Introduction to Data Science, Topic 6

- Instructor: Professor Henry Horng-Shing Lu,

  Institute of Statistics, National Chiao Tung University, Taiwan

  Email: hslu@stat.nctu.edu.tw

- WWW: http://www.stat.nctu.edu.tw/misg/hslu/course/DataScience.htm
- Reference:

  M. A. Pathak, Beginning Data Science with R, 2014, Springer-Verlag.
- Evaluation: Homework: 50%, Term Project: 50%
- Office hours: By appointment

Course Outline

- **Introduction of data science**
- **Introduction of R**
- **Data Visualization**
- **Exploratory Data Analysis**
- **Regression**
- **Classification**
- **Text Mining**
- **Clustering**

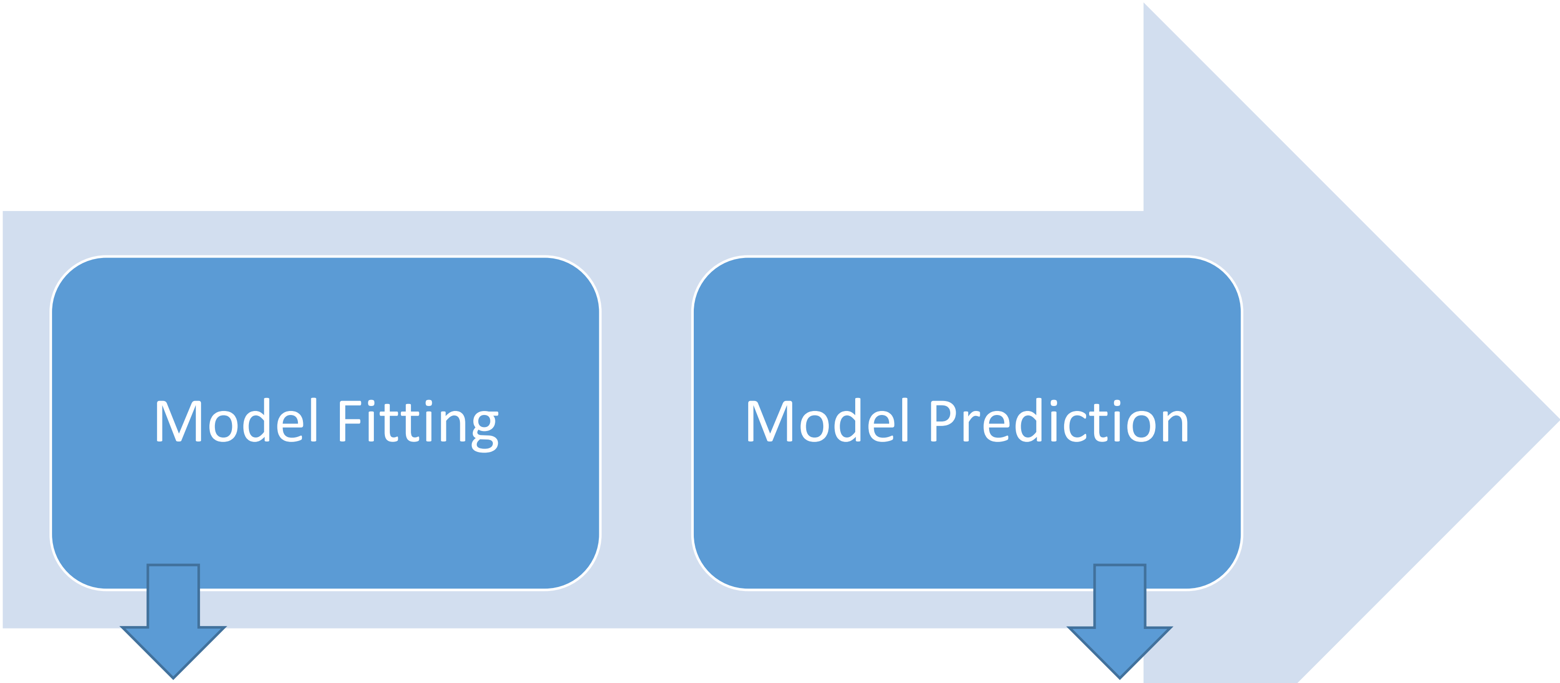
Regression with R

References:

Ch. 6, M. A. Pathak, Beginning Data Science with R, 2014, Springer-Verlag.

https://en.wikipedia.org/wiki/Data_science

Regression Techniques



For a given dataset with variables $X_1, X_2, \dots, X_n$ and Y we calculate the model parameters w that best meet certain criteria

Given a dataset with the predictor variables $X_1, X_2, \dots, X_n$ and model parameters $\beta_0, \beta_1, \dots, \beta_n$, we calculate the value of the target variable $\hat{Y}$. The dataset could be the same one we use for model fitting or it could also be a new dataset.

Regression Models

- Parametric Regression Models
  - Simple Linear Regression
  - Multivariate Linear Regression
  - Log-Linear Regression

- Nonparametric Regression Models
  - Locally Weighted Regression
  - Kernel Regression
  - Regression Trees

Simple Linear Regression

$$y = f(x; \beta) = \beta_0 + \beta_1 x + \varepsilon,$$

$$y, x, \varepsilon \in R, \beta_0, \beta_1 \in R$$

β_0: intercept

β_1: slope

Simple Linear Regression - Loss Function

- Residuals

$$\varepsilon_i = y_i - \beta_0 - \beta_1 x_i$$

- Sum of squared residuals

$$\sum_i \varepsilon_i^2 = \sum_i (y_i - \beta_0 - \beta_1 x_i)^2$$

一次微分取極值，但要確保是最小值的話
就要再取二次微分, 如果二次微分的值大於零
則可以保證會是local minima(二次微分的值大於
零 會是local maxima)

Simple Linear Regression – Model Fitting

- We obtain the coefficients having the minimum squared error which is also called the least-squares method.

$$\beta \quad = \underset{\beta}{\operatorname{argmin}} \sum_i \varepsilon_i^2 = \underset{\beta_0,\beta_1}{\operatorname{argmin}} \sum_i (y_i - \beta_0 - \beta_1 x_i)^2$$

今天要找二維的最小值
先對其一的變數微分 在對另一個微分

$$solve \Rightarrow$$

$$\hat{\beta}_1 = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sum_i (x_i - \overline{x})^2},$$

$$\hat{\beta}_0 = \overline{y} - \hat{\beta}_1 \overline{x}$$

Simple Linear Regression - R

- Import Data: iris.csv

- > str(iris)
- 'data.frame':	150 obs. of 5 variables:
- $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
- $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
- $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
- $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
- $ Species   : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

Simple Linear Regression – Coefficient

```
> fit_1<-lm(Sepal.Length~Petal.Length,data=iris)
> fit_1

Call:
lm(formula = Sepal.Length ~ Petal.Length, data = iris)

Coefficients:
 (Intercept)  Petal.Length
    4.3066       0.4089
```

$\boxed{\hat{\beta}_0}$ $\hat{\beta}_1$

$$\text{Sepal.Length} = 4.3066 + 0.4089 \times \text{Petal.Length} + \varepsilon$$

