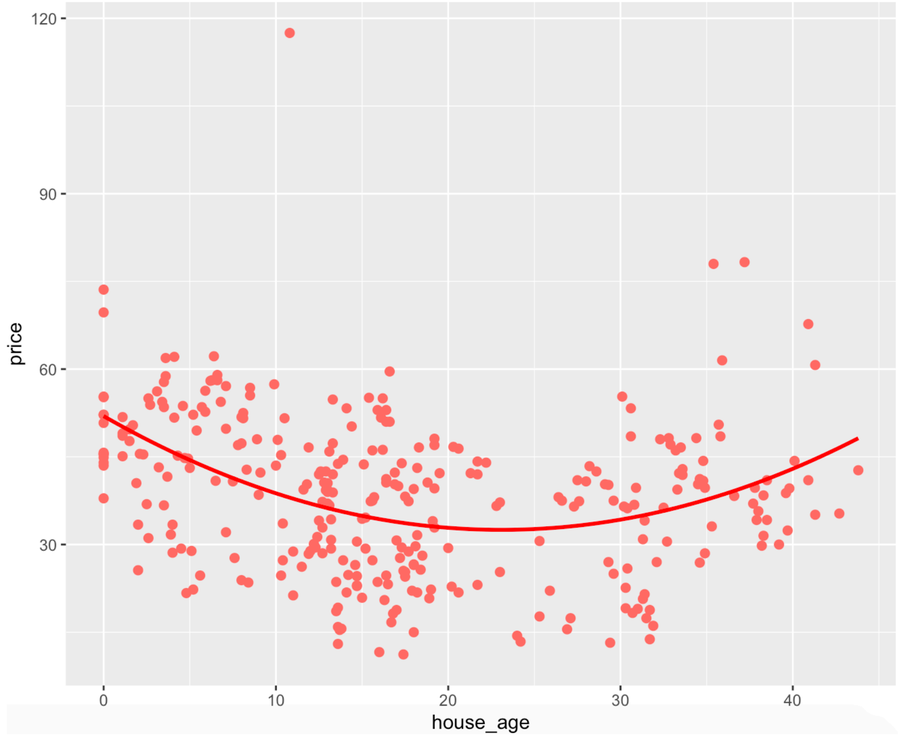


Train/ Test Set: Parametric Method

- First, we fit linear and quadratic models to training data



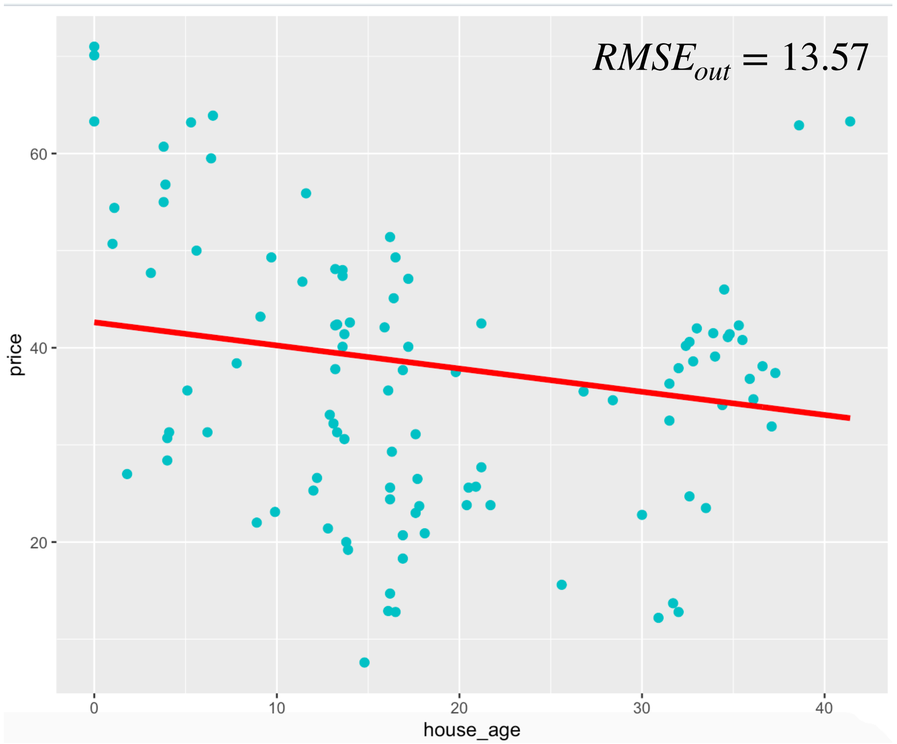
$$f(X) = aX + b$$



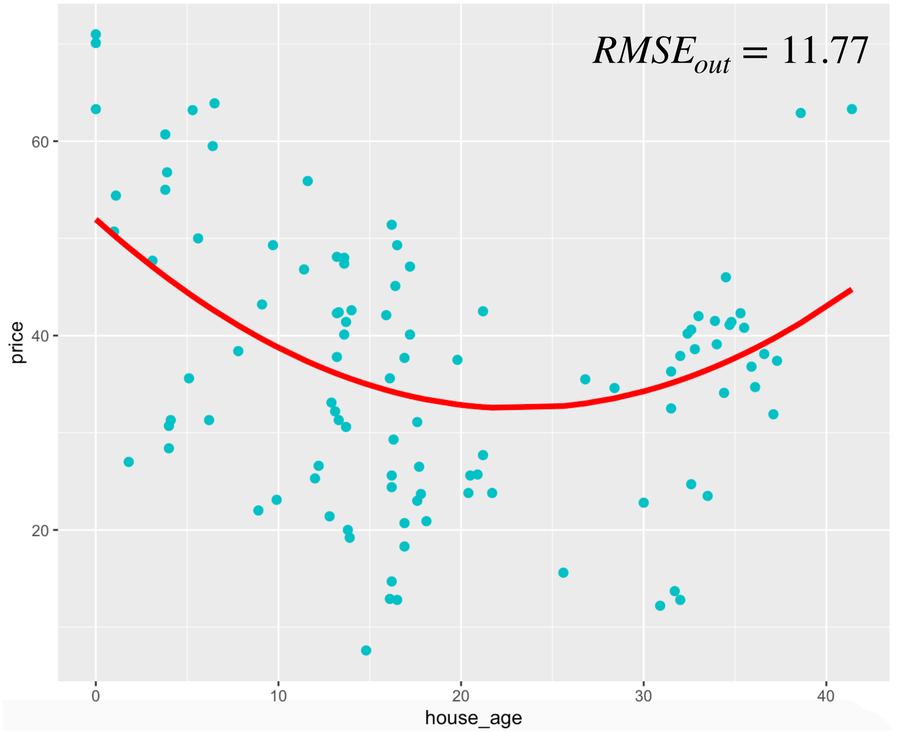
$$f(X) = aX^2 + bX + c$$

Train/ Test Set: Parametric Method

- Now, we use these fits to the test data



$$f(X) = aX + b$$



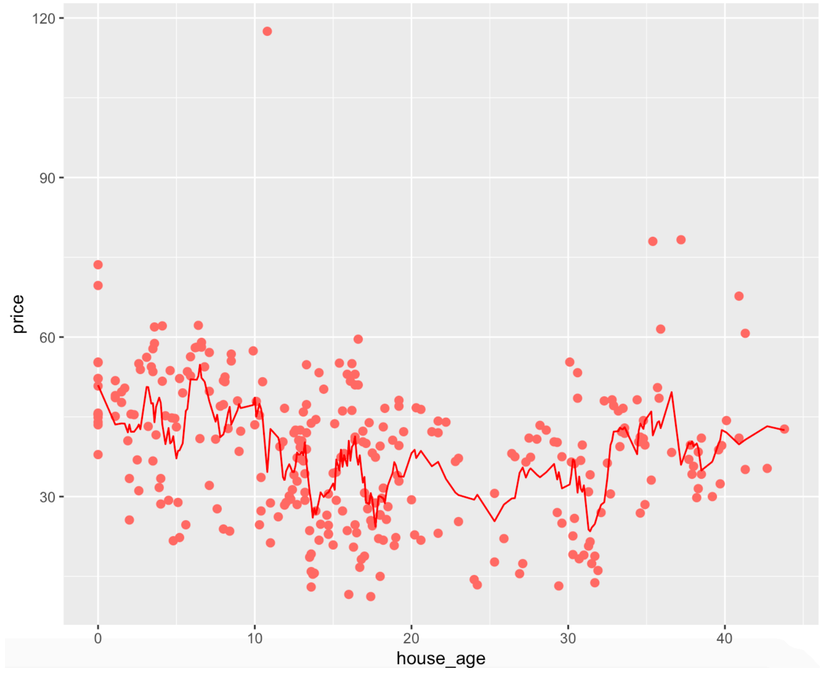
$$f(X) = aX^2 + bX + c$$

The quadratic model is preferred

