Data Analytics

Course Taught at IIFT

Day 10: Logistic and Nonlinear Regression Analysis

Dr. Tanujit Chakraborty

Centre for Data Sciences

IIIT Bangalore

BINARY LOGISTIC REGRESSION

BINARY LOGISTIC REGRESSION

- It is a multiple regression with an outcome variable (or dependent variable) to be a categorical dichotomic and explanatory variables that can be either continuous or categorical.

- In other words, the interest is in predicting which of two possible events are going to happen given certain other information.

- For example in Drug efficacy testing, logistic regression could be used to analyze the factors that determine whether the drug cures a particular disease or not.

- The Logistic Curve will relate the explanatory variable X to the probability of the event occurring.

BINARY LOGISTIC REGRESSION

Used to develop models when the output or response variable y is binary

The output variable will be binary, coded as either success or failure

Models probability of success p which lies between 0 and 1

Linear model is not appropriate

$$p = \frac{e^{a+b_1x_1+b_2x_2+---+b_kx_k}}{1+e^{a+b_1x_1+b_2x_2+---+b_kx_k}}$$

p: probability of success
x_i's : independent variables
a, b_1, b_2, ---: coefficients to be estimated

If estimate of $p \geq 0.5$, then classified as success, otherwise as failure

BINARY LOGISTIC REGRESSION

Usage: When the dependant variable (Y variable) is binary

Example: Develop a model to predict the number of visits of family to a vacation resort based on the salient characteristics of the families. The data collected from 30 households is given in Resort_Visit.csv

1. Reading the file and variables

```
> mydata =  read.csv('Resort_Visit.csv',header = T,sep = ",")
> visit = mydata$Resort_Visit
> income = mydata$Family_Income
> attitude = mydata$Attitude.Towards.Travel
> importance = mydata$Importance_Vacation
> size = mydata$House_Size
> age = mydata$Age._Head
```

2. Converting response variable to discrete

```
> visit = factor(visit)
```

BINARY LOGISTIC REGRESSION

3. Correlation Matrix

> cor(mydata)

	Resort_Visit	Family_Income	Attitude_Travel	Importance_Vacation	House_Size	Age_Head
Resort_Visit	1.00	-0.60	-0.27	-0.42	-0.59	-0.21
Family_Income	-0.60	1.00	0.30	0.23	0.47	0.21
Attitude_Travel	-0.27	0.30	1.00	0.19	0.15	-0.13
Importance_Vacation	-0.42	0.23	0.19	1.00	0.30	0.11
House_Size	-0.59	0.47	0.15	0.30	1.00	0.09
Age_Head	-0.21	0.21	-0.13	0.11	0.09	1.00

Interpretation: Correlation between X variables should be low

BINARY LOGISTIC REGRESSION

4. Checking relation between Xs and Y

> aggregate(income ~visit, FUN = mean)

> aggregate(attitude ~visit, FUN = mean)

> aggregate(importance ~visit, FUN = mean)

> aggregate(size ~visit, FUN = mean)

> aggregate(age ~visit, FUN = mean)

Resort_Visit	Mean				
	Family_Income	Attitude_Travel	Importance_Vacation	House_Size	Age_Head
0	58.5200	5.4000	5.8000	4.3333	53.7333
1	41.9133	4.3333	4.0667	2.8000	50.1333

Higher the difference in means, stronger will be the relation to response variable.

BINARY LOGISTIC REGRESSION

5. Checking relation between Xs and Y – box plot

> boxplot(income ~ visit)

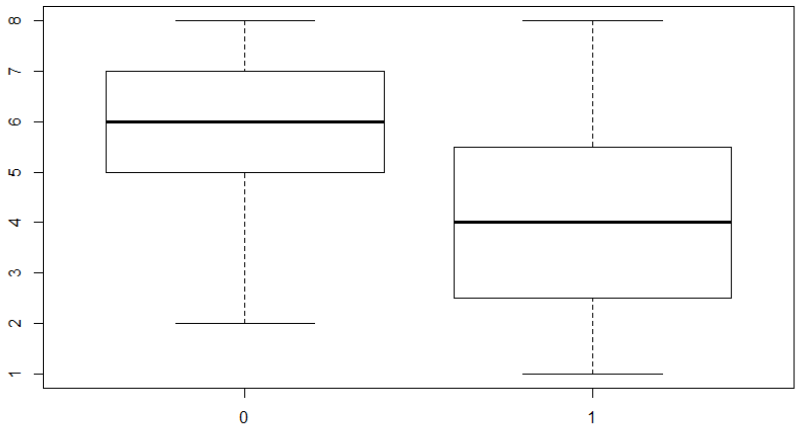
> boxplot(attitude ~ visit)

> boxplot(importance ~ visit)

> boxplot(size ~ visit)

> boxplot(age ~ visit)

Income Vs visit



BINARY LOGISTIC REGRESSION

6. Perform Logistic regression

> model = glm(visit ~ income + attitude + importance + size + age, family = binomial(logit))

> summary(model)

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	15.49503	6.68017	2.32	0.0204
Income	-0.11698	0.06605	-1.771	0.0766
attitude	-0.28129	0.33919	-0.829	0.4069
importance	-0.46157	0.32006	-1.442	0.1493
size	-0.80699	0.49314	-1.636	0.1018
age	-0.07019	0.07199	-0.975	0.3295

BINARY LOGISTIC REGRESSION

6. Perform Logistic regression - Anova

> anova(model,test = 'Chisq')

	Df	Deviance	Resid.Df	Resid.Dev	Pr(>Chi)
NULL	29	41.589			
income	1	12.9813	28	28.608	0.00031
attitude	1	0.4219	27	28.186	0.51598
importance	1	3.8344	26	24.351	0.05021
size	1	3.4398	25	20.911	0.06364
age	1	1.0242	24	19.887	0.31152

Since p value < 0.05 for Income redo the modelling with important factor (income) only.