Simple Linear Regression – Without Intercept

- Model:

$$y = f(x; w) = \hat{\beta}_1 x + \varepsilon, \qquad y, x, \varepsilon \in R, \qquad \hat{\beta}_1 \in R$$

$\hat{\beta}_1$: slope

Sepal.Length =1.349 × Petal.Length +ε

- Implement:

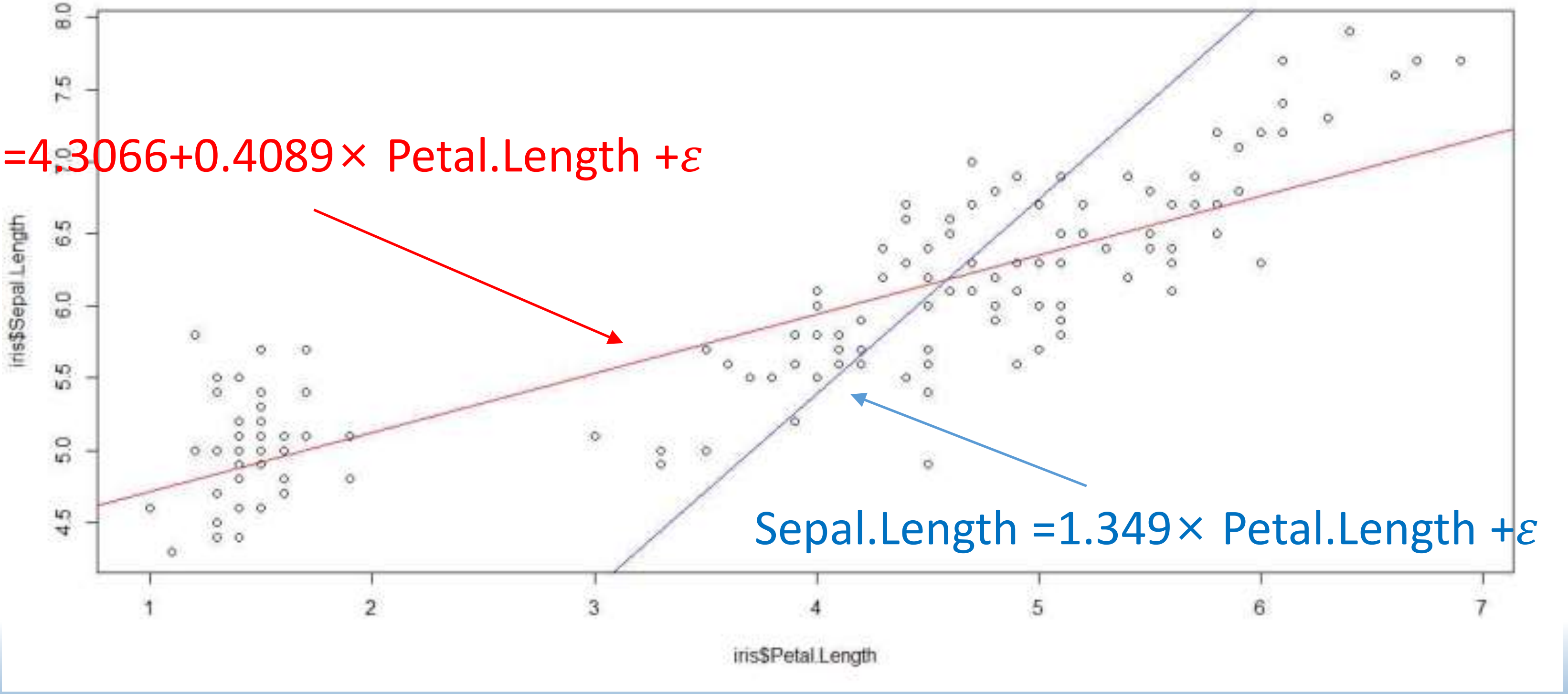
```
> fit_2<-lm(Sepal.Length~Petal.Length-1,data=iris)
> fit_2

Call:
lm(formula = Sepal.Length ~ Petal.Length - 1, data = iris)

Coefficients:
Petal.Length
       1.349
```

$\hat{\beta}_1$

assume B1 = 0

Simple Linear Regression - Compare

```
fit_1<-lm(Sepal.Length~Petal.Length,data=iris)
fit_2<-lm(Sepal.Length~Petal.Length-1,data=iris)
plot(iris$Petal.Length,iris$Sepal.Length)
abline(fit_1,col='red')
abline(fit_2,col='blue')
```



$\text{Sepal.Length} = 4.3066 + 0.4089 \times \text{Petal.Length} + \varepsilon$

$\text{Sepal.Length} = 1.349 \times \text{Petal.Length} + \varepsilon$

Simple Linear Regression - Categorical Variable

- “Species” is a nominal variable, and these names are as below.

```
> class(iris$Species)
[1] "factor“



> table(iris$Species)

    setosa versicolor  virginica
        50         50         50
```

Simple Linear Regression - Categorical Variable

```
> fit_3<-lm(Sepal.Length~Species,data=iris)
> fit_3
```

$$\text{Sepal.Length} = 5.006 + 0.930 \times \text{Speciesversicolor} + 1.582 \times \text{Speciesvirginica} + \varepsilon$$

```
Call:
lm(formula = Sepal.Length ~ Species, data = iris)

Coefficients:
  (Intercept)  Speciesversicolor   Speciesvirginica
      5.006            0.930             1.582
```

Simple Linear Regression - Categorical Variable

- Let’s show the design matrix by model.matrix function.

```
> model.matrix(fit_3)
   (Intercept) Speciesversicolor Speciesvirginica
1          1             0             0
2          1             0             0
3          1             0             0
4          1             0             0
5          1             0             0
6          1             0             0
 …
```

Dummy variables

A matrix of the regression model: $\boldsymbol{Y} = \boldsymbol{X}\hat{\beta} + \boldsymbol{\varepsilon}$

Simple Linear Regression – Group by mean

- The model coefficient for a value of make is the average price of cars of that make.

```
> by(iris$Sepal.Length,iris$Species,mean)
iris$Species: setosa
[1] 5.006
---------------------------------------------------------------------------
iris$Species: versicolor
[1] 5.936
---------------------------------------------------------------------------
iris$Species: virginica
[1] 6.588
```

Simple Linear Regression - Model Diagnostics

- How well it fits the data?
  - The summary() function prints the five-number summary of the model residuals.

```
> summary(fit_1)

Call:
lm(formula = Sepal.Length ~ Petal.Length, data = iris)

Residuals:
     Min       1Q   Median       3Q      Max
-1.24675 -0.29657 -0.01515  0.27676  1.00269

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.30660    0.07839   54.94   <2e-16 ***
Petal.Length  0.40892    0.01889   21.65   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4071 on 148 degrees of freedom
Multiple R-squared:   0.76,	Adjusted R-squared:  0.7583
F-statistic: 468.6 on 1 and 148 DF,  p-value: < 2.2e-16
```

Extension – Error Assumption

- Given a sample of n individuals, we observe data $(y_1, x_1), (y_2, x_2), \cdots, (y_n, x_n)$.
- Variables y and x are assumed to be related through

$$\mathrm{E}(y_i|x_i) = \mu_{y|x} = \beta_0 + \beta_1 x_i$$

or

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i,$$

where the error $\epsilon_i \overset{\text{i.i.d.}}{\sim} N(0, \sigma^2)$, and $\beta_0 =$ intercept, $\beta_1 =$ slope.