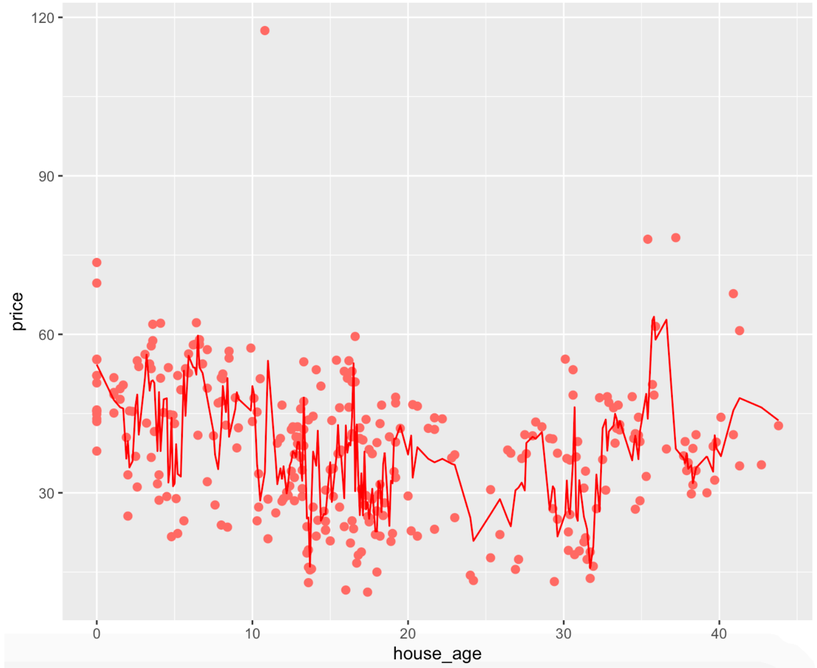
Train/ Test Set: Nonparametric Method

- We consider Y = real estate price, X = house age
- K-nearest neighbors method is used for estimating f in the training data

K = 50



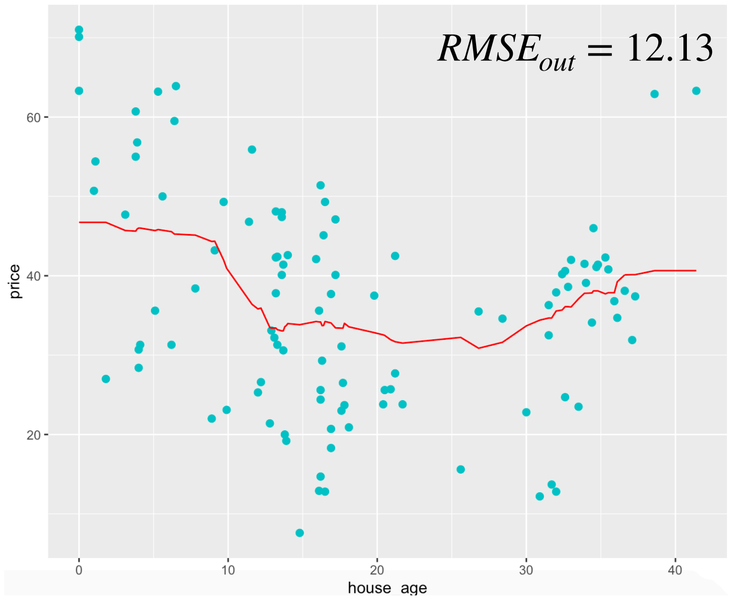
K = 10



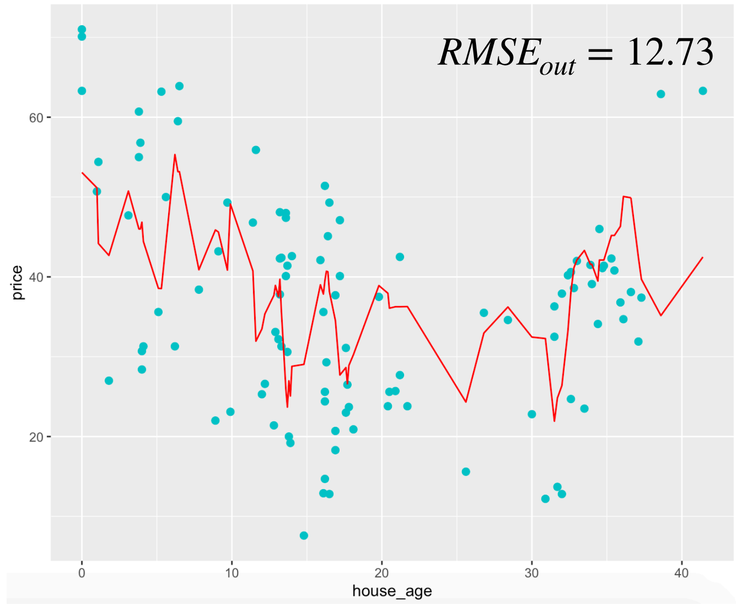
K = 3

Train/ Test Set: Nonparametric Method

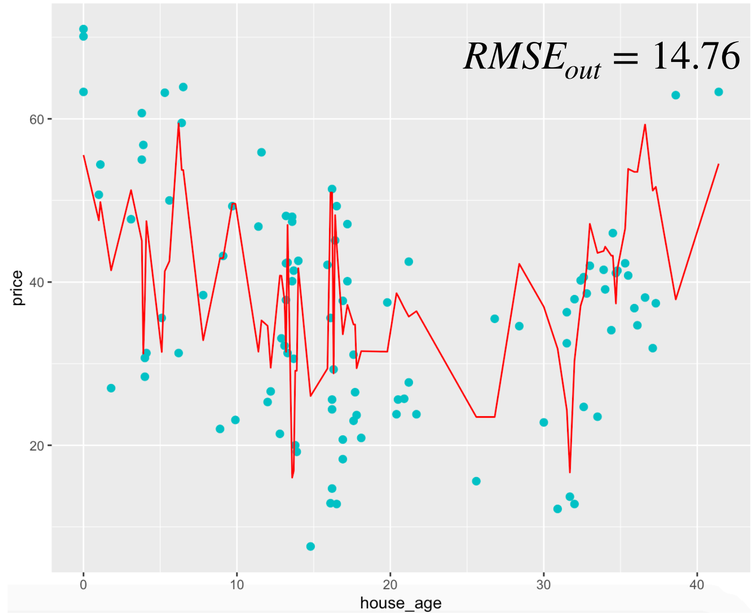
- Now, we use these K-nearest neighbors fits to the test data