BINARY LOGISTIC REGRESSION

7. Perform Logistic regression - Modified

	Estimate	Std Error	z value	p value
(Intercept)	6.36727	2.32544	2.738	0.00618
Income	-0.12778	0.04634	-2.758	0.00582

Since p value < 0.05 for Income, the response variable can be modelled in terms of those two factors

The logistic model is:

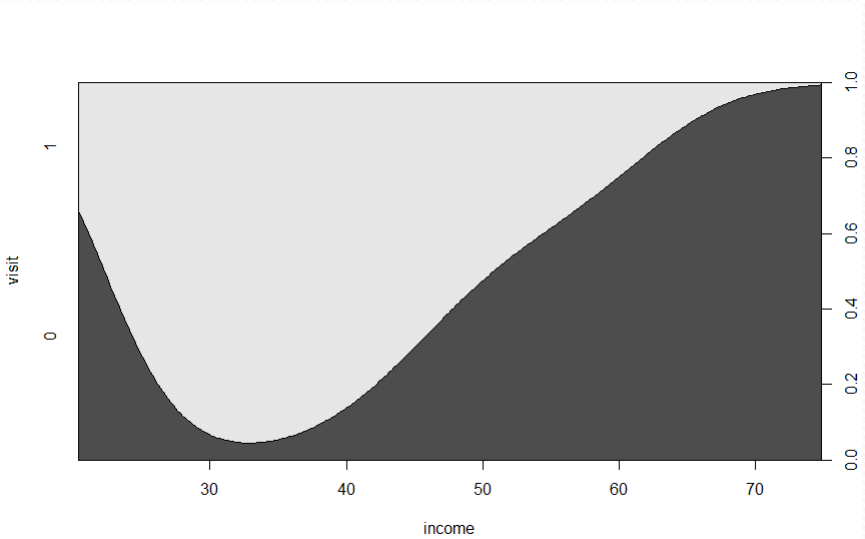
$$y = \frac{e^{6.36727\ -0.12778 * Annual\_Income}}{1 + e^{6.36727\ -0.12778 * Annual\_Income}}$$

BINARY LOGISTIC REGRESSION

8. Conditional Density plots (Response Vs Factors)

Describing how the conditional distribution of a categorical variable y changes over a numerical variable x

> cdplot(visit ~ income)



BINARY LOGISTIC REGRESSION

9. Fitted Values and residuals

> predict(model,type = 'response')

> residuals(model,type = 'deviance')

> predclass = ifelse(predict(model, type ='response')>0.5,"1","0")

SL No.	Actual	Fitted	Residuals	Predicted Class	SL No.	Actual	Fitted	Residuals	Predicted Class
1	0	0.970979	-2.66073	1	16	1	0.904132	0.448954	1
2	0	0.059732	-0.35097	0	17	1	0.939523	0.353222	1
3	0	0.021049	-0.20627	0	18	1	0.880611	0.50426	1
4	0	0.202309	-0.67236	0	19	1	0.345537	1.457845	0
5	0	0.292461	-0.83182	0	20	1	0.724535	0.802777	1
6	0	0.014893	-0.17324	0	21	1	0.925508	0.393479	1
7	0	0.677783	-1.50501	1	22	1	0.677559	0.882337	1
8	0	0.038723	-0.28105	0	23	1	0.680103	0.878079	1
9	0	0.109432	-0.48145	0	24	1	0.516151	1.150092	1
10	0	0.030543	-0.24908	0	25	1	0.680326	0.877704	1
11	0	0.017609	-0.1885	0	26	1	0.77062	0.721887	1
12	0	0.050856	-0.32309	0	27	1	0.629425	0.962235	1
13	0	0.04202	-0.29301	0	28	1	0.954395	0.305541	1
14	0	0.601981	-1.35739	1	29	1	0.841493	0.587498	1
15	0	0.499424	-1.17643	0	30	1	0.900286	0.45835	1

BINARY LOGISTIC REGRESSION

10. Model Evaluation

```
> mytable = table(visit, predclass)
> mytable
> prop.table(mytable)
```

	Predicted Count		Total
Actual Count	0	1	
0	11	4	15
1	3	12	15
Total	14	16	30

Statistics	Value
Accuracy %	76.666
Error %	23.333

Accuracy of ≥ 75 % is considerably good.

ORDINAL LOGISTIC REGRESSION

ORDINAL LOGISTIC REGRESSION

Used to develop models when the output or response variable y is ordinal.

The output variable will be categorical, having more than two categories.

ORDINAL LOGISTIC REGRESSION

Example 1: The data on system test defect density along with testing effort and test coverage is given in ST_Defects.csv. The defect density is classified as Low Medium High. Develop a model to estimate the system testing defect density class based on testing effort and test coverage ?

Read the data file and variables

```
> mydata = read.csv('ST_Defects.csv', header = T, sep = ",")
> dd = mydata$DD
> effort = mydata$Effort
```

- coverage = mydata$Test.Coverage
- dd = factor(dd)

ORDINAL LOGISTIC REGRESSION

Example 1: The data on system test defect density along with testing effort and test coverage is given in ST_Defects.csv. The defect density is classified as Low Medium High. Develop a model to estimate the system testing defect density class based on testing effort and test coverage ?

Make one of the classes (say “Low”) of output variable as the baseline level

```
> library(MASS)
> mymodel = polr(dd ~ effort + coverage)
> summary(mymodel)
```

ORDINAL LOGISTIC REGRESSION

Example 1: The data on system test defect density along with testing effort and test coverage is given in ST_Defects.csv. The defect density is classified as Low Medium High. Develop a model to estimate the system testing defect density class based on testing effort and test coverage ?

Coefficients

effort	coverage
0.0234	0.0257

Intercepts

High | Low	Low | Medium
1.4947	3.925

ORDINAL LOGISTIC REGRESSION

Example 1: The data on system test defect density along with testing effort and test coverage is given in ST_Defects.csv. The defect density is classified as Low Medium High. Develop a model to estimate the system testing defect density class based on testing effort and test coverage ?

Predicted values

```
> pred = predict(mymodel)
> fit = fitted(mymodel)
>fit
> output = cbind(dd, pred)
>write.csv(output, "E:/Part 2/output.csv")
```

ORDINAL LOGISTIC REGRESSION

Example 1: The data on system test defect density along with testing effort and test coverage is given in ST_Defects.csv. The defect density is classified as Low Medium High. Develop a model to estimate the system testing defect density class based on testing effort and test coverage ?

Comparing Actual Vs Predicted

```
> mytable = table(dd, pred)
> mytable
> prop.table(mytable)
```

		Predicted		
Actual		High	Low	Medium
	High	8	42	0
	Low	0	105	0
	Medium	1	44	0

ORDINAL LOGISTIC REGRESSION

Example 1: The data on system test defect density along with testing effort and test coverage is given in ST_Defects.csv. The defect density is classified as Low Medium High. Develop a model to estimate the system testing defect density class based on testing effort and test coverage ?

Comparing Actual Vs Predicted (in %)

		Predicted		
Actual		High	Low	Medium
	High	4.0	21.0	0.00
	Low	0.00	52.50	0.00
	Medium	0.50	22.0	0.00

Accuracy = 4 + 52.5 + 0.00 = 0.565 = 56.5%

MODELING NONLINEAR RELATIONS

MODELING NONLINEAR RELATIONS

The linear regression is fast and powerful tool to model complex phenomena.

But makes several assumptions about the data including the assumption of linear relationship exists between predictors and response variable.

