Grading

Assessment	Individual Homework Assignments	32% (=4*8%)
	Exam 1	28%
	Exam 2	28%
	Group Project Presentation	10%
	Group Project Participation	2%

Grading

- **10% Group Project**

2%	4%	4%
Data Exploration & Visualization	***[Classification] Decision Tree***	***[Classification] Logistic Regression***
(1) Each Variables Statistics and Specification	*(1) Splitting the data into training and test data*	*(1) Splitting the data into training and test data*
(2) Outlier Detection by using Box Plot and Pre-processing	*(2) Growing a tree and display basic results*	*(2) Logistic regression and display the results*
	(3) Plotting a tree and make an interpretation	*(3) Interpretation of significancy and coefficient*
	(4) Accuracy on the training & test data	*(4) Accuracy on the training & test data*

(counter-intuitive interesting finding)

Grading

- **Group Project Presentation 10% + [Additional Points]**

- **An additional 2 points will be awarded to 2–3 groups that deliver top-quality presentations.**

- **An additional 1 point will be awarded to 2–3 groups that demonstrate mid-level quality.**

- **No extra points will be given to the remaining teams.**

Data Exploration & Visualization

(1) Each Variables Statistics and Specification

 # have a look at the structure of the dataset

> str(iris)

```
> 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 ...
```

- 150 observations (i.e., rows) and 5 variables (i.e., attributes or columns)
- The first four variables are **numeric.**
- The last variable, Species, is **categoric** (**called “factor” in R**) and has **three levels** of values.

Data Exploration & Visualization

(1) Each Variables Statistics and Specification

Table 1. Variable Definitions and Summary Statistics

Variable	Definition	Mean	Std. Dev.	Min	Max
CompletionRate (%)	Extent (ratio) to which a consumer reads e-book content before writing a review	75.921	32.641	0	100
Reviewing	A binary variable indicating whether a customer writes a review after the e-book consumption (no = 0, yes = 1)	0.075	0.264	0	1
ReviewValence	Numerical review ratings given by each consumer	4.527	0.957	1	5
ReviewLength	Number of words in a review given by each consumer	8.411	23.793	1	798
ReviewExtremity	The deviation from the mean value of total reviews	0.708	0.644	0.467	3.533
Gender	A binary variable indicating gender of consumers (Female = 0, Male = 1)	0.050	0.218	0	1
Age	Consumer's age	41.015	8.969	17	98
RegistrationDate	The number of days elapsed since a consumer registered on the platform to July 19, 2016 (cutoff day)	857.518	568.438	53	2,870
Price (USD)	Fixed price of a book	4.312	2.910	0.182	73.18
PublicationDate	The number of days elapsed since an e-book was published to July 19, 2016 (cutoff day)	361.077	392.844	52	2,624
Adult	A binary variable indicating whether an e-book belongs to an adult category	0.870	0.336	0	1

Note. Std. Dev., standard deviation.