K = 50



K = 10

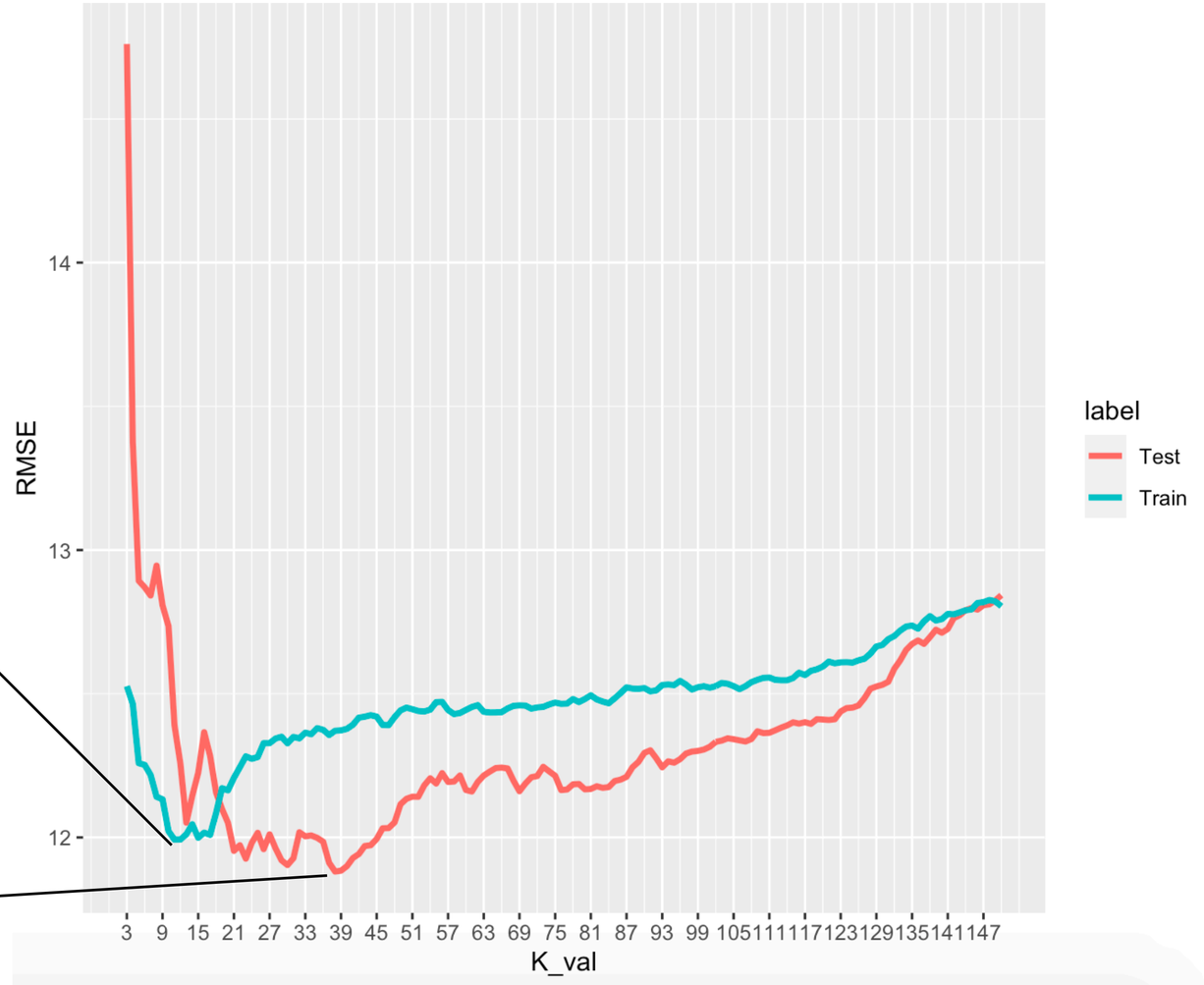


K = 3

Train/ Test Set: Nonparametric Method

Train data: best K = 12

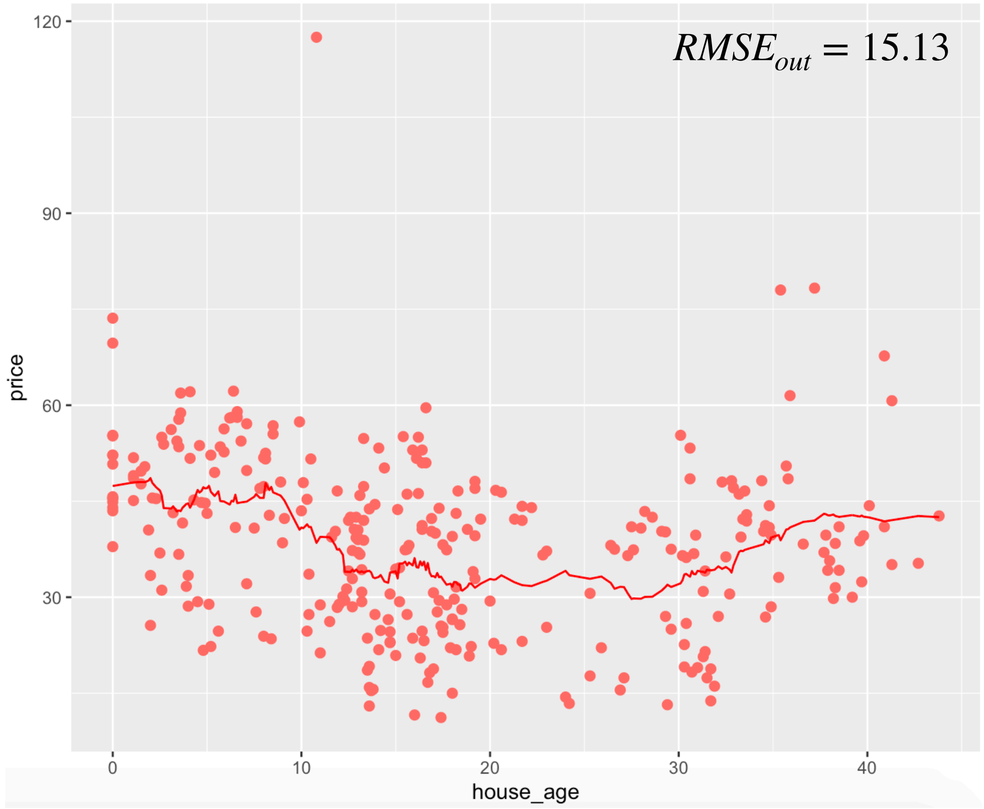
Test data: best K = 39



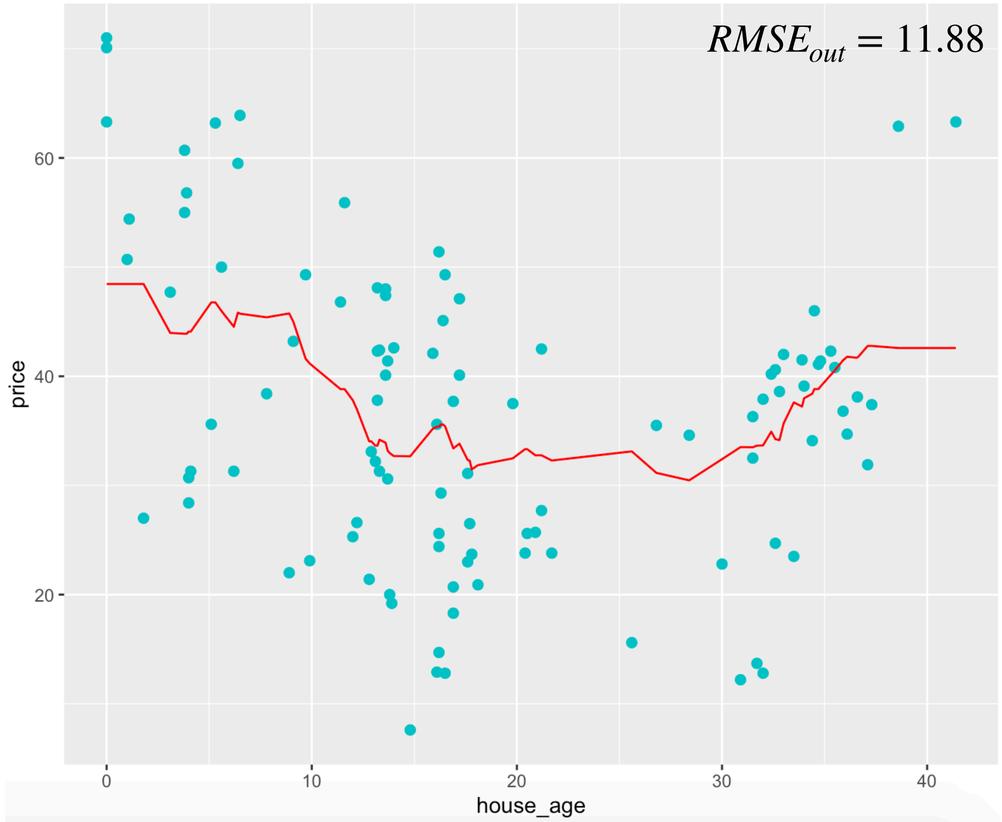
K = 39 balances the **simplicity and explanatory power**

Train/ Test Set: Nonparametric Method

K-nearest neighbors at “best” K = 39



Training set



Testing set

Train/ Test Set: Take-Home Messages

- Without test set, the RMSE from the training set is usually optimistic
- The RMSE from test set helps us from going too wrong
- The “best” model is the one that balances the simplicity and explanatory power