Extension – Model Assumptions

1. y_i and x_i are related in a straight line fashion (linear).
2. The variance of the error (or y) is the same along the whole line and the observations are independent (equal variance and independent).
3. y is normally distributed around the line (normal). (Note: The larger n is, the less important this assumption is for the tests and confidence intervals calculation).

Extension – Properties of OLSE

- The least-squares estimators $\hat{\beta}_0$ and $\hat{\beta}_1$ are linear combinations of observations $y_i, i = 1, \cdots, n$.
- It can be shown that $\mathrm{E}(\hat{\beta}_0) = \beta_0$ and $\mathrm{E}(\hat{\beta}_1) = \beta_1$.
- **Gauss-Markov theorem**: for the above linear regression model, the least-squares estimators have minimum variance when compared with all other unbiased estimators that are linear combinations of the y_i.
- The least-squares estimators are **best linear unbiased estimators**.

Extension – Variance Estimation

- An estimator of σ^2 is given by

$$\hat{\sigma}^2 = s^2_{y|x} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-2} = \frac{\mathsf{SSE}}{n-2},$$

where SSE is called the *error sum of squares.*

- It can be shown that

$$\mathrm{Var}(\hat{\beta}_1) = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}.$$

We can estimate $\mathrm{Var}(\hat{\beta}_1)$ by

$$\hat{\mathrm{Var}}(\hat{\beta}_1) = \frac{s^2_{y|x}}{\sum_{i=1}^{n}(x_i - \bar{x})^2}.$$

Extension – Test Between y and x

- Use t-test or CI for β_1:

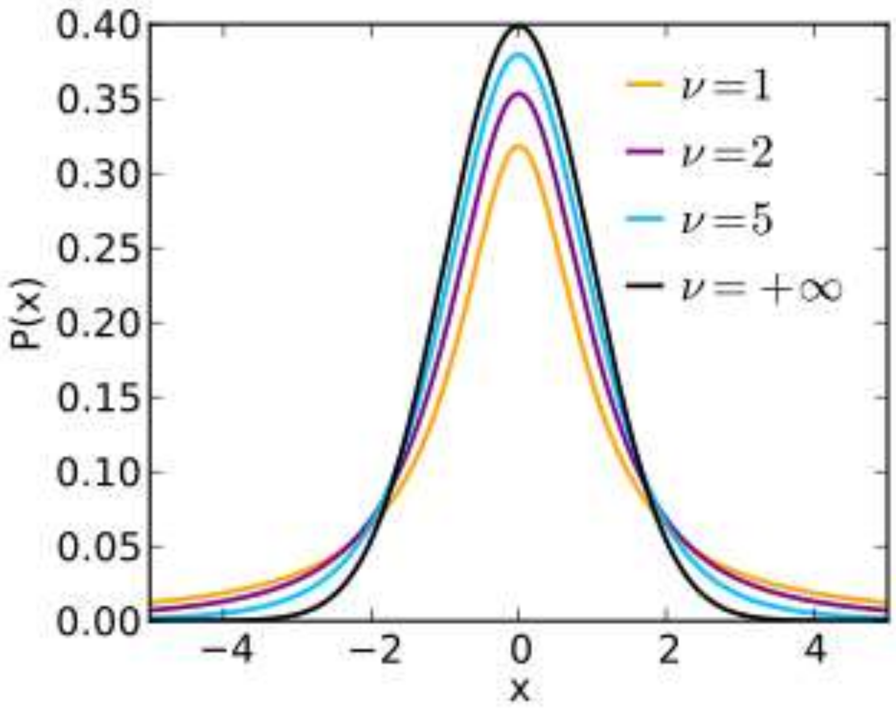
$$H_0 : \beta_1 = 0 \text{ versus } H_A : \beta_1 \neq 0$$

- Test statistic

$$t = \frac{\hat{\beta}_1}{\sqrt{\hat{\text{Var}}(\hat{\beta}_1)}} = \frac{\hat{\beta}_1}{\sqrt{s^2_{y|x} / \sum_{i=1}^{n}(x_i - \bar{x})^2}} \sim t(n-2)$$

- $(1-\alpha) \times 100\%$ CI for β_1

$$\hat{\beta}_1 \pm t_{1-\frac{\alpha}{2}}(n-2)\sqrt{s^2_{y|x} / \sum_{i=1}^{n}(x_i - \bar{x})^2}$$



Extension – Analysis of Variance

- The regression can be further understood in the framework of "analysis of variance", which generally means splitting total variation of y, i.e., $\sum_{i=1}^{n}(y_i - \bar{y})^2$ into component parts.
- We can show that

$$
\begin{array}{cccccc}
\sum_{i=1}^{n}(y_i - \bar{y})^2 & = & \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 & + & \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \\
 & = & \hat{\beta}_1^2 \sum_{i=1}^{n}(x_i - \bar{x})^2 & + & \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \\
\text{total} & = & \text{variation due} & + & \text{residual} \\
\text{variation} & & \text{to regression } (x) & & \text{variation} \\
\text{SST} & = & \text{SSR} & + & \text{SSE}
\end{array}
$$

Extension – R Square

- A summary measure of regression line is the coefficient of determination (R^2), which represents the fraction of variability explained by the regression.

$$R^2 = \frac{SSR}{SSY}$$

Extension – ANOVA table for regression

- We can assess the contribution of x by testing

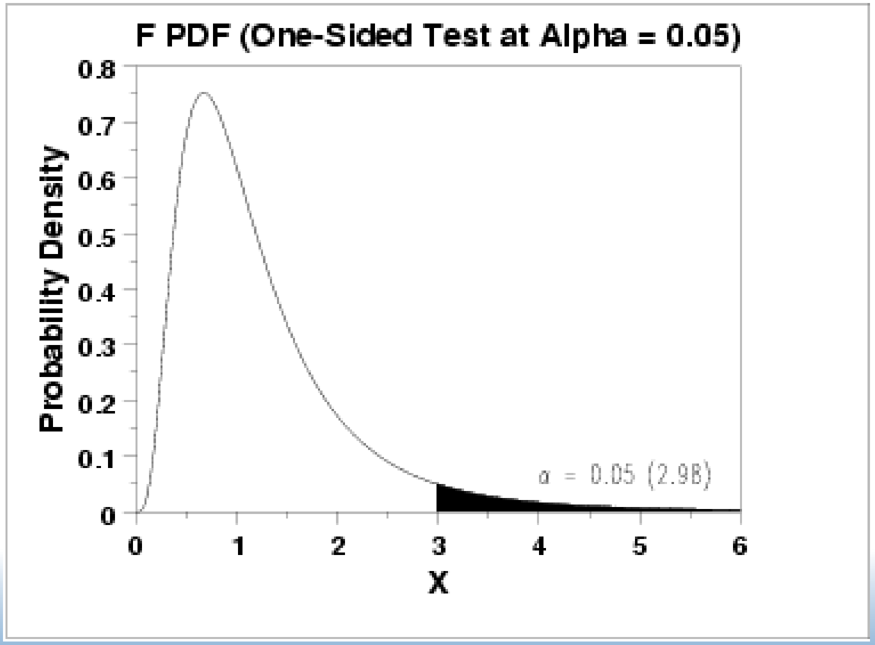
  $H_0 : x$ is not needed $(\beta_1 = 0)$ vs. $H_A : x$ is needed $(\beta_1 \neq 0)$

  by calculating $\dfrac{\text{MSR}}{\text{MSE}} = \dfrac{\text{SSR}/1}{\text{SSE}_{/(n-2)}} = F_{obs}$ and rejecting H_0 if this ratio exceed $F_{1-\alpha}(1, n-2)$ (the F-test).
- Notice that the F-test is equivalent to the t-test for the regression coefficient β_1,
  i.e., $F_{obs} = t^2$ and $F_{\alpha}(1, n-2) = t^2_{\alpha/2}(n-2)$.