When these assumptions are violated, the model breaks down quickly.

MODELING NONLINEAR RELATIONS

The linear model $y = x\beta + \varepsilon$ is general model

Can be used to fit any relationship that is linear in the unknown parameter β

Examples:

$$y = \beta_0 + \beta_1 x_1 + --- + \beta_k x_k + \varepsilon$$

$$y = \beta_0 + \beta_1 x_1 + \beta_{11} x_1^2 + \varepsilon$$

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 + \varepsilon$$

In general

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

where f(x) can be 1/x, $\sqrt{x}$, log(x), e^x , etc

MODELING NONLINEAR RELATIONS

Detection of non linear relation between predictor x and response variable y

Scatter Plot:

The plotted points are not lying lie in a straight line is an indication of non linear relationship between predictor and dependant variable

Component Residual Plots:

An extension of partial residual plots

Partial residual plots are the plots of residuals of one predictor against dependant variable

Component residual plots(crplots) adds a line indicating where the best fit line lies.

A significant difference between the residual line and the component line indicate that the predictor does not have a linear relationship wit the dependent variable

MODELING NONLINEAR RELATIONS

Example : The data given in Nonlinear_Thrust.csv represent the thrust of a jet turbine engine (y) and 3 predictor variables: x_1 = fuel flow rate, x_2 = pressure, and x_3 = exhaust temperature. Develop a suitable model for thrust in terms of the predictor variables.

Read Data

```
> attach(mydata)
> cor(mydata)
```

	x1	x2	x3	y
x1	1.00	0.40	-0.20	0.54
x2	0.40	1.00	-0.30	-0.36
x3	-0.20	-0.30	1.00	0.35
y	0.54	-0.36	0.35	1.00

There is no strong correlation between y and x's

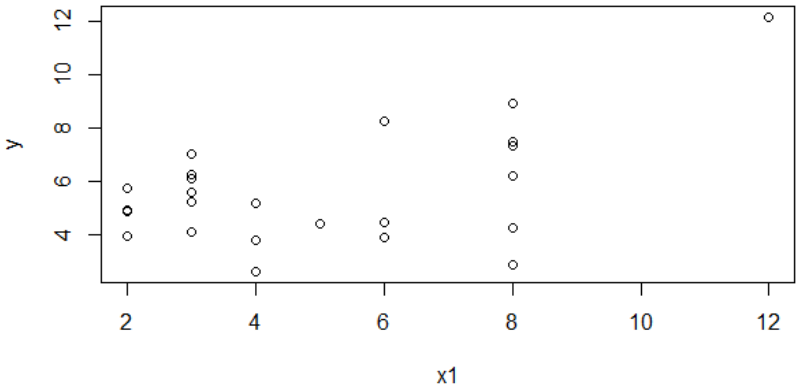
MODELING NONLINEAR RELATIONS

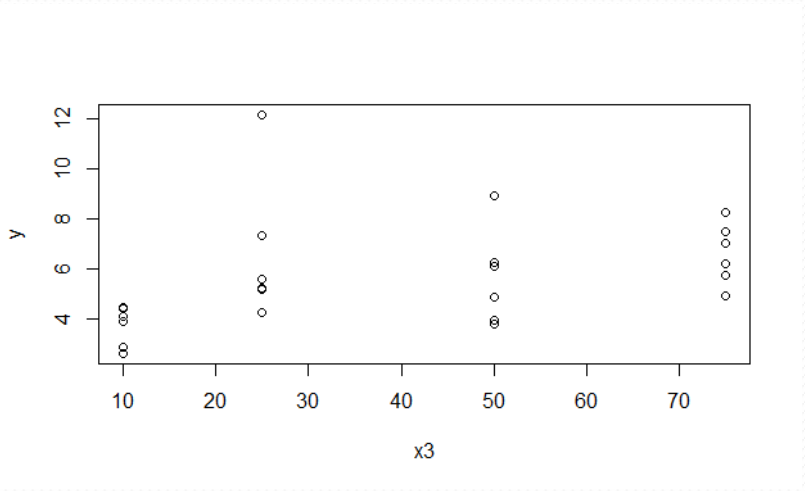
Draw Scatter plots

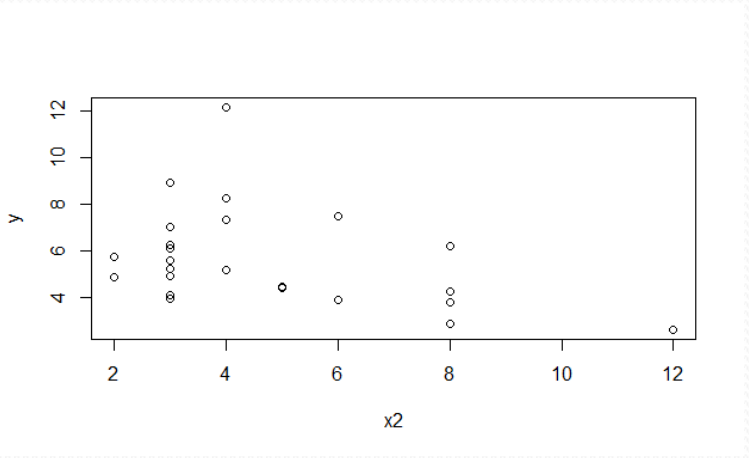
> plot(x1,y)

> plot(x2,y)

> plot(x3,y)







There is no strong correlation between y and x's

MODELING NONLINEAR RELATIONS

Develop the model

```
> mymodel = lm(y ~ x1 + x2 + x3, data = mydata)
> summary(mymodel)
```