Train/ Test Set: Practice Examples

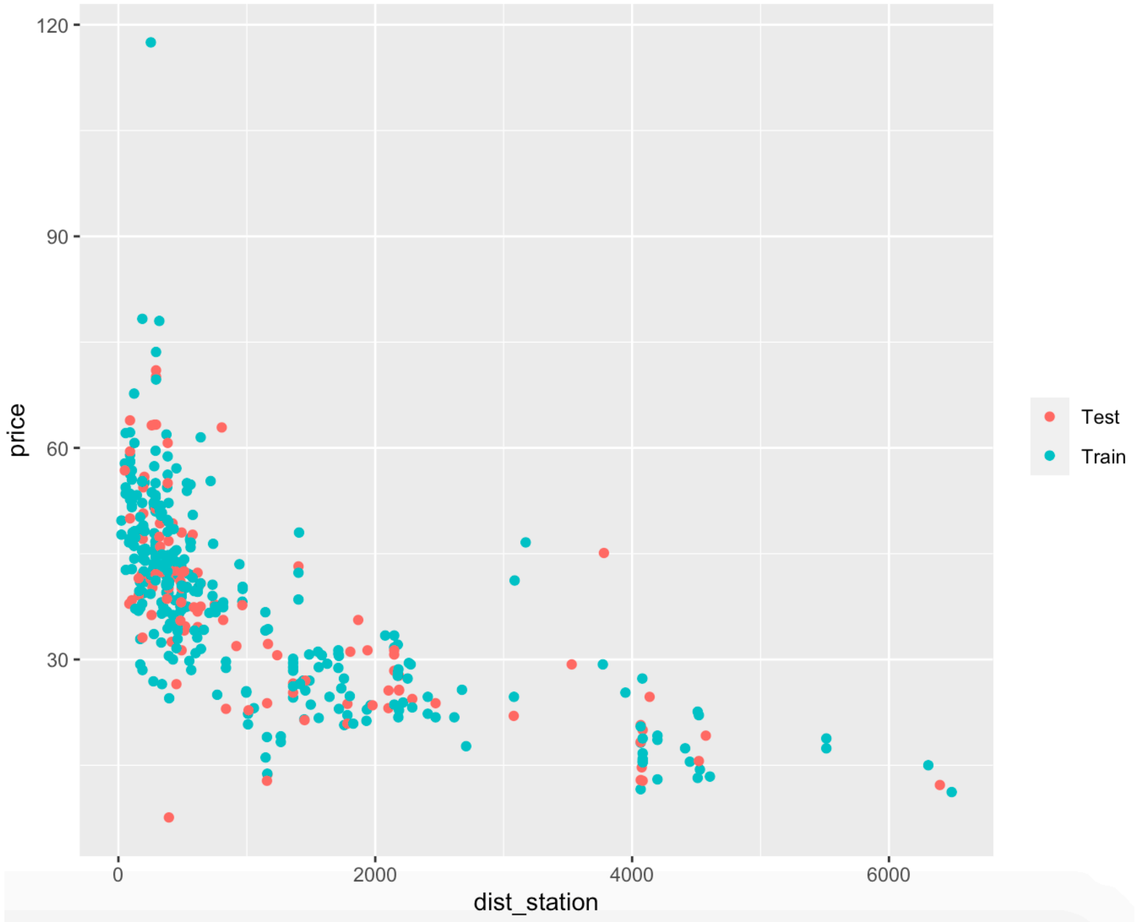
- Thus far, we have assessed model accuracy with Y = real estate price and X = house age
- Now, we would like to assess model accuracy when X = distance to station or X = number of stores
- Open the “Intro_Stats_Learning_Practice_Dist.R” and “Intro_Stats_Learning_Practice_Store.R” from the folder in this link https://drive.google.com/drive/u/0/folders/1Rd5nf2486caBpr48VcVi2aaDX_XtVrKH

Train/ Test Set: Price versus Distance

- Assume that Y = real estate price and X = distance to station
- **Questions**:
    - Between quadratic and linear models, which one we prefer? Should we use higher order polynomial models?
    - For K-nearest neighbors method, what is the best choice of K?

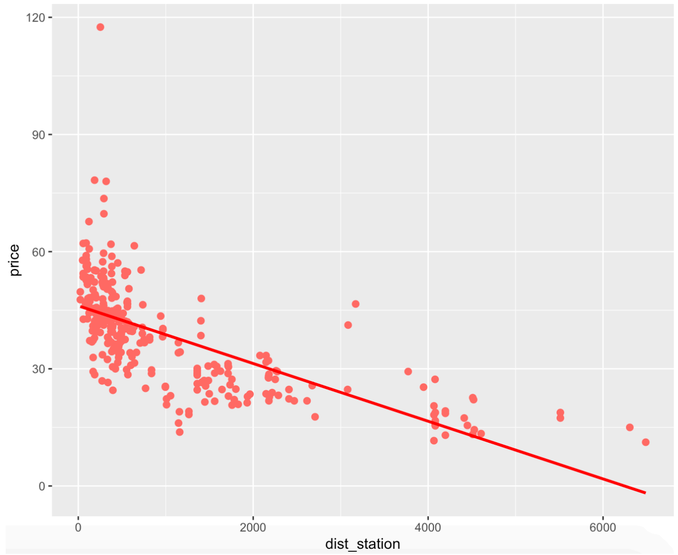
Train/ Test Set: Price versus Distance

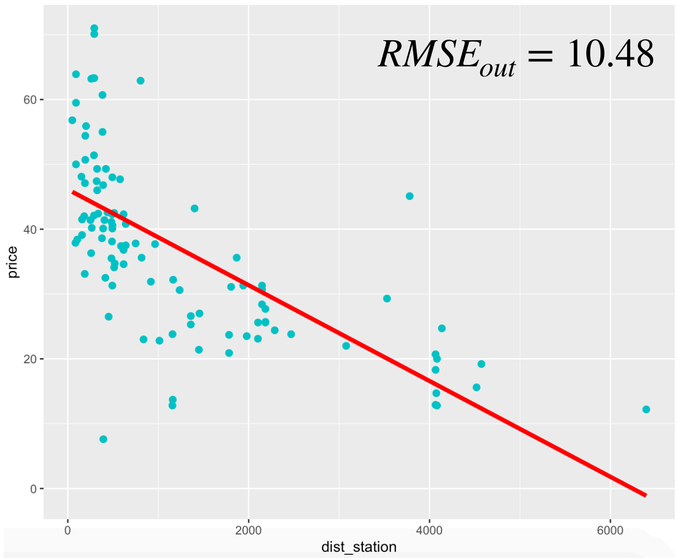
- Y = real estate price, X = distance to station