Data Exploration & Visualization

(1) Each Variables Statistics and Specification

DV

IVs

Table 1. Variable Definitions and Summary Statistics

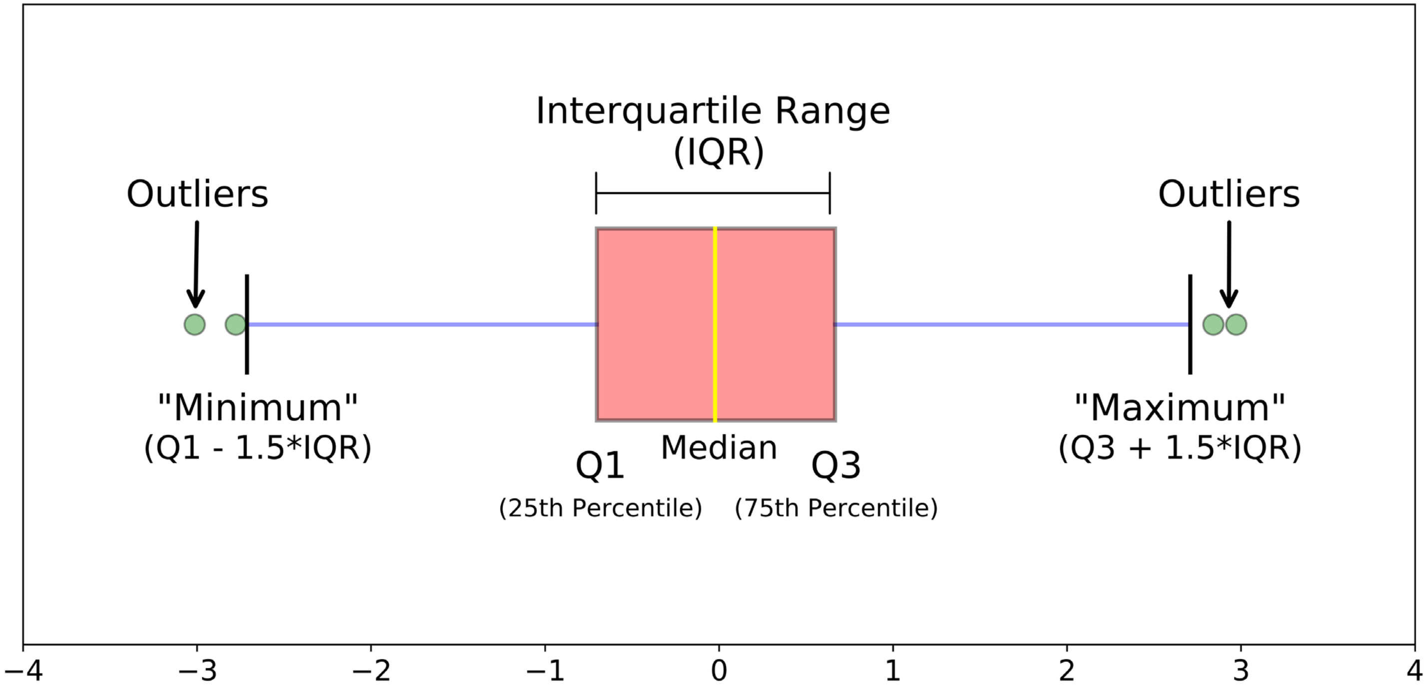
Variable	Definition	Mean	Std. Dev.	Min	Max
CompletionRate (%)	Extent (ratio) to which a consumer reads e-book content before writing a review	75.921	32.641	0	100
Reviewing	A binary variable indicating whether a customer writes a review after the e-book consumption (no = 0, yes = 1)	0.075	0.264	0	1
ReviewValence	Numerical review ratings given by each consumer	4.527	0.957	1	5
ReviewLength	Number of words in a review given by each consumer	8.411	23.793	1	798
ReviewExtremity	The deviation from the mean value of total reviews	0.708	0.644	0.467	3.533
Gender	A binary variable indicating gender of consumers (Female = 0, Male = 1)	0.050	0.218	0	1
Age	Consumer's age	41.015	8.969	17	98
RegistrationDate	The number of days elapsed since a consumer registered on the platform to July 19, 2016 (cutoff day)	857.518	568.438	53	2,870
Price (USD)	Fixed price of a book	4.312	2.910	0.182	73.18
PublicationDate	The number of days elapsed since an e-book was published to July 19, 2016 (cutoff day)	361.077	392.844	52	2,624
Adult	A binary variable indicating whether an e-book belongs to an adult category	0.870	0.336	0	1

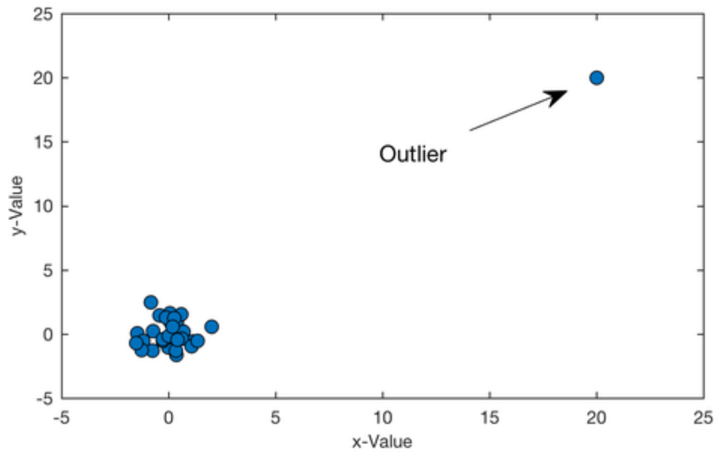
Note. Std. Dev., standard deviation.