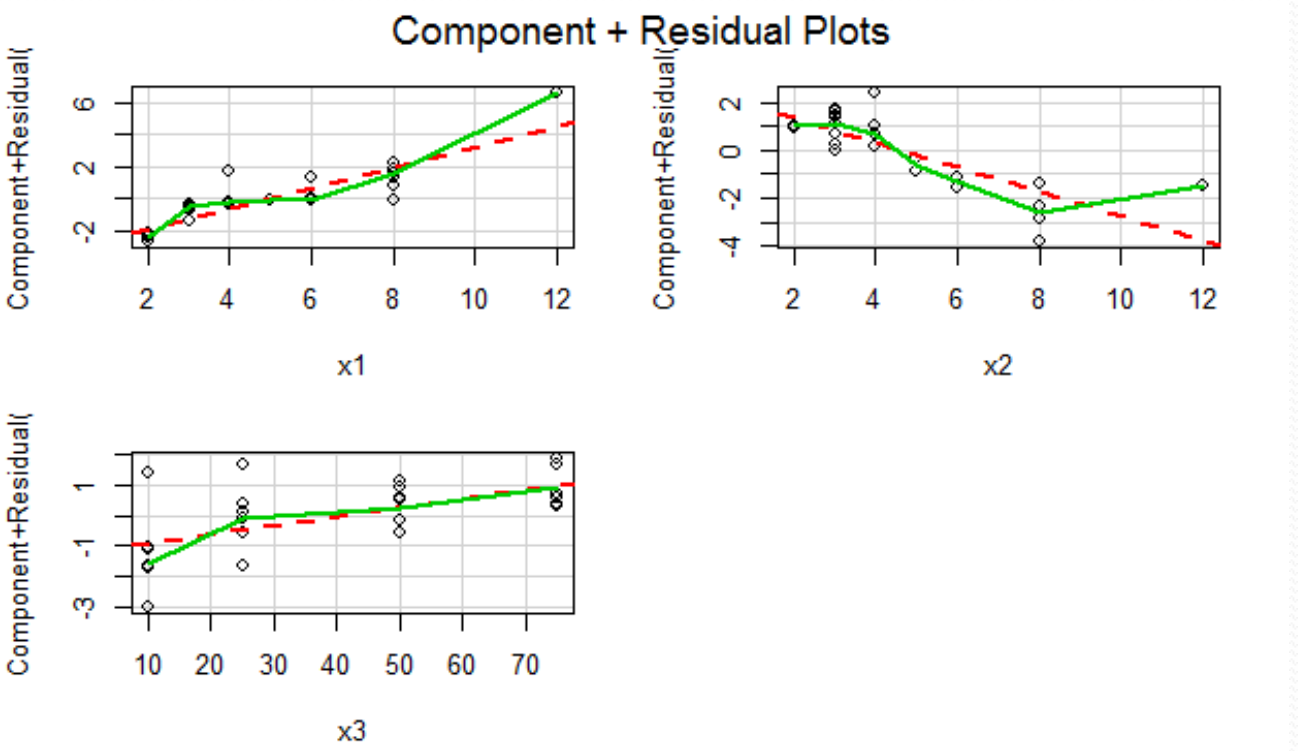
	Estimate	Std. Error	t	p value
(Intercept)	3.58315	0.726839	4.93	0.0001
x1	0.651547	0.0855	7.62	0.0000
x2	-0.509866	0.097132	-5.249	0.0000
x3	0.028888	0.009021	3.202	0.00428

R^2	0.786
Adjusted R^2	0.7563

MODELING NONLINEAR RELATIONS

Develop the model

```
> library(car)
> crPlots(mymodel)
```

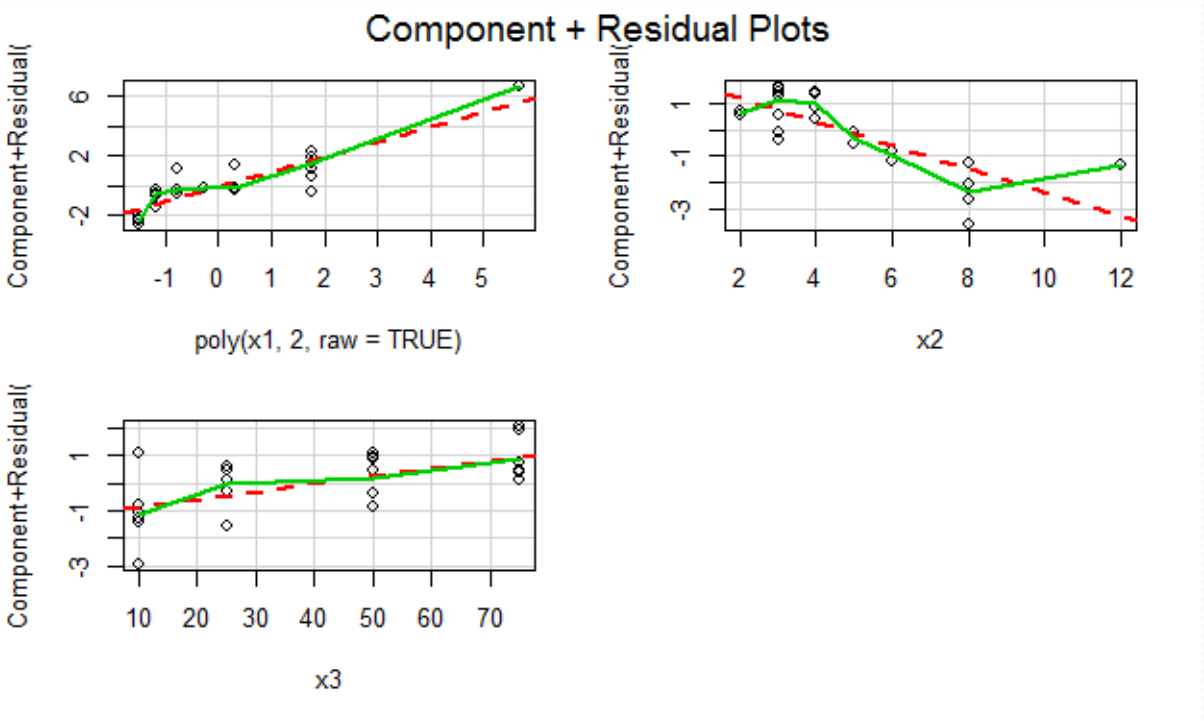


Since the best fit line different from residual line, it is possible improve the model by adding higher order terms

MODELING NONLINEAR RELATIONS

Develop the model

```
> mymodel = lm(y ~ poly(x1, 2, raw = TRUE) + x2 + x3, data = mydata)
> crPlots(mymodel)
```