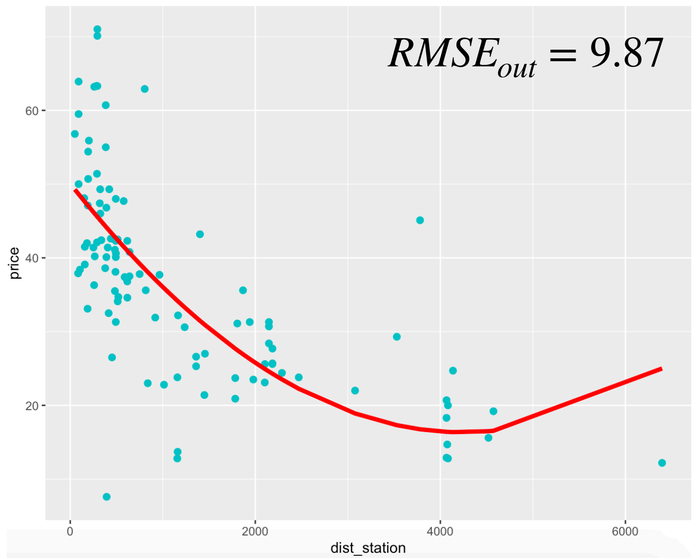
Train/ Test Set: Price versus Distance

We first consider parametric methods





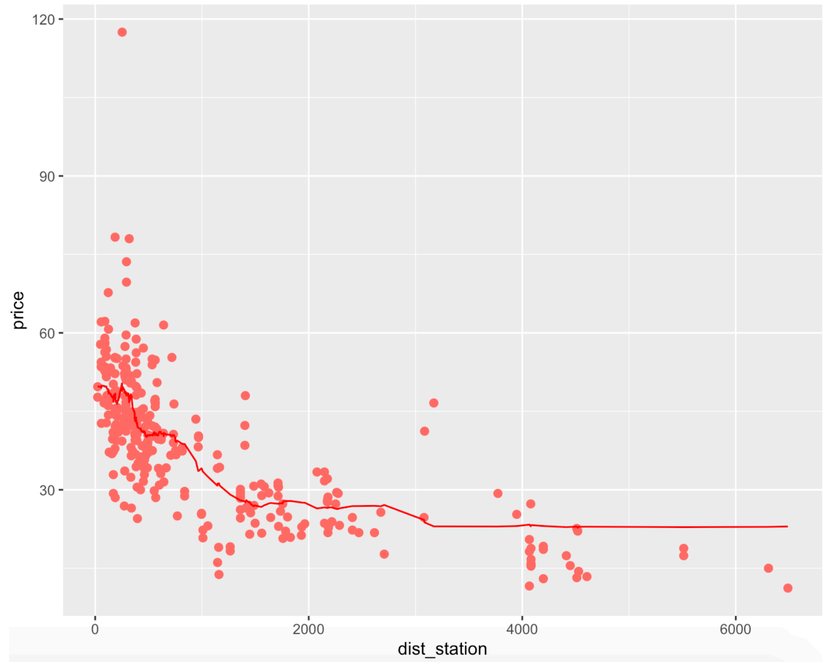
$$f(X) = aX + b$$



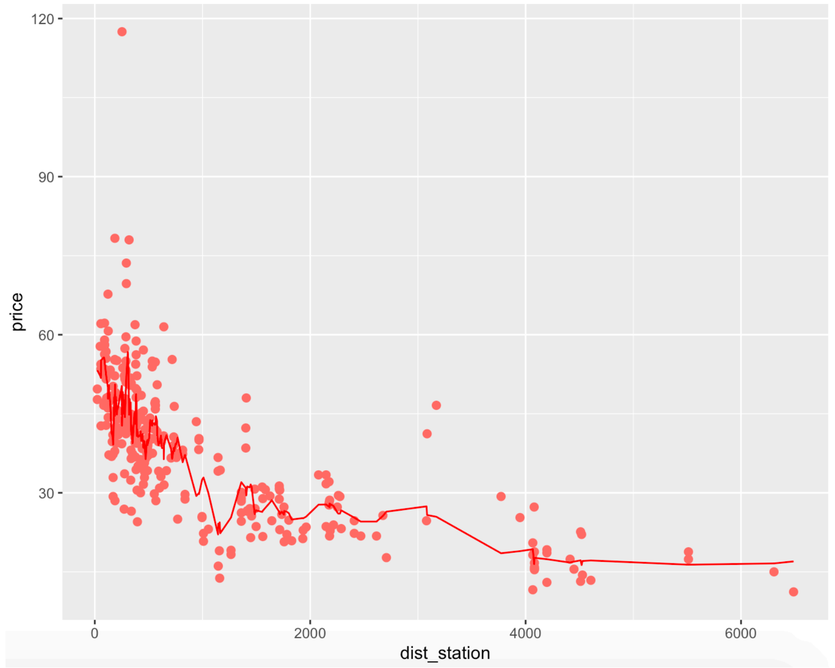
$$f(X) = aX^2 + bX + c$$

Train/ Test Set: Price versus Distance

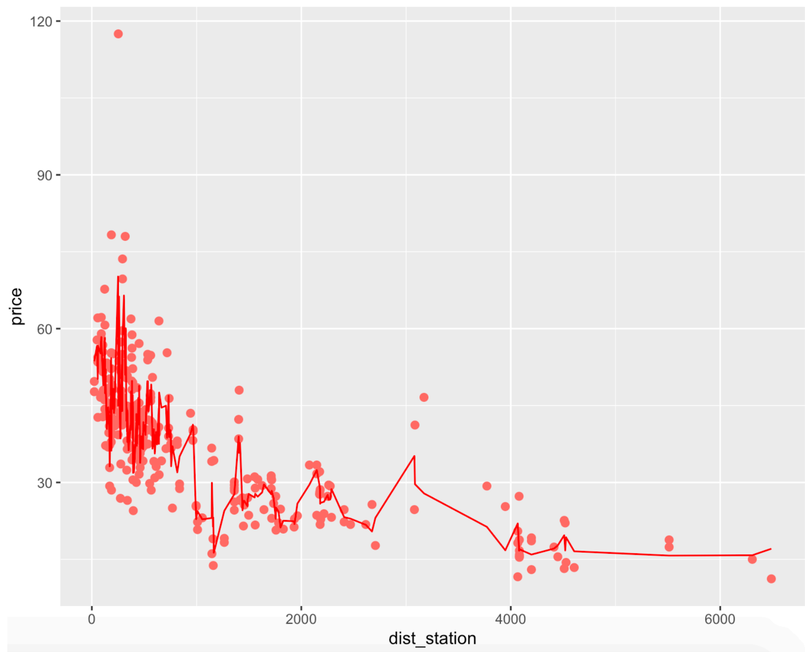
We then consider K-nearest neighbors method