Extension – ANOVA table for regression

source	df	sum of squares (SS)	mean square (MS)	variance ratio (F)
regression	1	$\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2$=SSR	MSR=SSR/1	MSR/MSE
residual	$n-2$	$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$=SSE	MSE=SSE/(n-2)	
total	$n-1$	$\sum_{i=1}^{n}(y_i - \bar{y})^2$=SSY		

$$F = \frac{MSR}{MSE}$$



Simple Linear Regression - Model Diagnostics

- Residual
  - The residual standard error is a measure of how well the model fits the data or goodness of fit.

```
> residuals(fit_1)
         1          2          3          4 
 0.22090540  0.02090540 -0.13820238 -0.31998683 
         5          6          7          8 
 0.12090540  0.39822871 -0.27909460  0.08001317 
…

# fit_1$residuals  (same result)
```

Simple Linear Regression – Model Prediction

- Fittind value

```
> predict(fit_1)
       1        2        3        4        5        6 
4.879095 4.879095 4.838202 4.919987 4.879095 5.001771 
       7        8        9       10       11       12 
4.879095 4.919987 4.879095 4.919987 4.919987 4.960879 
…
# fit_1$fitted.values (same result)
```

- Prediction value

```
> set.seed(123);newdata<-data.frame(Petal.Length=fit_1$fitted.values[1:5]+runif(1))
> predict(fit_1,newdata)
       1        2        3        4        5 
6.419371 6.419371 6.402649 6.436092 6.419371 
```

Regression Models

- Parametric Regression Models
  - Simple Linear Regression
  - Multivariate Linear Regression
  - Log-Linear Regression

- Nonparametric Regression Models
  - Locally Weighted Regression
  - Kernel Regression
  - Regression Trees

Multivariate Linear Regression

Let

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \qquad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}$$

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}, \qquad \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

With this compact notation, the linear regression model can be written in the form

$$\mathbf{y} = \mathbf{X}\beta + \epsilon$$

Multivariate Linear Regression

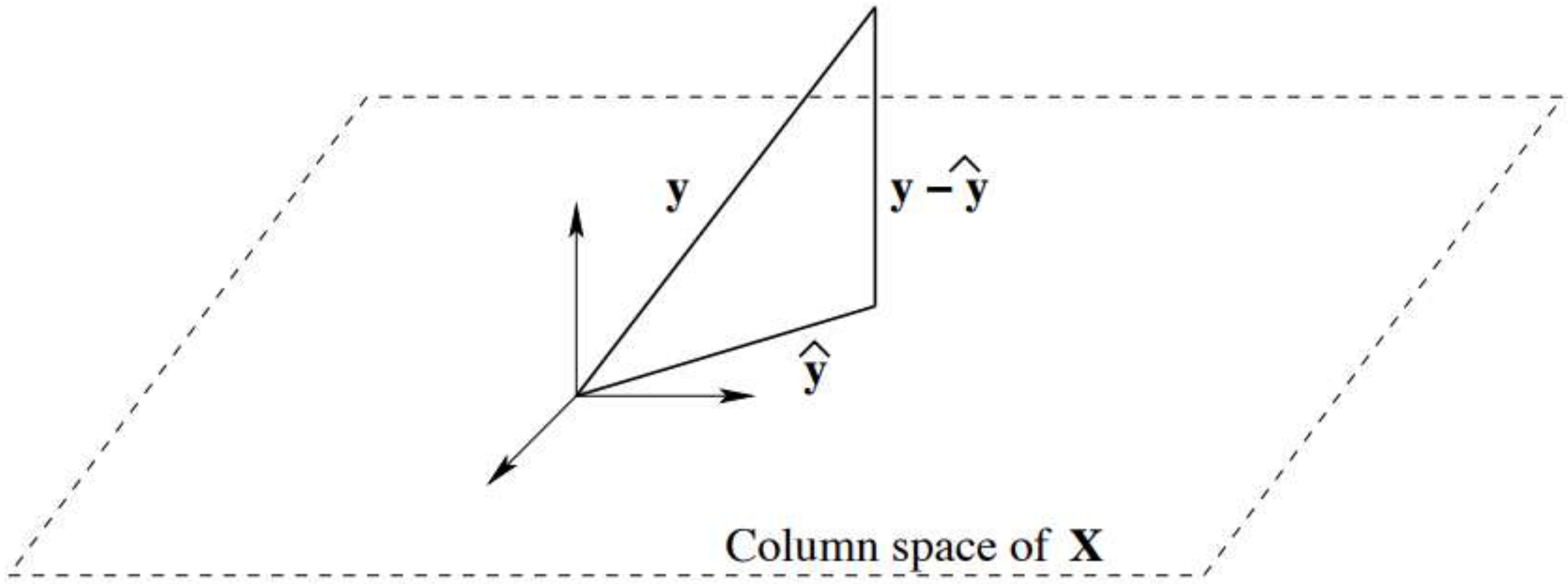
In linear algebra terms, the least-squares parameter estimates β are the vectors that minimize

$$\sum_{i=1}^{n} \epsilon_i^2 = \epsilon'\epsilon = (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$$

Any expression of the form $\mathbf{X}\beta$ is an element of a (at most) $(k+1)$-dimensional hyperspace in $\mathbb{R}^n$ spanned by the $(k+1)$ columns of X. Imagine the columns of $\mathbf{X}$ to be fixed, they are the data for a specific problem, and imagine β to be variable. We want to find the "best" β in the sense that the sum of squared residuals is minimized.

Multivariate Linear Regression

Here $\hat{\mathbf{y}}$ is the projection of the n-dimensional data vector $\mathbf{y}$ onto the hyperplane spanned by $\mathbf{X}$.



