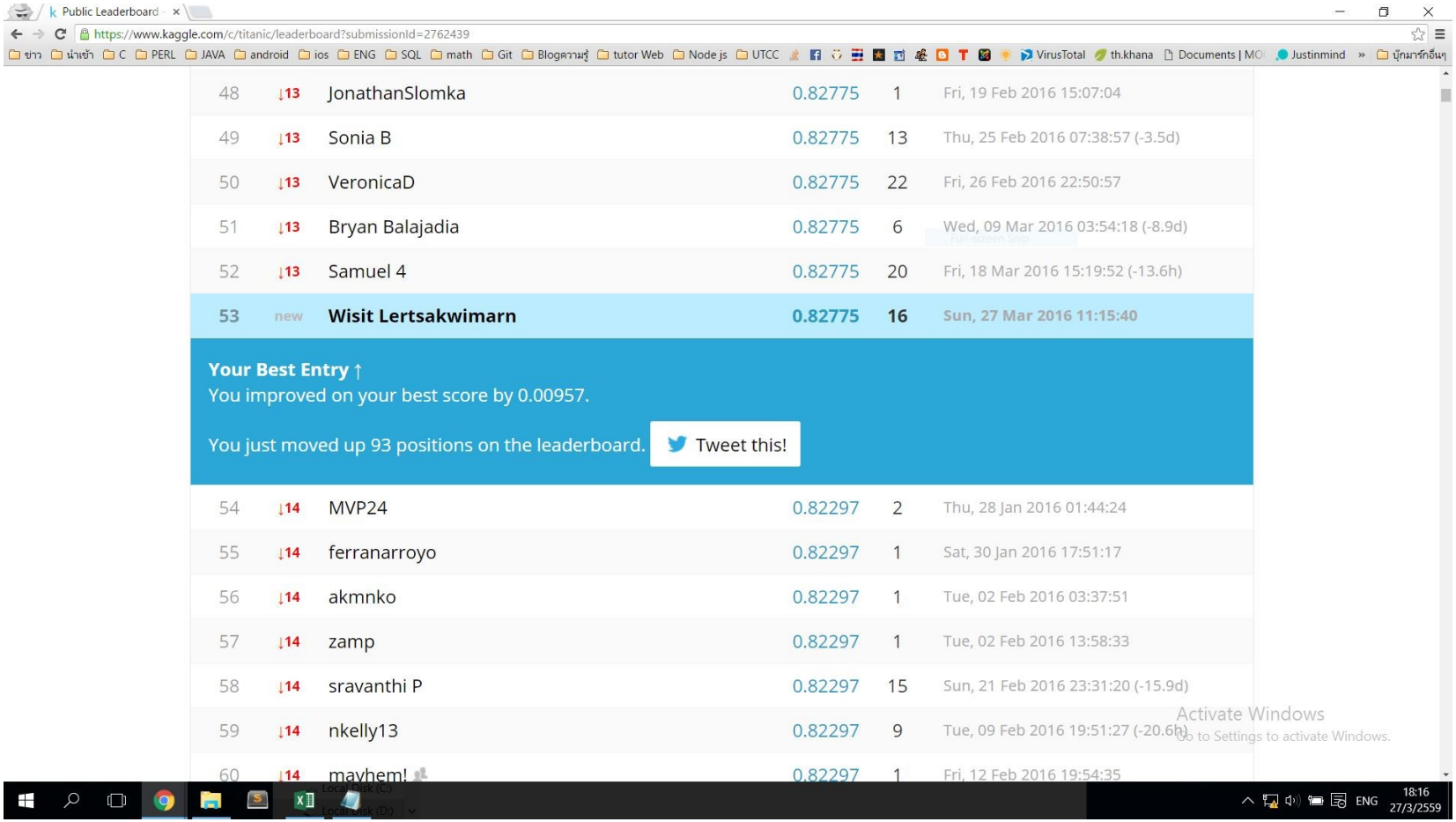
```
#write by 5605104046 นาย วิศิษฐ์ เลิศศักดิ์วิมาน

#write by 5605104043 นาย อานนท์ กันทา

# Load the data sets

setwd("D:/Com/ICT_UTCC/DATA MINING/titanic-master/")

train <- read.csv("Data/CSV/train.csv", stringsAsFactors = FALSE)  # 891 obj ชุด

test <- read.csv("Data/CSV/test.csv", stringsAsFactors = FALSE)    # 418 obj ชุด

#Import library

library(rpart)

library(party)

library(rattle)

library(filewrite)

library(randomForest)

# Clean data sets (1)- (8)

#(1) ทำการใส่ค่า NA ชื่อเต็มคือ Not Available(ข้อมูลสูญหาย) ลงไปในช่องว่างในชุดข้อมูลฝึกสอน

train[train == ""] <- NA

#(2) ทำการสร้างSurvivedขึ้นมาในชุดข้อมูลทดสอบtest และกำหนดค่า NA ชื่อเต็มคือ Not Available (ข้อมูลสูญหาย) ลงไปในตัวแปรSurvived ในชุดข้อมูลทดสอบ

test$Survived <- NA

#(3) ทำการรวมชุดข้อมูลฝึกสอนและข้อมูลทดสอบไว้ด้วยกัน เพื่อ ความสะดวกในการทำความสะอาดข้อมูล โดยชุดข้อมูลทดสอบต่อท้ายชุดข้อมูลฝึกสอน และเก็บผลลัทธ์ไว้ในตัวแปรออบเจ็ค combi

combi <- rbind(train, test)
```

```
#(4) ทำการแยกส่วนของตัวแปร Nameเป็น2ส่วน แยกด้วยฟังก์ชันstrsplit ตัวอักษรที่ใช้แยกคือ , กับ . และจัดเก็บไว้ในตัวแปร Title
#    ค้นหา' 'ช่องว่าง1ช่อง แทนด้วย''ไม่มีช่องว่าง *ผลลัพธ์ที่ได้คือคำขึ้นต้นชื่อจากตัวแปรName และจัดเก็บไว้ในตัวแปร Title
#    และทำการกำหนดค่า Mlle แทนลงไปในคำขึ้นต้นช่องที่มีคำว่า Mme หรือ Mlle
#    และทำการกำหนดค่า Lady แทนลงไปในคำขึ้นต้นช่องที่มีคำว่า Dona หรือ Lady หรือ the Countess
#    และทำการกำหนดค่า Sir แทนลงไปในคำขึ้นต้นช่องที่มีคำว่า Capt หรือ Don หรือ Major หรือ Sir หรือ Jonkheer หรือ Dr
#    ตัวแปร Title *มีวัตถุประสงค์เพื่อหาคำขึ้นต้นชื่อ
#    สุดท้ายทำการแปลง Titleจากข้อมูลแบบvector ให้เป็นข้อมูลแบบfactor โดยฟังก์ชัน factor
combi$Title <- sapply(combi$Name, FUN=function(x) {strsplit(x, split='[,.]')[[1]][2]})