K = 50



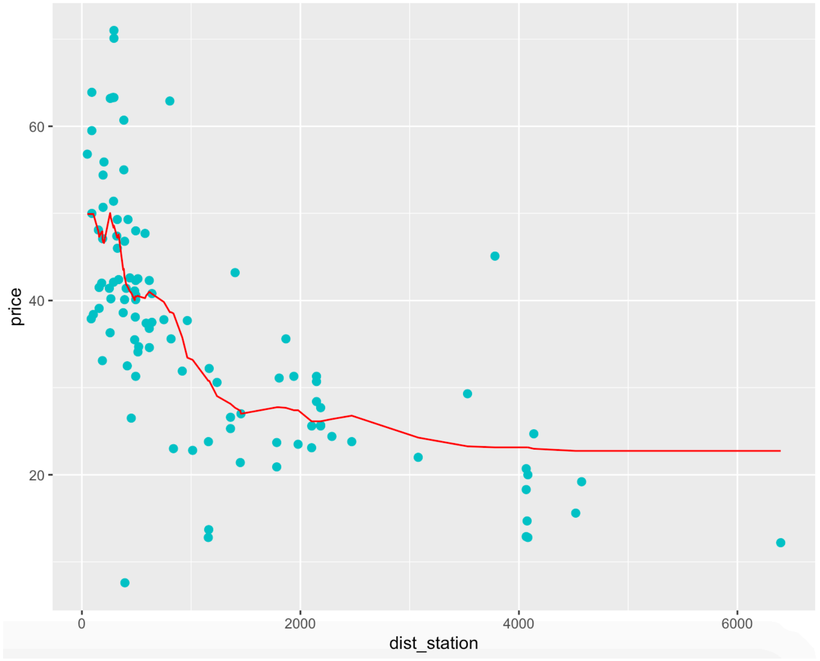
K = 10



K = 3

Training data

Train/ Test Set: Price versus Distance



K = 50

K = 10

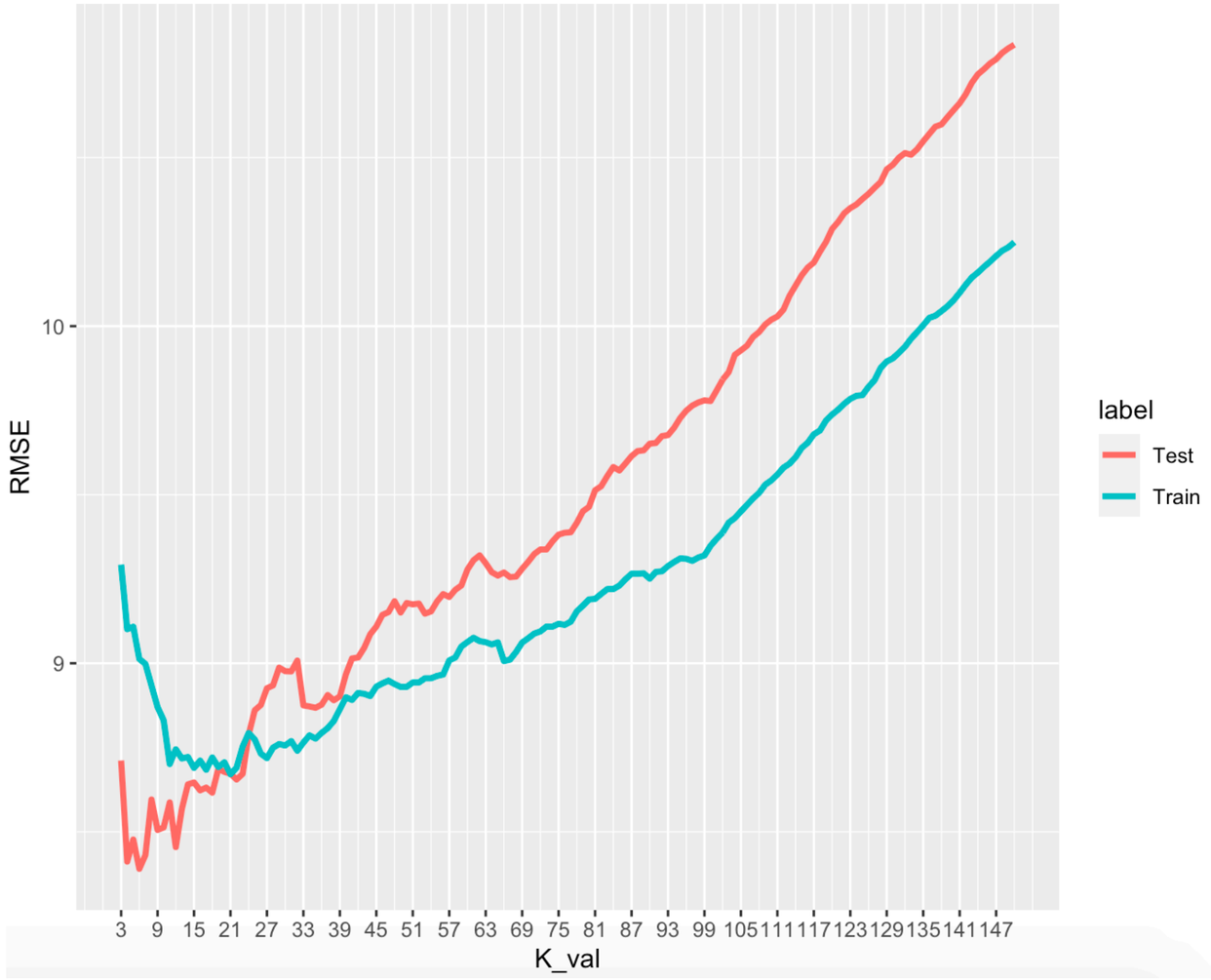
K = 3

Testing data

Train/ Test Set: Price versus Distance

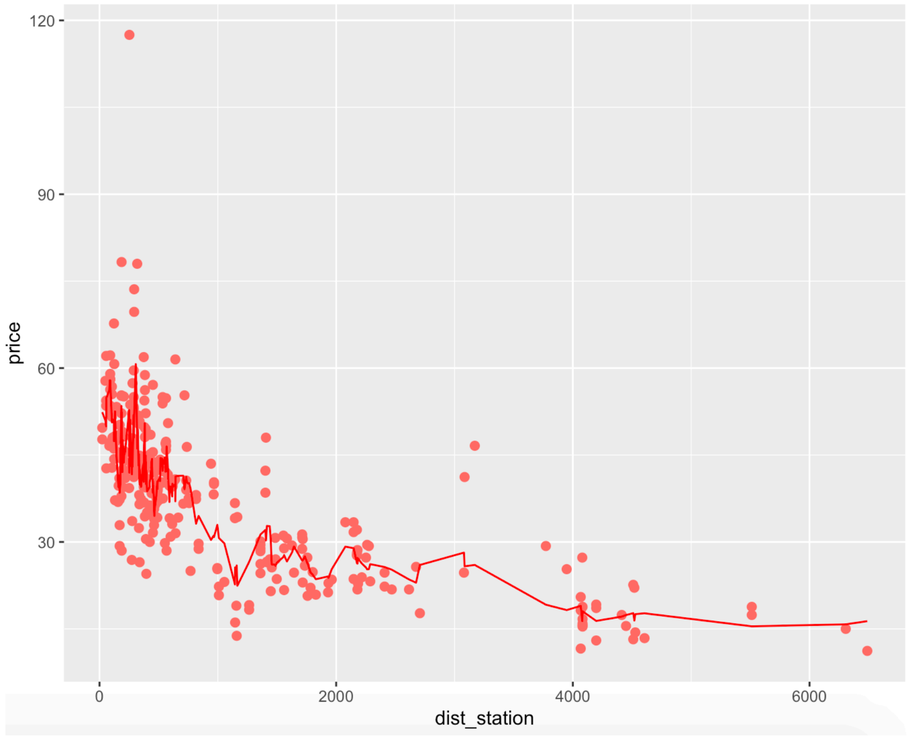
Train data: best K = 22

Test data: best K = 7



Train/ Test Set: Price versus Distance

K-nearest neighbors at “best” K = 7



Training data

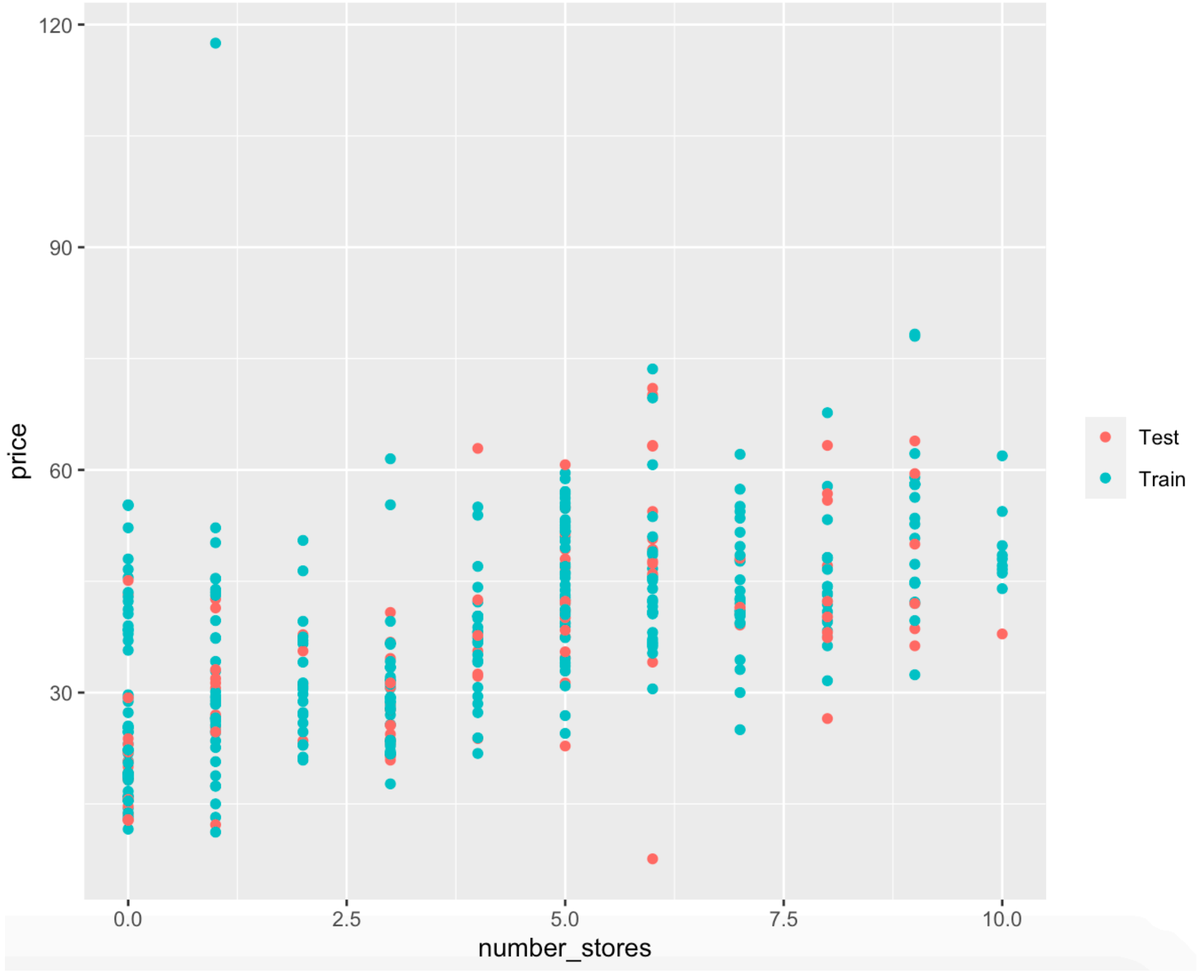
Testing data

Train/ Test Set: Price versus Number of Stores

- Assume that Y = real estate price and X = number of stores
- **Questions**:
    - Between quadratic and linear models, which one we prefer? Should we use higher order polynomial models?
    - For K-nearest neighbors method, what is the best choice of K?