Multivariate Linear Regression – Model Fitting

These vector normal equations are the same normal equations that one could obtain from taking derivatives. To solve the normal equations (i.e., to find the parameter estimates $\hat{\beta}$), multiply both sides with the inverse of $\mathbf{X}'\mathbf{X}$. Thus, the least-squares estimator of β is (in vector form)

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

Multivariate Linear Regression - Model Fitting

```
> fit_4<-lm(Sepal.Length~Sepal.Width+Petal.Length+Petal.Width,data=iris)
> fit_4

Call:
lm(formula = Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width,
    data = iris)

Coefficients:
 (Intercept)   Sepal.Width  Petal.Length   Petal.Width
      1.8560       0.6508       0.7091      -0.5565
```

$$Sepal.Length = 1.856 + 0.6508 \times Sepal.Width + 0.7091 \times Petal.Length - 0.5565 \times Petal.Width + \varepsilon$$

Multivariate Linear Regression – Summery

```
> summary(fit_4)
Call:
lm(formula = Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width,
    data = iris)

Residuals:
     Min       1Q   Median       3Q      Max
-0.82816 -0.21989  0.01875  0.19709  0.84570

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.85600    0.25078   7.401 9.85e-12 ***
Sepal.Width   0.65084    0.06665   9.765  < 2e-16 ***
Petal.Length  0.70913    0.05672  12.502  < 2e-16 ***
Petal.Width  -0.55648    0.12755  -4.363 2.41e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3145 on 146 degrees of freedom
Multiple R-squared:  0.8586,	Adjusted R-squared:  0.8557
F-statistic: 295.5 on 3 and 146 DF,  p-value: < 2.2e-16
```

Regression Models

- Parametric Regression Models
  - Simple Linear Regression
  - Multivariate Linear Regression
  - Log-Linear Regression

- Nonparametric Regression Models
  - Locally Weighted Regression
  - Kernel Regression
  - Regression Trees

Log-Linear Regression Models

- If the data has a nonlinear trend, a linear regression model cannot fit the data accurately. One approach for modeling nonlinear data is to fit a model for a transformed variable. A commonly used transformation is to use the log() function on the variables.

```
> fit_5<-lm(Sepal.Length~log(Petal.Length),data=iris)
> fit_5

Call:
lm(formula = Sepal.Length ~ log(Petal.Length), data = iris)

Coefficients:
      (Intercept)  log(Petal.Length)
           4.481              1.159
```

Log-Linear Regression Models

- The relationship between the variables is:

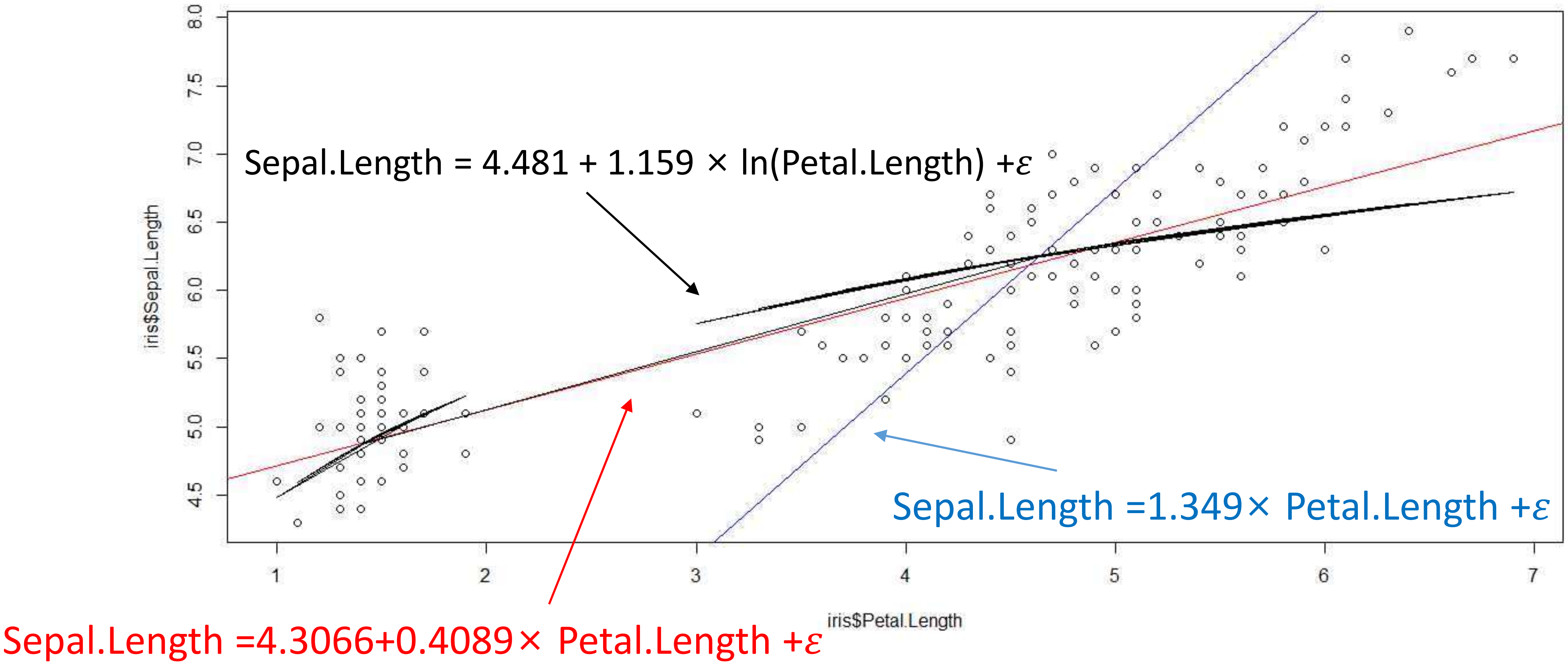
$$\text{Sepal.Length} = 4.481 + 1.159 \times \ln(\text{Petal.Length}) + \varepsilon$$

Log-Linear Regression Models - Summary

```
> summary(fit_5)

Call:
lm(formula = Sepal.Length ~ log(Petal.Length), data = iris)

Residuals:
     Min       1Q   Median       3Q      Max
-1.32471 -0.31249 -0.02543  0.29302  1.26704

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        4.48138    0.08543   52.46   <2e-16 ***
log(Petal.Length)  1.15907    0.06501   17.83   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.4683 on 148 degrees of freedom
Multiple R-squared:  0.6823,	Adjusted R-squared:  0.6802
F-statistic: 317.9 on 1 and 148 DF,  p-value: < 2.2e-16
```

Simple Linear Regression v.s. Log-Linear Regression Models



Regression Models

- Parametric Regression Models
  - Simple Linear Regression
  - Multivariate Linear Regression
  - Log-Linear Regression
- Nonparametric Regression Models
  - Locally Weighted Regression
  - Kernel Regression
  - Regression Trees

Nonparametric Regression Models

- Nonparametric statistics is the branch of statistics that is not based solely on parameterized families of probability distributions (common examples of parameters are the mean and variance). Nonparametric statistics is based on either being distribution-free or having a specified distribution but with the distribution's parameters unspecified. Nonparametric statistics includes both descriptive statistics and statistical inference.