combi$Title <- sub(' ', '', combi$Title)
combi$Title[combi$Title %in% c('Mme', 'Mlle')] <- 'Mlle'
combi$Title[combi$Title %in% c('Dona', 'Lady', 'the Countess')] <- 'Lady'
combi$Title[combi$Title %in% c('Capt', 'Don', 'Major', 'Sir', 'Jonkheer', 'Dr')] <- 'Sir'
combi$Title <- factor(combi$Title)
#(5)  ทำการใส่ขนาดของครอบครัวลงไปโดย นำจำนวนบุคคลที่เดินทางมาด้วย (ญาติพี่น้อง สามี ภรรยา) + จำนวนบุคคลที่เดินทางมาด้วย (พ่อแม่ ลูก) + 1(จำนวนของตัวเอง)
# และทำการกำหนดค่าจัดเก็บไว้ในตัวแปร FamilySize *มีวัตถุประสงค์เพื่อหาขนาดของครอบครัว
combi$FamilySize <- combi$SibSp + combi$Parch + 1
#(6) ทำการตัดนามสกุลจากตัวแปรName และจัดเก็บไว้ในตัวแปร Title ในตัวแปร Surname
#ทำรวมตัวแปรSurnameกับFamily Size เข้าด้วยกันโดยขึ้นต้นด้วยค่าในตัวแปรFamily Sizeและตามด้วยค่าในตัวแปรSurname
# โดยทำแทน Small ลงไปใน FamilyID ที่ FamilySize <= 2
```

```
# ตัวแปร FamilyID *มีวัตถุประสงค์เพื่อแยกประเภทขนาดความใหญ่ของครอบครัว

# สุดท้ายทำการแปลง FamilyID จากข้อมูลแบบvector ให้เป็นข้อมูลแบบfactor โดยฟังก์ชัน factor

combi$Surname <- sapply(combi$Name, FUN=function(x) {strsplit(x, split='[,.]')[[1]][1]})

combi$FamilyID <- paste(as.character(combi$FamilySize), combi$Surname, sep="")

combi$FamilyID[combi$FamilySize <= 2] <- 'Small'

famIDs <- data.frame(table(combi$FamilyID))

famIDs <- famIDs[famIDs$Freq <= 2,]