Data Exploration & Visualization

(2) Outlier Detection by using Box Plot and Pre-processing

- A box plot (or box-whisker plot) shows the distribution of a variable and **identifies outliers**.
- Elements of a generic boxplot:





[Classification] Decision Tree

(1) Splitting the data into training and test data

```
# load the data into data.frame
titanic <- read.csv("titanic.csv", header=TRUE, stringsAsFactors=TRUE)
str(titanic)                                         # 1309 rows

# split the data into training and testing data sets
# we will first randomly select 2/3 of the rows
set.seed(345)                                        # for reproducible results
train = sample(1:nrow(titanic), nrow(titanic)*(2/3)) # replace=F by default
train

# Use the train index set to split the dataset
titanic.train = titanic[train,]                      # 872 rows
titanic.test = titanic[-train,]                      # the other 437 rows
```

[Classification] Decision Tree

 # display basic results

```
> fit
n= 872

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 872 342 No (0.60779817 0.39220183)
   2) sex=male 546 109 No (0.80036630 0.19963370)
     4) age>=13.5 508  87 No (0.82874016 0.17125984) *
     5) age< 13.5 38  16 Yes (0.42105263 0.57894737) *
   3) sex=female 326  93 Yes (0.28527607 0.71472393)
     6) pclass=Lower 158  74 No (0.53164557 0.46835443)
      12) fare>=19.7354 33   9 No (0.72727273 0.27272727) *
      13) fare< 19.7354 125  60 Yes (0.48000000 0.52000000)
        26) age>=24.5 69  32 No (0.53623188 0.46376812) *
        27) age< 24.5 56  23 Yes (0.41071429 0.58928571) *
     7) pclass=Middle,Upper 168   9 Yes (0.05357143 0.94642857) *
```

Split

Number of Obs.

Number Incorrect

Assigned Class

Proportions of neg/pos

Leaf Node

(2) Growing a tree and display basic results

Numbering scheme:
Root node has number 1.
Children of node x :
- left child: 2x
- right child: 2x + 1