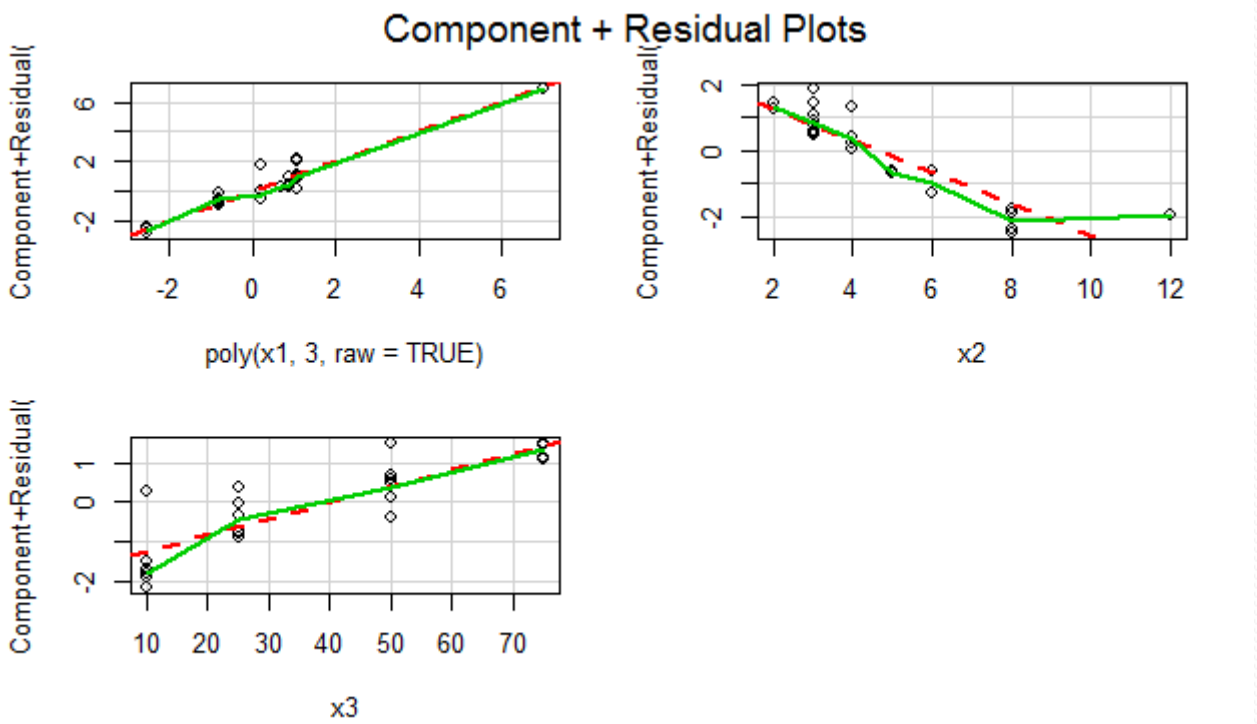


Since the best fit line different from residual line, it is possible improve the model by adding higher order terms

MODELING NONLINEAR RELATIONS

Develop the model

```
> mymodel = lm(y ~ poly(x1, 3, raw = TRUE) + x2 + x3, data = mydata)
> crPlots(mymodel)
```

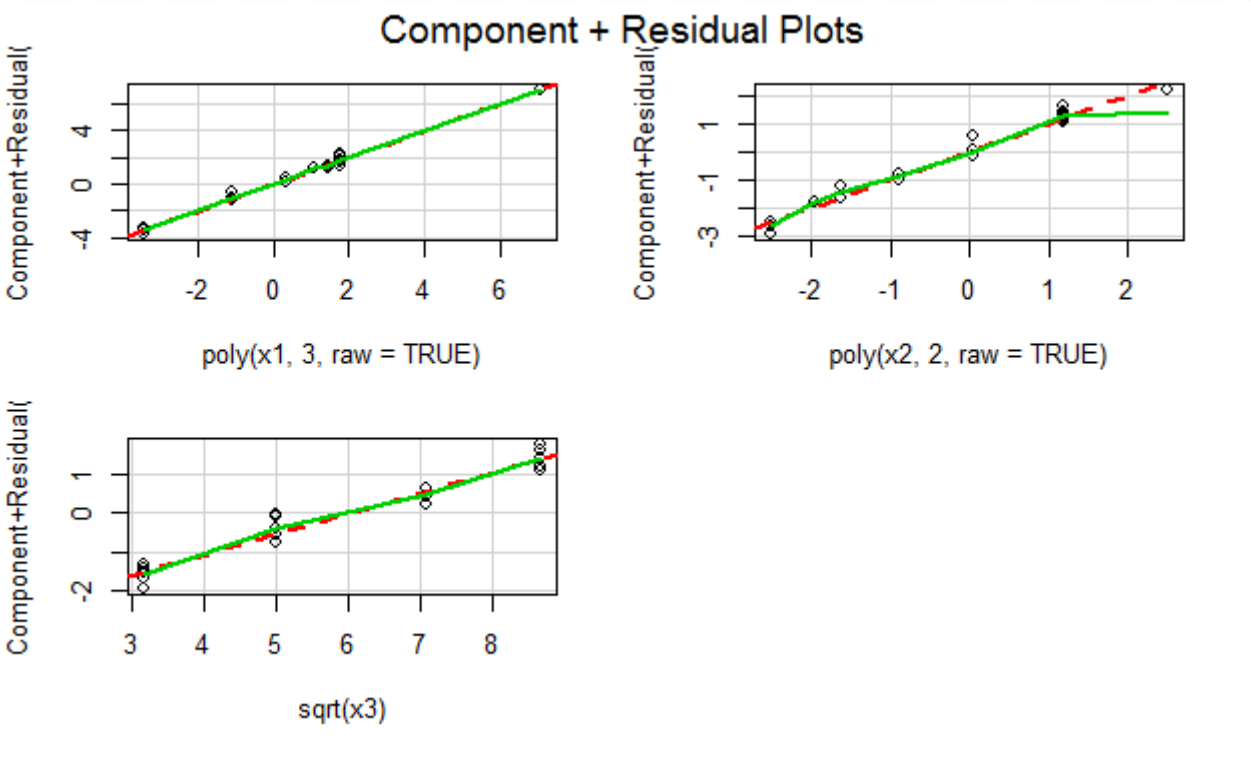


Since the best fit line is more or less overlapping residual line, hence adding square and cube terms of x1 will improve the model. Similarly add additional terms or functions of x2 and x3 to improve the model

MODELING NONLINEAR RELATIONS

Develop the model: Final Model

```
> mymodel = lm(y ~ poly(x1, 3, raw = TRUE) + poly(x2, 2, raw = TRUE) + sqrt(x3), data = mydata)
> crPlots(mymodel)
```