Locally Weighted Regression(LWR)

- LOESS and LOWESS (locally weighted scatterplot smoothing) are two strongly related non-parametric regression methods that combine multiple regression models in a k-nearest-neighbor-based meta-model. "LOESS" is a later generalization of LOWESS; although it is not a true acronym, it may be understood as standing for "LOcal regrESSion"

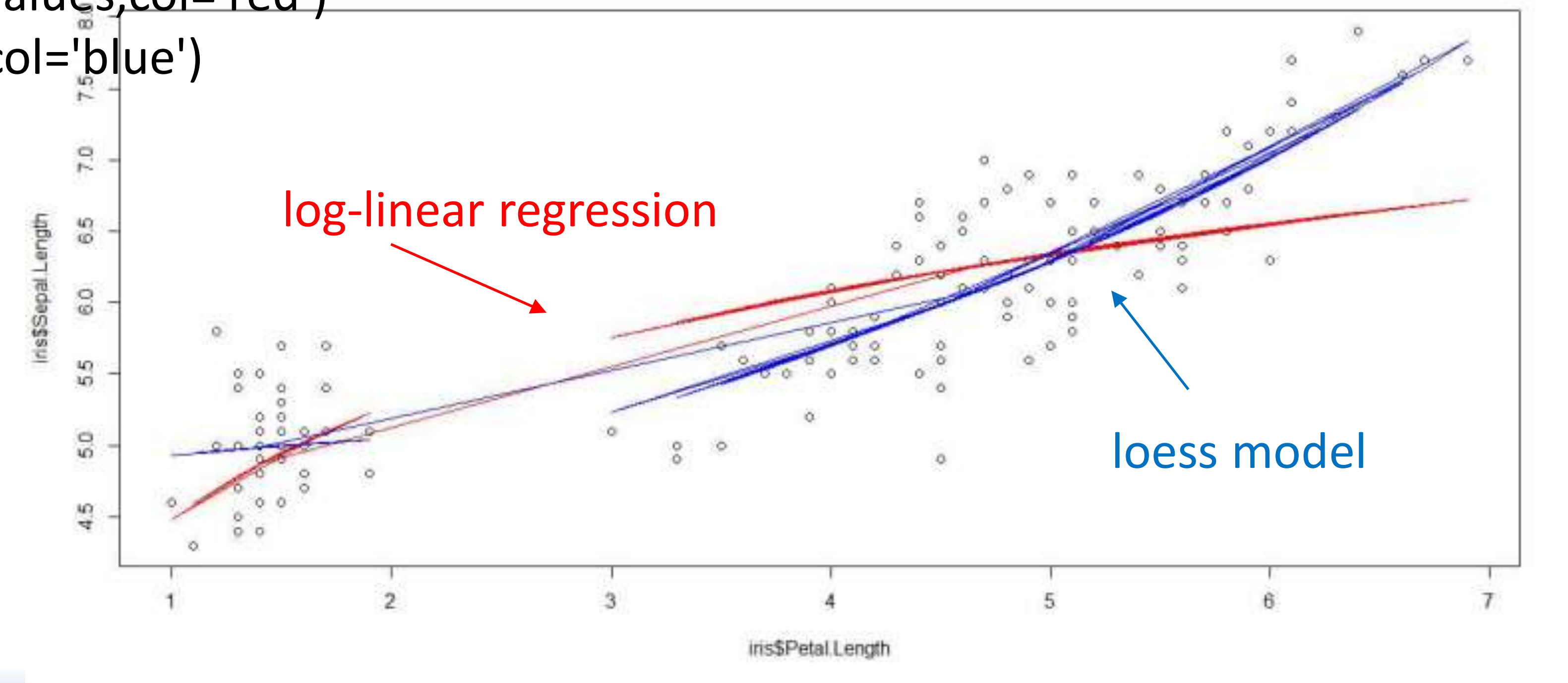
Locally Weighted Regression(LWR)

```
> fit_6<-loess(Sepal.Length~Petal.Length,data=iris)
> summary(fit_6)
Call:
loess(formula = Sepal.Length ~ Petal.Length, data = iris)

Number of Observations: 150
Equivalent Number of Parameters: 4.11
Residual Standard Error: 0.363
Trace of smoother matrix: 4.47  (exact)

Control settings:
  span     :  0.75
  degree   :  2
  family   :  gaussian
  surface  :  interpolate          cell = 0.2
  normalize:  TRUE
 parametric:  FALSE
drop.square:  FALSE
```

Locally Weighted Regression(LWR)

- Compare log-linear regression model and loess model

```
plot(iris$Petal.Length,iris$Sepal.Length)
lines(iris$Petal.Length,fit_5$fitted.values,col='red')
lines(iris$Petal.Length,fit_6$fitted,col='blue')
```



Regression Models

- Parametric Regression Models
  - Simple Linear Regression
  - Multivariate Linear Regression
  - Log-Linear Regression

- Nonparametric Regression Models
  - Locally Weighted Regression
  - Kernel Regression
  - Regression Trees

Kernel Regression

- Kernel regression is another nonparametric approach where we compute the value of the predictor variable at each data point by taking a weighted average of the target variable at all data points. The weights are given by the kernel function, which we can think of as a measure of distance between two data points. The data points nearer to the candidate data point have high weight and those further away from the data point have low weight.

Kernel Regression – Kernel Functions

1. Gaussian kernel:

$$K(x, x_i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-x_i)^2}{2h^2}}.$$

2. Uniform kernel:

$$K(x, x_i) = \frac{1}{2} I\left(\left|\frac{x - x_i}{h}\right| \leq 1\right).$$

3. Triangular kernel:

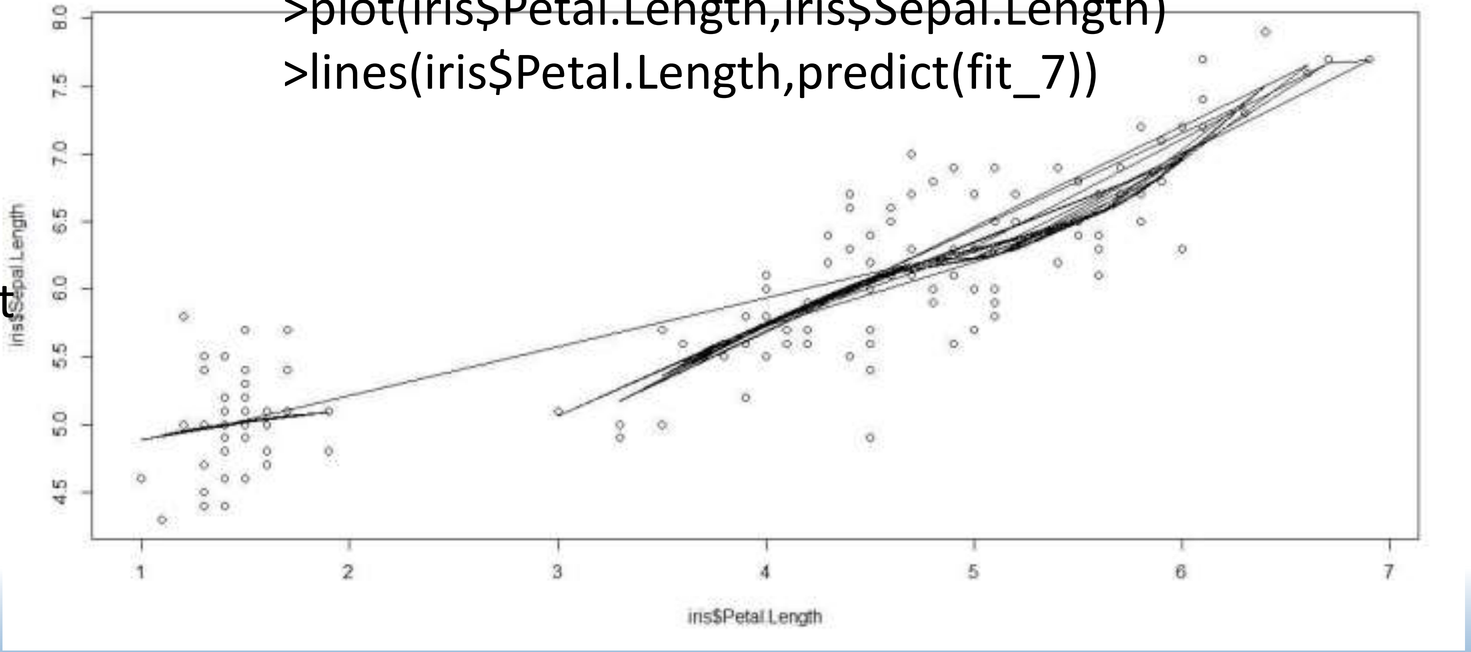
$$K(x, x_i) = \left(1 - \left|\frac{x - x_i}{h}\right|\right) I\left(\left|\frac{x - x_i}{h}\right| \leq 1\right).$$