[Classification] Decision Tree

 # tree interpretation

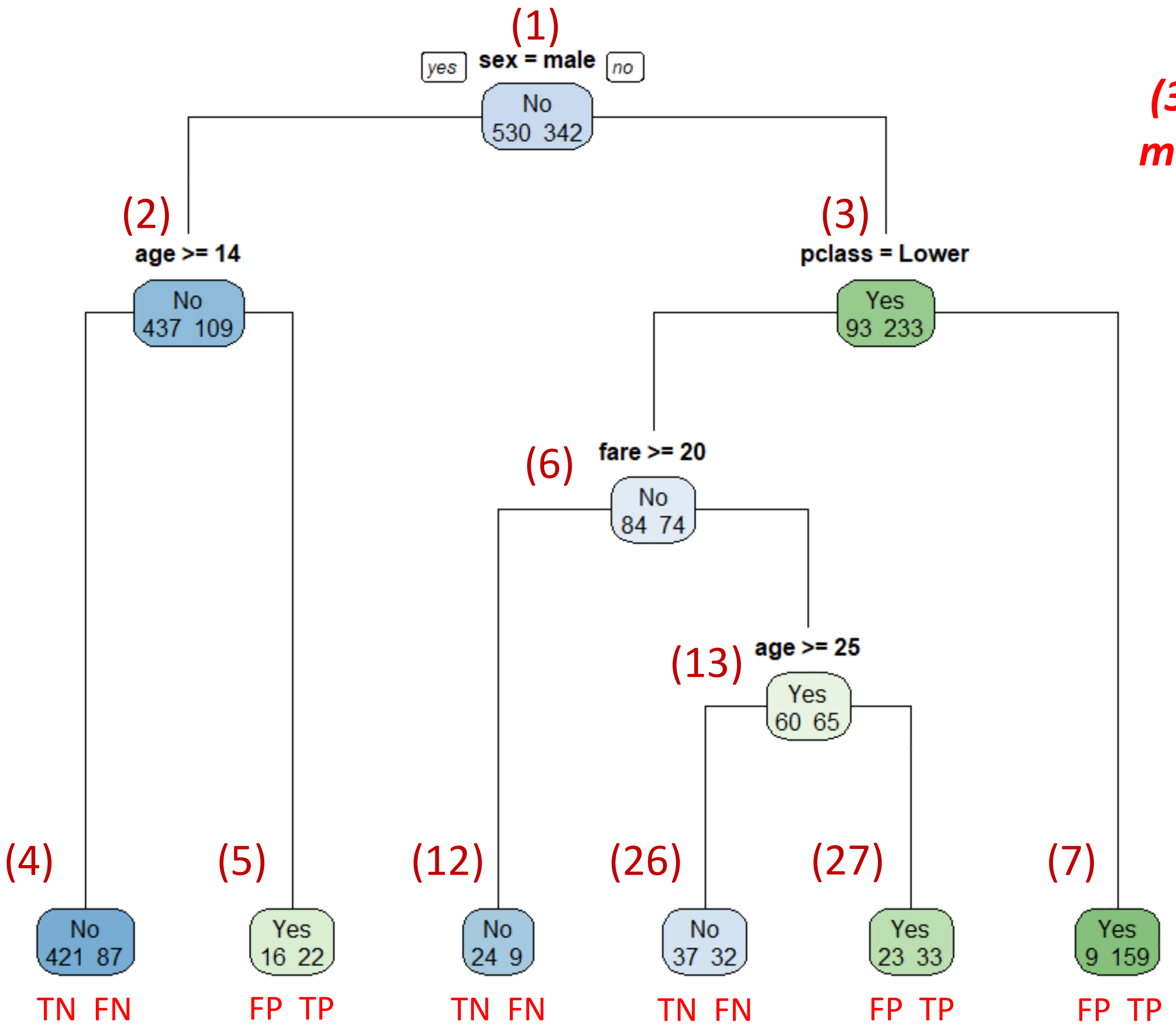
Definitions
True Positive (TP):
Pred Pos & Actual Pos

False Positive (FP):
Pred Pos & Actual Neg

True Negative (TN):
Pred Neg & Actual Neg

False Negative (FN):
Pred Neg & Actual Pos

Summary
TP = 22+33+159 = 214
FP = 16+23+9 = 48
TN = 421+24+37 = 482
FN = 87+9+32 = 128



(3) Plotting a tree and make an interpretation

[Classification] Decision Tree

(3) Plotting a tree and make an interpretation

```
# extract the vector of predicted class for each observation in titanic.train
titanic.pred <- predict(fit, titanic.train, type="class")
# extract the actual class of each observation in titanic.train
titanic.actual <- titanic.train$survived

# now build the confusion matrix
# which is the contingency table of predicted vs actual
confusion.matrix <- table(titanic.pred, titanic.actual)
confusion.matrix
```

	titanic.actual	
titanic.pred	No	Yes
No	482 (TN)	128 (FN)
Yes	48 (FP)	214 (TP)

[Classification] Decision Tree

```
# Accuracy on the Training Data
titanic.pred <- predict(fit, titanic.train, type="class")
titanic.actual <- titanic.train$survived
confusion.matrix <- table(titanic.pred, titanic.actual)
pt <- prop.table(confusion.matrix)
#accuracy
pt[1,1] + pt[2,2]
[1] 0.798
```

```
# Accuracy on the Testing data
titanic.pred <- predict(fit, titanic.test, type="class")
titanic.actual <- titanic.test$survived
confusion.matrix <- table(titanic.pred, titanic.actual)
addmargins(confusion.matrix)
pt <- prop.table(confusion.matrix)
#accuracy
pt[1,1] + pt[2,2]
[1] 0.801
```