MODELING NONLINEAR RELATIONS

Develop the model: Final Model

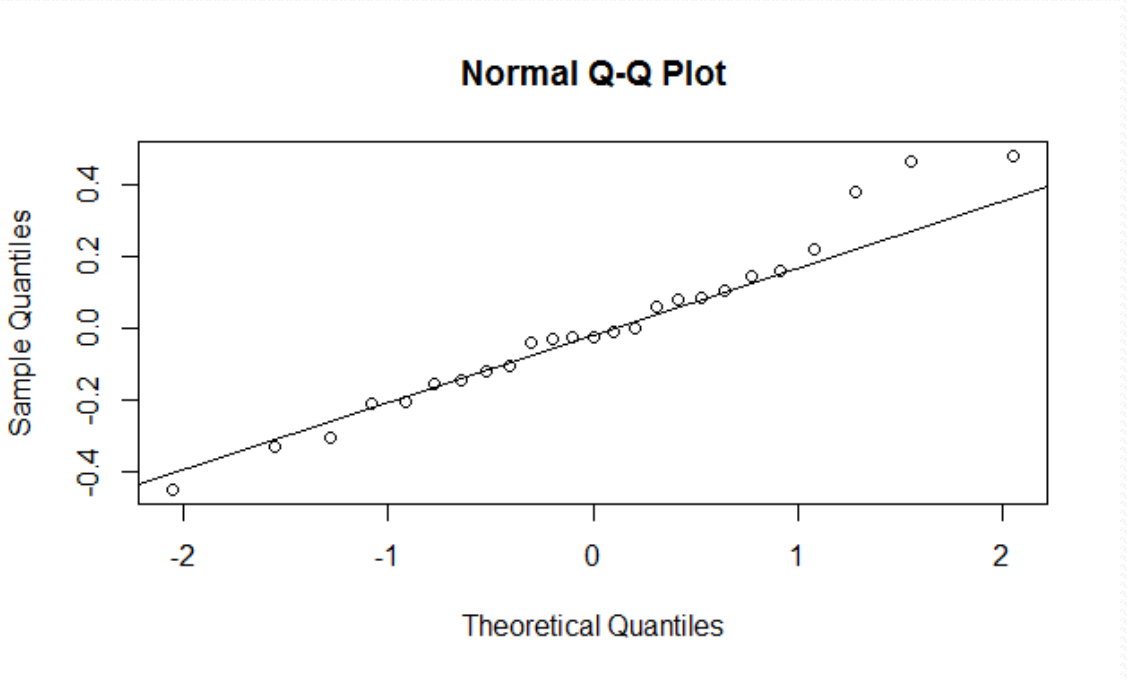
	Estimate	Std. Error	t	p value
(Intercept)	-3.48301	0.705793	-4.935	0.000107
x_1	5.503467	0.36278	15.17	0.0000
x_1^2	-0.77878	0.056814	-13.708	0.0000
x_1^3	0.037516	0.002685	13.971	0.0000
x_2	-1.81437	0.146304	-12.401	0.0000
x_2^2	0.097886	0.010374	9.435	0.0000
$\sqrt{x_3}$	0.527417	0.030664	17.2	0.0000

R^2	0.9881
Adjusted R^2	0.9841

MODELING NONLINEAR RELATIONS

Develop the model: Final Model

```
> res = residuals(mymodel)
> qqnorm(res)
> qqline(res)
> shapiro.test(res)
```



Shapiro test for Normality	
w	0.9704
p value	0.6569

REGRESSION SPLINES

REGRESSION SPLINES

Spline

A continuous function formed by connecting linear segments

A function constructed piecewise from polynomial functions

Knots

The points where the segments are connected

Spline of degree D

A function formed by connecting polynomial segments of degree D so that

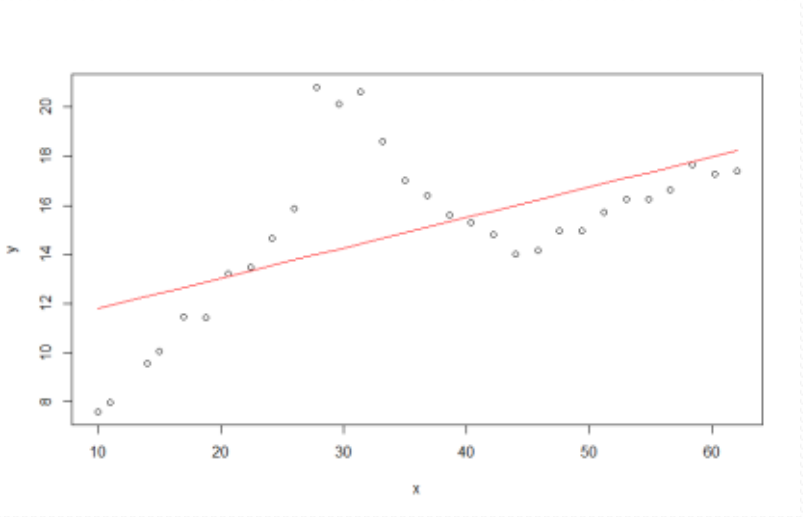
- Function is continuous
- Function has D – 1 continuous derivatives

Usage

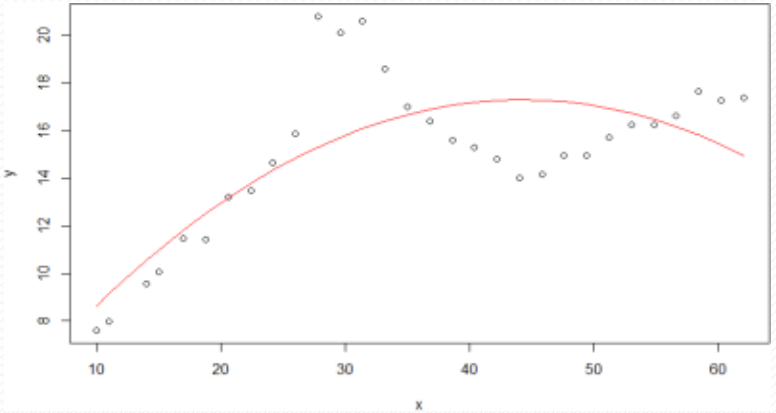
Develop models when relationship between y and x's is piecewise polynomial

REGRESSION SPLINES

y vs x (linear)