4. Epanechnikov kernel:

$$K(x, x_i) = \frac{3}{4}\left(1 - \left(\frac{x - x_i}{h}\right)^2\right) I\left(\left|\frac{x - x_i}{h}\right| \leq 1\right).$$

Kernel Regression – Kernel Functions

```
> library(np)
> fit_7<-npreg(Sepal.Length~Petal.Length,data=iris,ckertype='gaussian', ckerorder=2)
> summary(fit_7)

Regression Data: 150 training points, in 1 variable(s)
            Petal.Length
Bandwidth(s):    0.2070386

Kernel Regression Estimator: Local-Constant
Bandwidth Type: Fixed
Residual standard error: 0.3441714
R-squared: 0.8268231

Continuous Kernel Type: Second-Order Gaussian
No. Continuous Explanatory Vars.: 1

>plot(iris$Petal.Length,iris$Sepal.Length)
>lines(iris$Petal.Length,predict(fit_7))
```



Regression Models

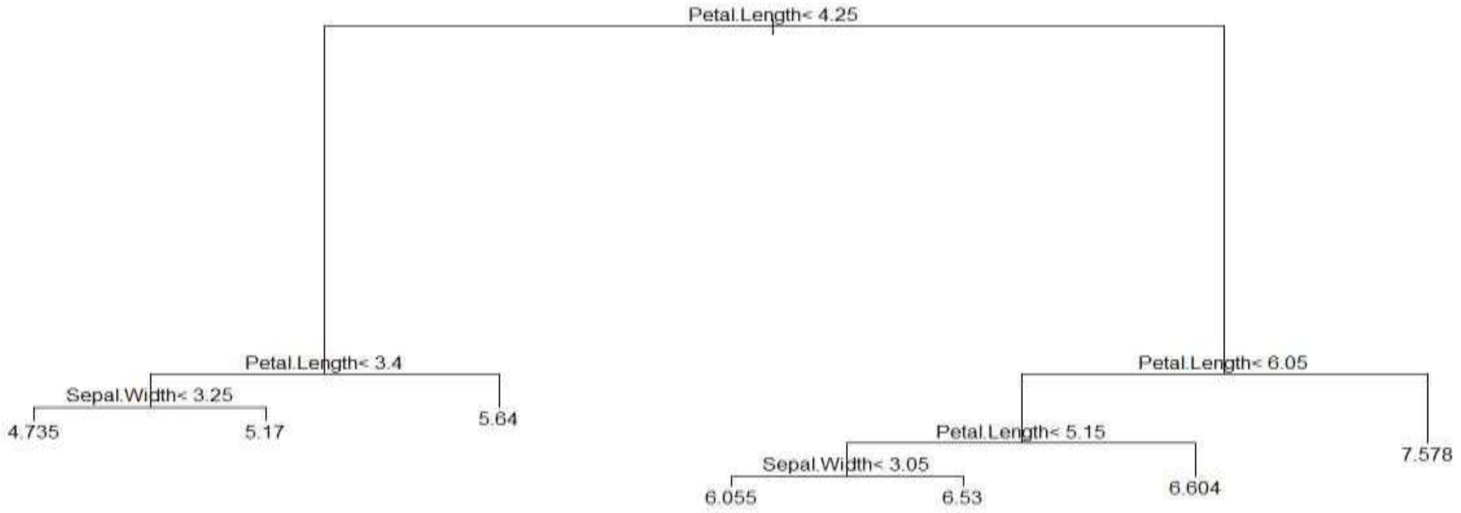
- Parametric Regression Models
  - Simple Linear Regression
  - Multivariate Linear Regression
  - Log-Linear Regression
- Nonparametric Regression Models
  - Locally Weighted Regression
  - Kernel Regression
  - Regression Trees

Regression Trees

- Decision trees are one of the most widely used models in all of machine learning and data mining. A tree is a data structure where we have a root node at the top and a set of nodes as its children. The child nodes can also have their own children, or be terminal nodes in which case they are called leaf nodes. A tree has a recursive structure as any of its node is the root of a subtree comprising the node's children.

Regression Trees

```
> library(rpart)
> fit_8<-rpart(Sepal.Length~Sepal.Width+Petal.Length+Petal.Width,data=iris)
> plot(fit_8);text(fit_8)
```