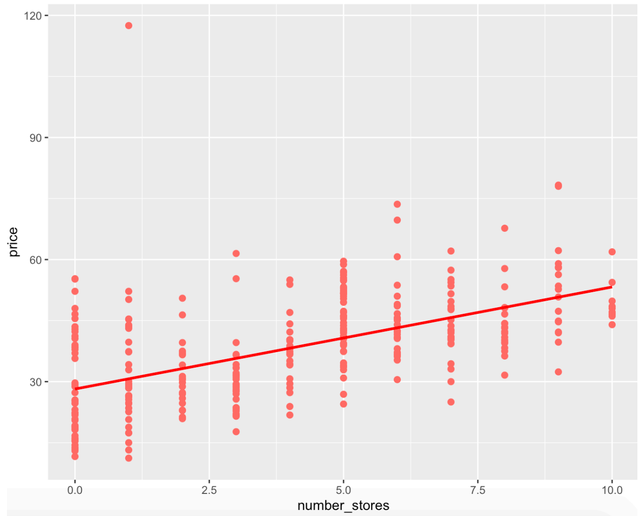
Train/ Test Set: Price versus Number of Stores

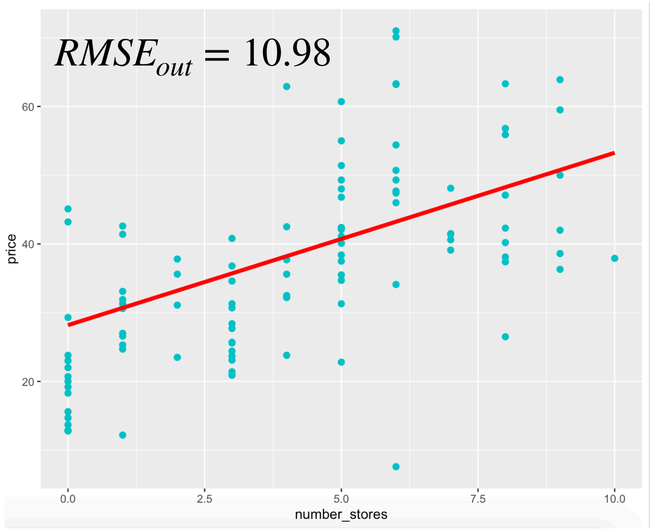
- Now, we consider Y = real estate price, X = number of stores



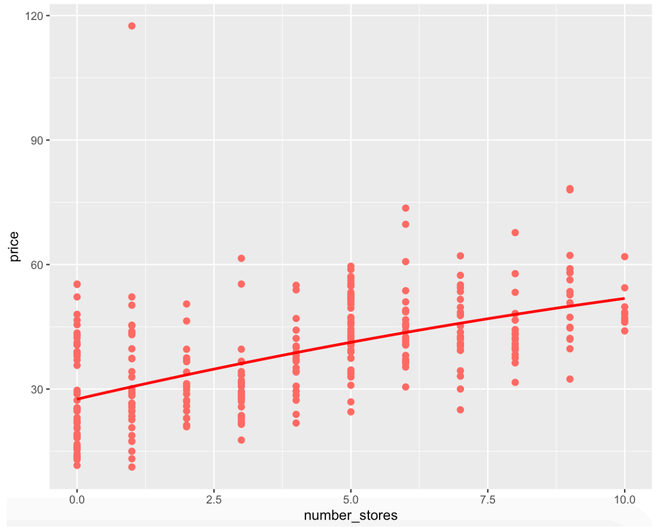
Train/ Test Set: Price versus Number of Stores