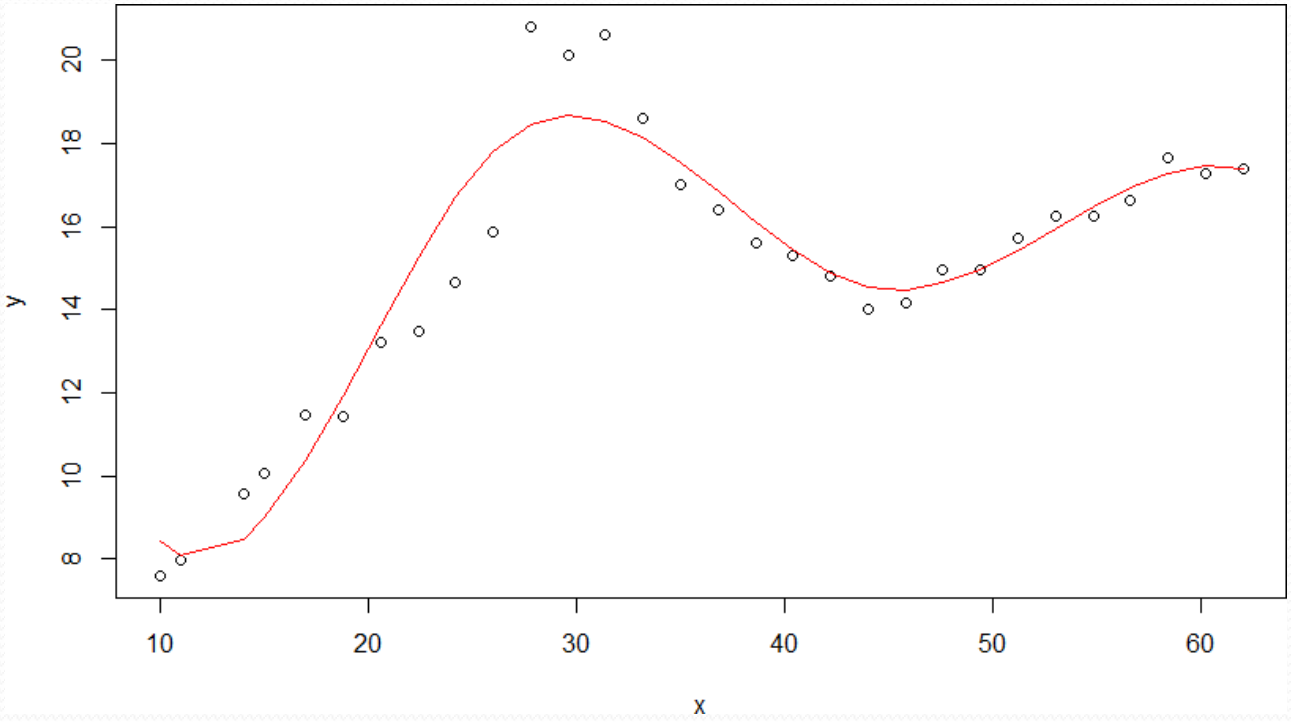


y vs x (Polynomial)



REGRESSION SPLINES

y vs x (Piecewise polynomial - Spline)



REGRESSION SPLINES

Example 1: The data on defect finding rate (design phase) and the corresponding defect finding rate (coding phase) of 20 similar projects is given in Reg_Spline_DFR.csv. Fit a suitable model to predict defect finding rate in coding phase in terms of defect finding rate in design phase?

Reading data

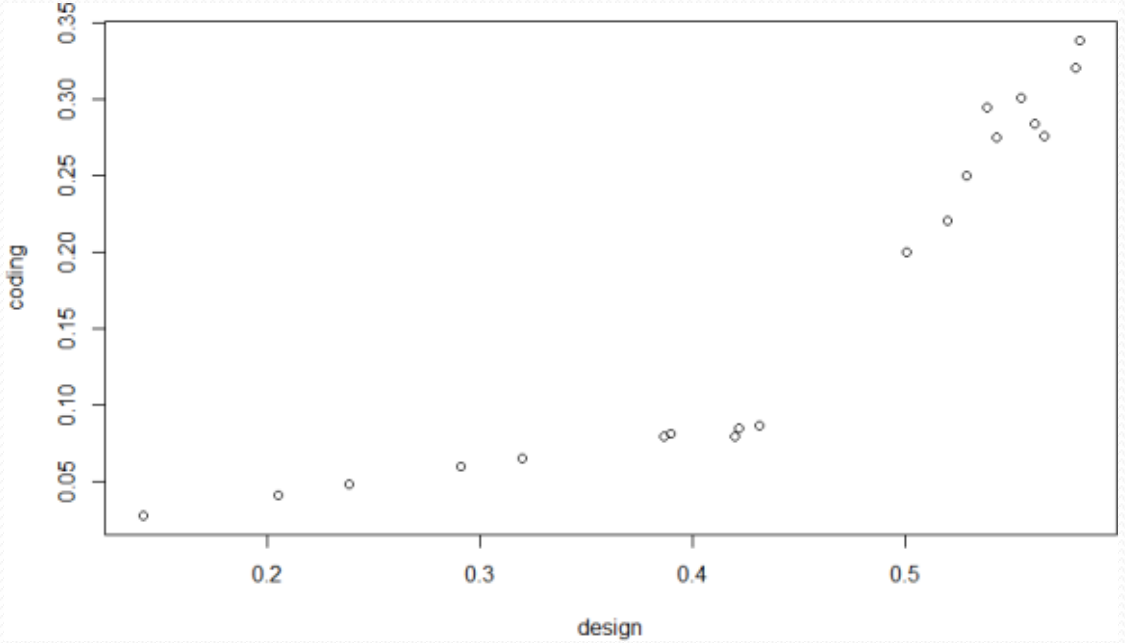
```
> design = mydata$Design
> coding = mydata$Coding
> plot(design, coding)
```

REGRESSION SPLINES

Example 1:

Exploring the relationship

```
> plot(design, coding)
```



REGRESSION SPLINES

Example 1:

Fitting a linear model

```
> mymodel = lm(coding ~ design)
> summary(mymodel)
```

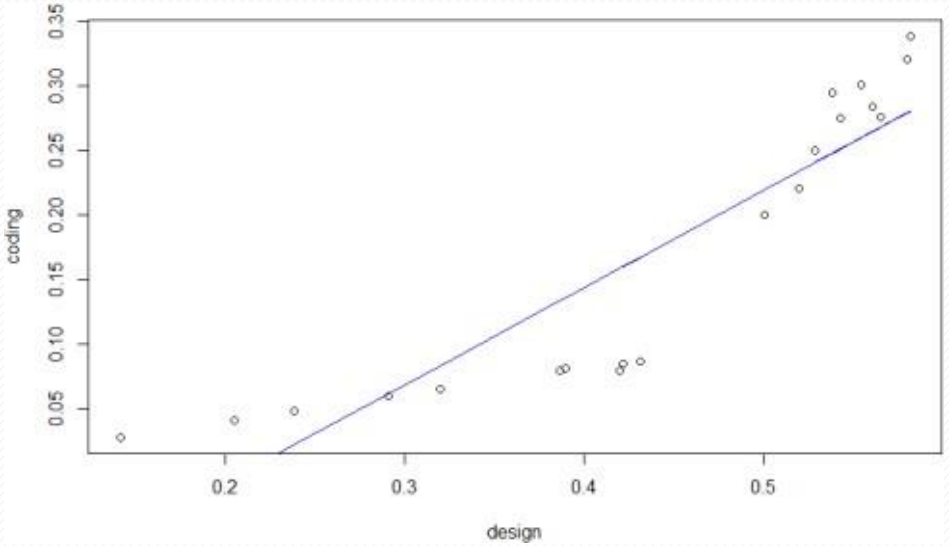
Statistics	Value
R^2	0.7862
R^2 adjusted	0.7744
F Statistics	66.21
P value	0.0000

REGRESSION SPLINES

Example 1:

Plotting the model

```
> pred = predict(mymodel)
> plot(design, coding)
> lines( design, pred, col ="blue")
```



REGRESSION SPLINES

Example 1:

Introducing knot at design = 0.44

```
> design44 = design - 0.44
> design44[design44 < 0] = 0
```

Fitting linear spline model

```
> mymodel = lm(coding ~ design + design44)
> summary(mymodel)
```

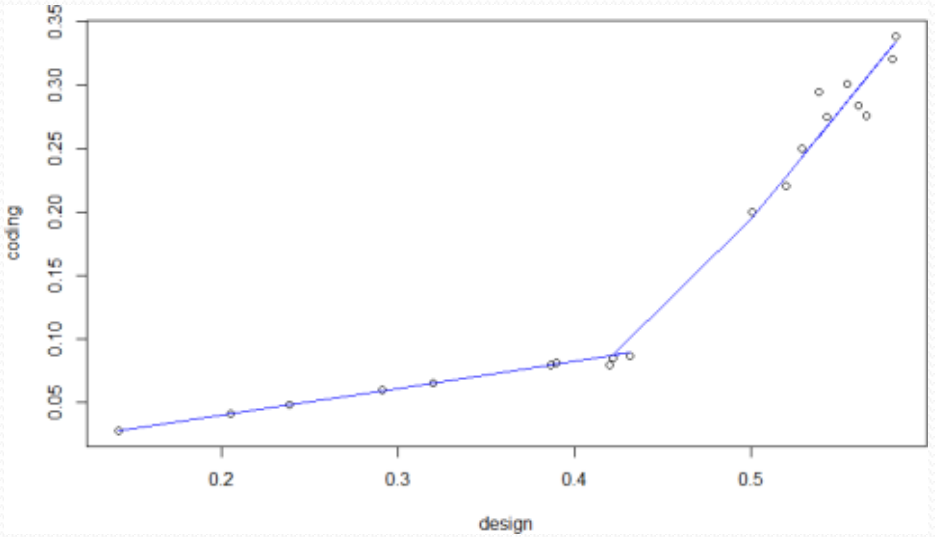
Statistics	Value
R^2	0.9823
R^2 adjusted	0.9802
F Statistics	472.2
P value	0.000

REGRESSION SPLINES

Example 1:

Plotting the linear spline model

```
> pred = predict(mymodel)
> plot(design, coding)
> lines(design, pred, col ="blue")
```



Note: Model is good but not a continuous function

REGRESSION SPLINES

Example 1:

Fitting cubic spline model

```
> designsq = design^2
> designcb = design^3
> design44cb = design44^3
> mymodel = lm(coding ~ poly(design, 3, raw =TRUE) + design44cb)
> summary(mymodel)
```

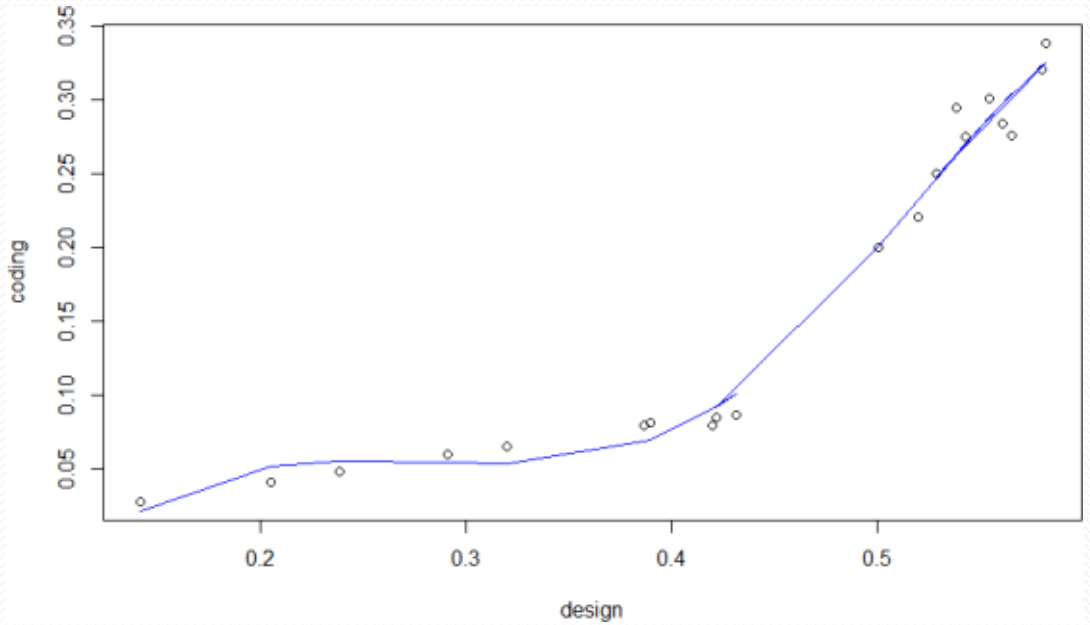
Statistics	Value
R^2	0.9782
R^2 adjusted	0.9724
F Statistics	168.5
P value	0.000

REGRESSION SPLINES

Example 1:

Plotting the linear spline model

```
> pred = predict(mymodel)
> plot(design, coding)
> lines(design, pred, col ="blue")
```



Homework: EXPLORE Multivariate Adaptive Regression Splines (**MARS)**

CHEAT SHEET

Dependent Variable Type (Ys)	Independent Variable Type (Xs)	Modelling Technique
Numerical	Numerical	1. Linear Regression or Best Subset Regression 2. Non-linear Regression or Regression Splines 3. Regression Trees, Neural Nets, etc.
Numerical	Categorical + Numerical	1. Linear Regression with Dummy Variables 2. Polynomial Regression with Dummy Variables 3. Regression Trees, Neural Nets, etc.
Categorical	Numerical	1. Logistics Regression 2. Classification Trees 3. Support Vector Machines, Neural Nets, etc.
Categorical	Categorical + Numerical	1. Logistic Regression with Dummy Variables 2. Classification Trees 3. Advanced Neural Nets, etc.
Numerical (Time dependent)	Numerical Exogenous Variables	1. ARIMA, ETS, Naïve Model 2. Autoregressive Neural Network 3. RNN, LSTM, etc.

Reference

The detail material related to this lecture can be found in

- ISLR Book : https://www.statlearning.com/

Mid-Term Examination

- There will be NO alternative exam for Mid-Term. DO NOT miss the exam.
- Answer ALL the questions. Each question carries equal marks.
- There is NO NEGATIVE marking and NO PART marking.
- For a question, there is ONLY ONE correct answer.
- Exam is of MCQ type (15 questions and 30 marks).