(4) Accuracy on the training & test data

[Classification] Logistic Regression

(1) Splitting the data into training and test data

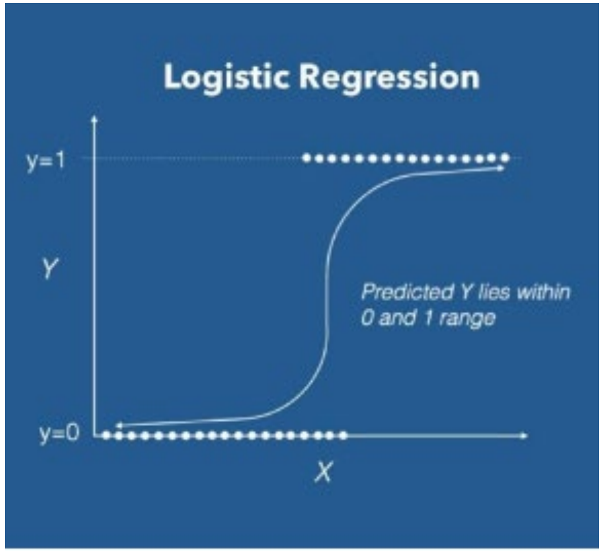
```
# load the data
bank.df <- read.csv("UniversalBank.csv")
# convert output as factor
bank.df$PersonalLoan <- as.factor(bank.df$PersonalLoan)
# treat Education as categorical
bank.df$Education <- factor(bank.df$Education, levels = c(1, 2, 3),
                     labels = c("Undergrad", "Graduate", "Advanced/Professional"))

# split the data into training and test data sets
set.seed(2)   # for reproducible results
train <- sample(1:nrow(bank.df), (0.6)*nrow(bank.df))
train.df <- bank.df[train,]
test.df <- bank.df[-train,]
```

[Classification] Logistic Regression

(2) Logistic regression and display the results

```
# run logistic regression
# use glm() (general linear model) with family = "binomial" to fit a logistic
logit.reg <- glm(PersonalLoan ~ Age + Experience + Income + Family + CCAvg +Education
            + Mortgage + SecuritiesAccount + CDAccount + Online + CreditCard,
                  data = train.df, family = "binomial")
```