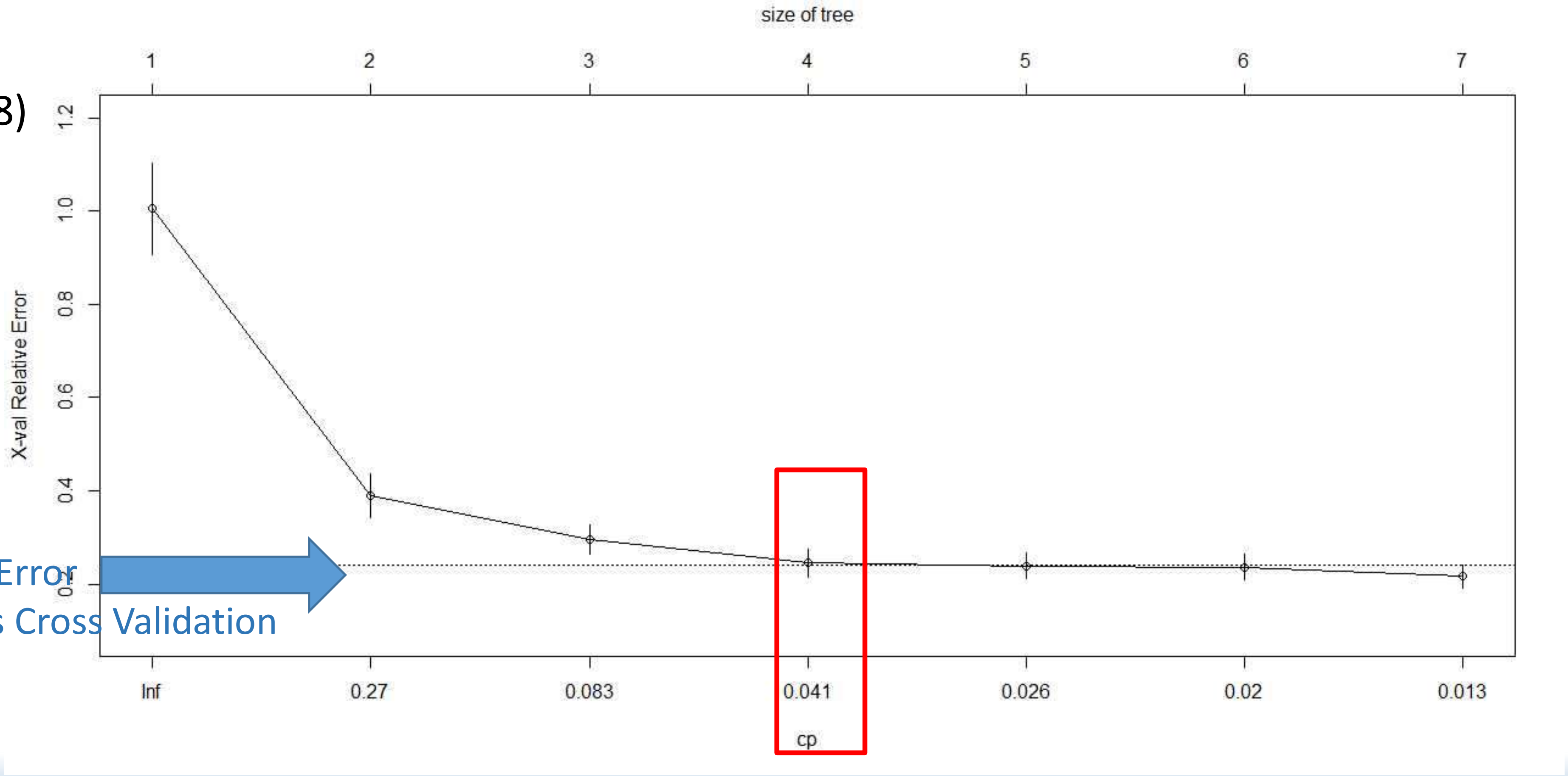


Regression Trees – Prune Tree

- Pruning is a technique in machine learning that reduces the size of decision trees by removing sections of the tree that provide little power to classify instances. Pruning reduces the complexity of the final classifier, and hence improves predictive accuracy by the reduction of overfitting.

Regression Trees – Prune Tree

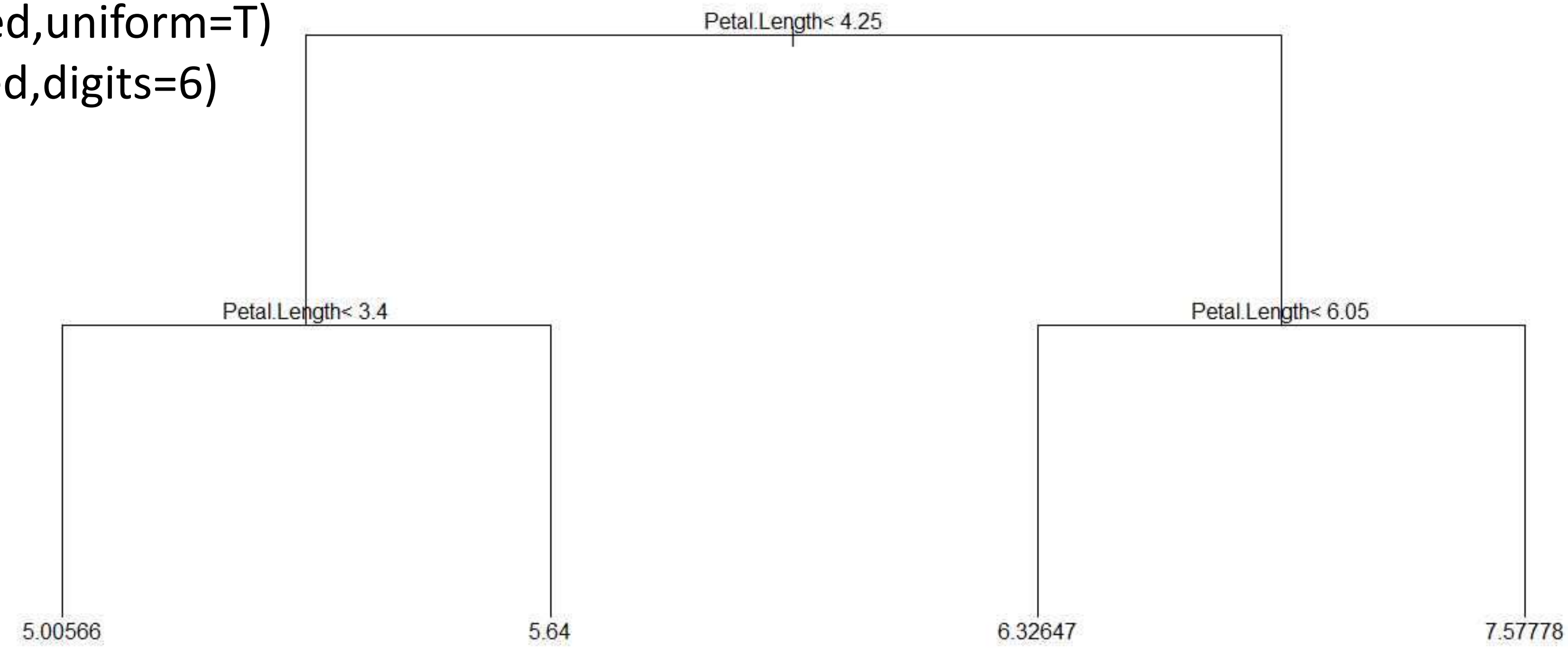
>plotcp(fit_8)



The smallest Error
With 10 Folds Cross Validation

Regression Trees – Prune Tree

```
fit_8_pruned<-prune(fit_8,cp=.041)
plot(fit_8_pruned,uniform=T)
text(fit_8_pruned,digits=6)
```



Homework 6 (submitted to e3.nctu.edu.tw before Oct 29, 2018)

- Use R and/or the other software to perform regression analyze the data set that you select
- Explain the results you obtain
- Discuss possible problems you plan to investigate for future studies
- Possible source of open data:

  UCI Machine Learning Repository

  (http://archive.ics.uci.edu/ml/datasets.html)

References

1. Ch. 6, M. A. Pathak, Beginning Data Science with R, 2014, Springer-Verlag.
2. http://mezeylab.cb.bscb.cornell.edu/labmembers/documents/supplement%205%20-%20multiple%20regression.pdf
3. https://en.wikipedia.org/wiki/Nonparametric_statistics
4. https://en.wikipedia.org/wiki/Local_regression#Definition_of_a_LOESS_model
5. https://en.wikipedia.org/wiki/Kernel_(statistics)#cite_note-1
6. https://www.rdocumentation.org/packages/np/versions/0.60-6/topics/npreg
7. https://www.statmethods.net/advstats/cart.html
8. http://scg.sdsu.edu/ctrees_r/