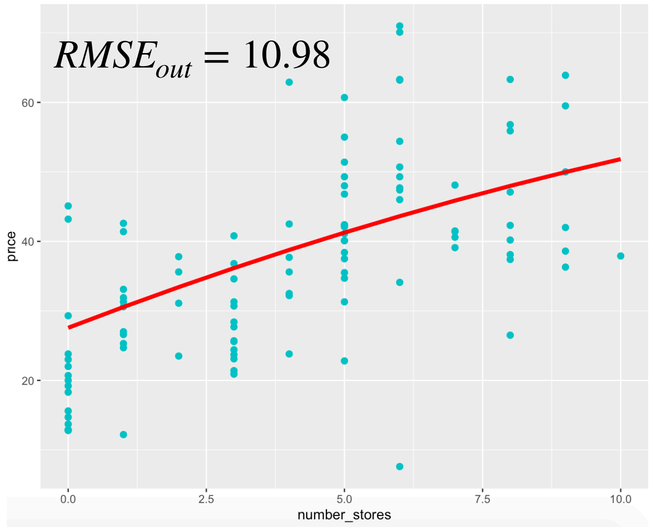
Parametric methods





$$f(X) = aX + b$$





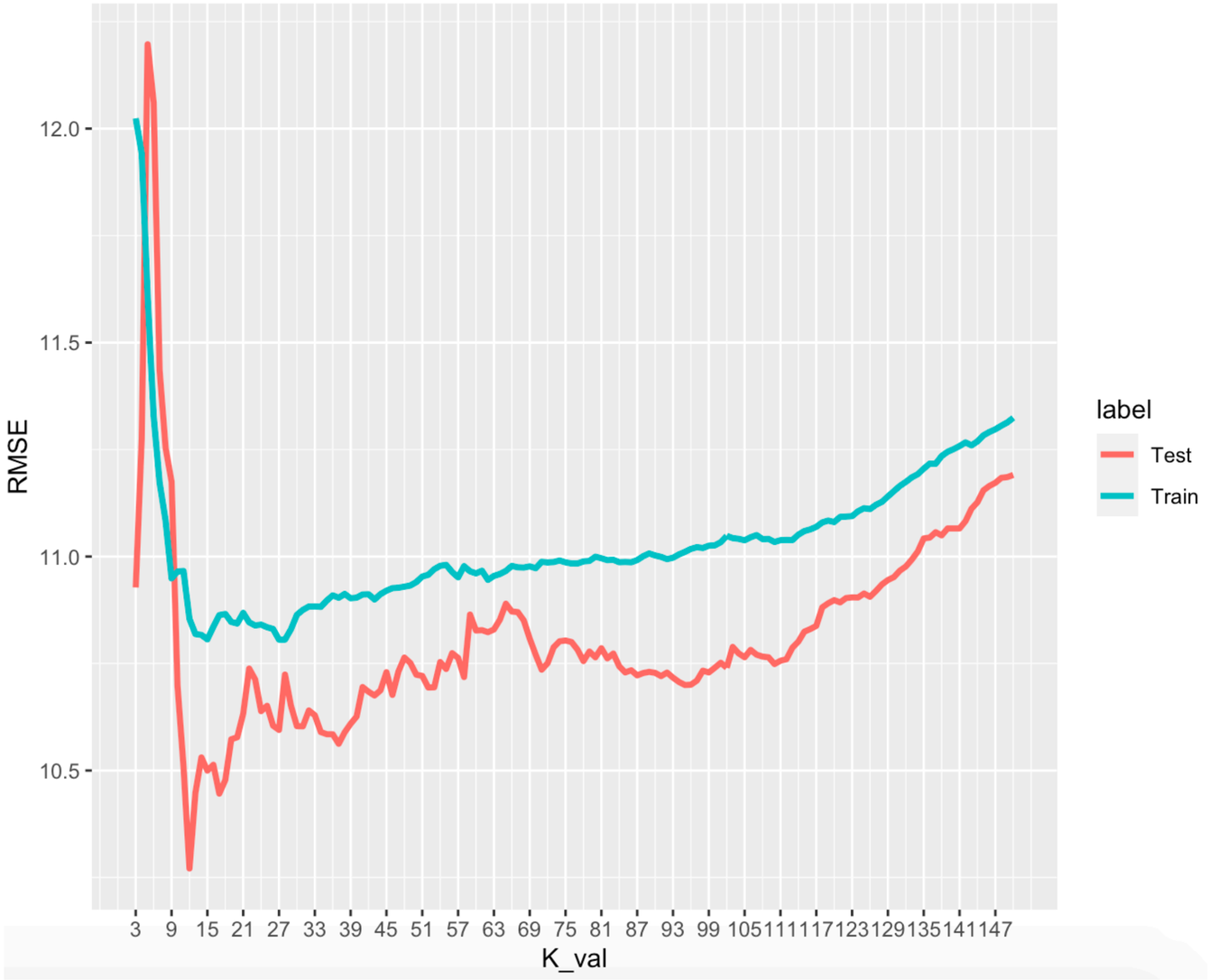
$$f(X) = aX^2 + bX + c$$

Train/ Test Set: Price versus Number of Stores

K-nearest neighbors method

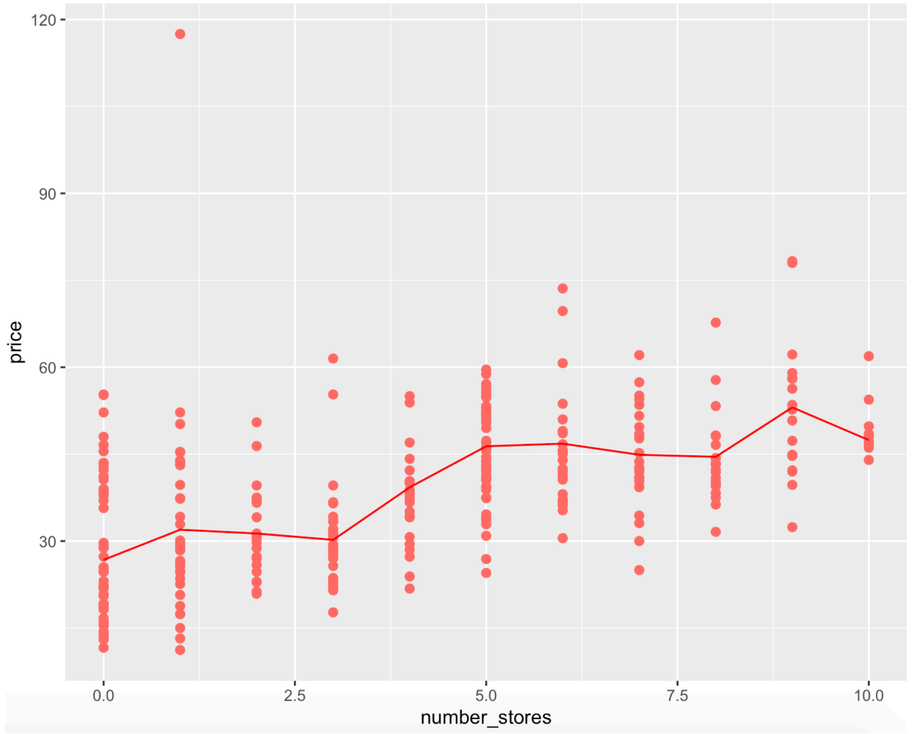
Train data: best K = 28

Test data: best K = 13

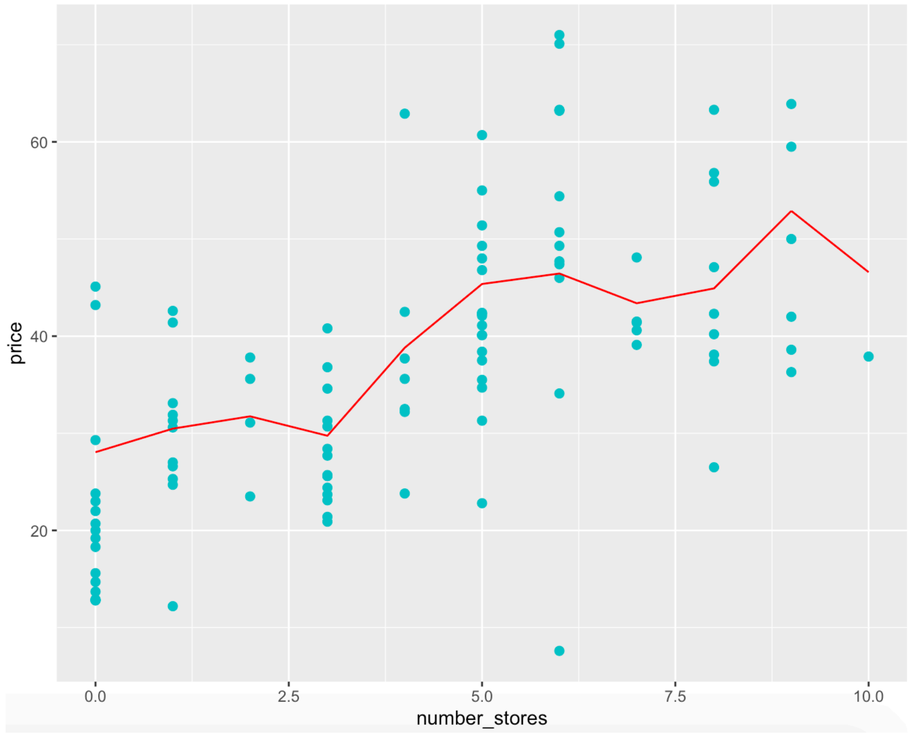


Train/ Test Set: Price versus Number of Stores

K-nearest neighbors at “best” K = 13



Training data



Testing data

Bias-Variance Trade-Off

- Recall what we have seen earlier with K-nearest neighbors method:
    - Large K: high bias, small variance
    - Small K: low bias, large variance

- What are the precise definitions of “bias” and “variance”?

Bias-Variance Trade-Off

- The training data: $(Y_1, X_1), \ldots, (Y_n, X_n)$
- $Y = f(X) + \varepsilon$
- We have a new data point X_* and Y_* is the corresponding value at this point
- Assume that $\hat{f}$ is an estimate of f based on the training data
- **Expected Mean Square Error** at X_* is given by:

$$EMSE = \mathbb{E}\left[(Y_* - \hat{f}(X_*))^2\right]$$

where the expectation is taken with respect to the training data

Bias-Variance Trade-Off

- We need a bit of calculations here
- $EMSE = \mathbb{E}\left[(Y_* - \hat{f}(X_*))^2\right]$

$$= \mathbb{E}\left[\left(Y_* - \mathbb{E}[\hat{f}(X_*)] + \mathbb{E}[\hat{f}(X_*)] - \hat{f}(X_*)\right)^2\right]$$

$$= \mathbb{E}\{(Y_* - \mathbb{E}[\hat{f}(X_*)])^2\} + \mathbb{E}\{\left(\mathbb{E}[\hat{f}(X_*)] - \hat{f}(X_*)\right)^2\}$$

- $Var(Y) = \mathbb{E}(Y^2) - \mathbb{E}^2(Y)$ for any random variable Y
- $\mathbb{E}\{\left(\mathbb{E}[\hat{f}(X_*)] - \hat{f}(X_*)\right)^2\} = Var(\hat{f}(X_*))$

Bias-Variance Trade-Off

- $EMSE = \mathbb{E}\left[(Y_* - \hat{f}(X_*))^2\right]$

$$= \mathbb{E}\{(Y_* - \mathbb{E}[\hat{f}(X_*)])^2\} + Var(\hat{f}(X_*))$$

- Recall that, $Y_* = f(X_*) + \varepsilon$
- $\mathbb{E}\{(Y_* - \mathbb{E}[\hat{f}(X_*)])^2\} = \mathbb{E}\{(f(X_*) + \varepsilon - \mathbb{E}[\hat{f}(X_*)])^2\}$

$$= \mathbb{E}\{(f(X_*) - \mathbb{E}[\hat{f}(X_*)])^2\} + \mathbb{E}(\varepsilon^2)$$

$$= (f(X_*) - \mathbb{E}[\hat{f}(X_*)])^2 + \mathbb{E}(\varepsilon^2)$$

Bias-Variance Trade-Off

- $EMSE = \mathbb{E}\left[(Y_* - \hat{f}(X_*))^2\right]$

$$= (f(X_*) - \mathbb{E}[\hat{f}(X_*)])^2 + Var(\hat{f}(X_*)) + \mathbb{E}(\varepsilon^2)$$

$$\{\text{Bias}(\hat{f}(X_*))\}^2$$

Squared Bias

The error that is introduced by approximating a complex model by a much simpler model

Variance

The amount by which $\hat{f}$ would change if we estimate it using a different training data set

Bias-Variance Trade-Off

- $EMSE = \mathbb{E}\left[(Y_* - \hat{f}(X_*))^2\right]$ = Squared Bias + Variance + $\mathbb{E}(\varepsilon^2)$

- Ideally, we want to make Squared Bias, Variance as small as possible

- How to make these terms small?

Bias-Variance Trade-Off

- Here are rules of thumbs:
    - **Small variance**: Less complex model
    - **Small bias**: More complex model

- The good model is one that balances the bias and variance