$$y = \frac{e^{(b_0 + b_1 X)}}{1 + e^{(b_0 + b_1 X)}}$$

[Classification] Logistic Regression

(3) Interpretation of significancy and coefficient

 # results of logistic regression

summary(logit.reg)

```
Coefficients:
                              Estimate Std. Error z value Pr(>|z|)    
(Intercept)                 -1.558e+01  2.235e+00  -6.971 3.14e-12 ***
Age                          7.432e-02  8.043e-02   0.924  0.35545    
Experience                  -5.930e-02  8.012e-02  -0.740  0.45922    
Income                       6.273e-02  4.005e-03  15.666  < 2e-16 ***
Family                       5.476e-01  9.788e-02   5.595 2.21e-08 ***
CCAvg                        1.652e-01  5.887e-02   2.805  0.00503 ** 
EducationGraduate            4.229e+00  3.614e-01  11.701  < 2e-16 ***
EducationAdvanced/Professional 4.221e+00  3.622e-01  11.653  < 2e-16 ***
Mortgage                     1.134e-03  7.789e-04   1.456  0.14543    
SecuritiesAccount           -7.064e-01  3.820e-01  -1.849  0.06443 .  
CDAccount                    3.588e+00  4.345e-01   8.257  < 2e-16 ***
Online                      -5.603e-01  2.162e-01  -2.592  0.00955 ** 
CreditCard                  -1.223e+00  2.842e-01  -4.301 1.70e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

[Classification] Logistic Regression

(3) Interpretation of significancy and coefficient

 # interpretation for Income
summary(logit.reg)

- b_1 is estimated as 0.06273

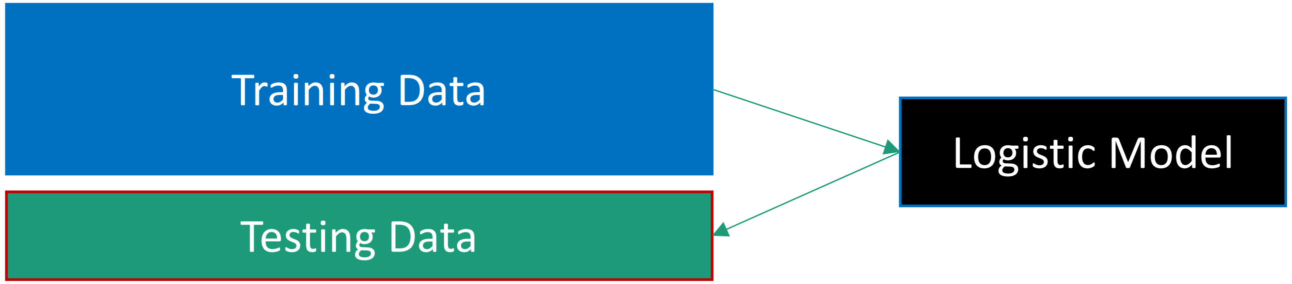
$$\textbf{\textit{odds ratio}} = e^{b_1} = e^{0.06273} = \mathbf{1.0647}$$

An increase of 1000$ of income **multiplies** the **odds** of acceptance of a personal loan by 1.0647

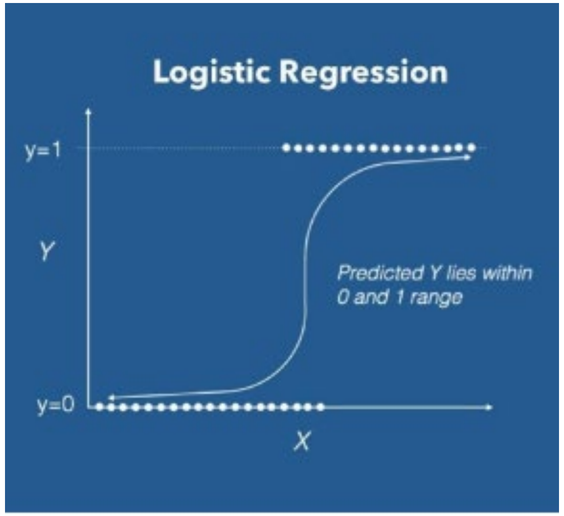
An increase of 1000$ of income is associated with an **increase** of 6.47% in the **odds** of acceptance of a personal loan

[Classification] Logistic Regression

 # use **predict()** **with** **type = "response"** **to compute predicted probabilities**
 # i.e., the estimated probability of an observation being in class "1"
logitPredict <- predict(logit.reg, test.df, type = "response")



(4) Accuracy on the training & test data



 # Convert probability to a classification
 # if probability > cutoff, then class = 1, otherwise class =0
logitPredictClass <- ifelse(logitPredict > 0.5, 1, 0)

[Classification] Logistic Regression

 # Confusion matrix

```
> actual <- test.df$PersonalLoan
> predict <- logitPredictClass
> cm <- table(predict, actual)
                    actual
predict    0          1
      0    1794   65
      1    18      123
```

 # consider class "1" as positive

```
> tp <- cm[2,2]
> tn <- cm[1,1]
> fp <- cm[2,1]
> fn <- cm[1,2]
```

 # Accuracy

```
> (tp + tn)/(tp + tn + fp + fn)
[1] 0.9585
```

 # TPR = Recall = Sensitivity

```
> tp/(fn+tp)
[1] 0.6542553
```

 # TNR = Specificity

```
> tn/(fp+tn)
[1] 0.9900662
```

 # FPR

```
> fp/(fp+tn)
[1] 0.009933775
```

 # FNR

```
> fn/(fn+tp)
[1] 0.3457447
```

(4) Accuracy on the training & test data