combi$FamilyID[combi$FamilyID %in% famIDs$Var1] <- 'Small'

combi$FamilyID <- factor(combi$FamilyID)

#(7) ทำการกำหนดค่าลงไปในตัวแปรFare ที่มีIndex = 1044 โดยค่าที่ใส่เป็นค่ากึ่งกลาง(มัธยฐาน) จากFareทั้งหมด na.rm=TRUE หมายถึงให้ทำการคำนวณทียกเว้นข้อมูลสูญหาย(NA)

combi$Fare[1044] <- median(combi$Fare, na.rm=TRUE)

#(8)  ทำการกำหนดค่าลงไปในตัวแปรEmbarked , แถวลำดับที่62กับ830 ด้วยค่าSเนื่องจากค่าSถูกแทนมากที่สุด

# ฟังก์ชันc คือเก็บข้อมูลอยู่ในรูปแบบvector

combi$Embarked[c(62,830)] = "S"

# ทำการสร้างโมเดลในการพยากรณ์หาค่าอายุAge ที่ว่างอยู่ จากข้อมูลฝึกสอนกับข้อมูลทดสอบ โดยมีตัวแปรเป้าหมายคือAge และมีตัวแปรใช้เพื่อหาตัวแปรเป้าหมาย คือ

# Pclass , Sex  ,  SibSp  ,  Parch  ,  Fare  ,  Embarked  ,  Title  ,  FamilySize

# โดยใช้ฟังก์ชัน rpart และใช้ method="anova" ใช้สำหรับต้นไม้ถดถอย regression tree <- พยากรณ์เชิงปริมาณ

# ผลลัพธ์ที่ได้จาก ฟังก์ชันrpart เป็นการสร้างโมเดลการพยากรณ์อยู่ในรูปแบบของออบเจ็ค ให้เก็บไว้ไว้ที่ตัวแปรออบเจ็ค  AgeModel

# ทำการพยากรณ์โดยใช้AgeModel ในการพยากรณ์หาค่าข้อมูลสูญหาย(NA)ในตัวแปรAge และทำการกำหนดค่าแทนลงไปในค่าข้อมูลสูญหาย(NA)
```

```
# *ผลลัพธ์ที่ได้ คือ *แทนค่าที่ได้จากการพยากรณ์ลงไปในค่าข้อมูลสูญหาย(NA) ในตัวแปรAge

AgeModel <- rpart(Age ~ Pclass + Sex + SibSp + Parch + Fare + Embarked + Title + FamilySize, data=combi[!is.na(combi$Age),], method="anova")

combi$Age[is.na(combi$Age)] <- predict(AgeModel, combi[is.na(combi$Age),])

#ทำการแปลง (Sex, Embarked ,Survived) จากข้อมูลแบบvector ให้เป็นข้อมูลแบบfactor โดยฟังก์ชัน factor

combi$Sex <- factor(combi$Sex)

combi$Embarked <- factor(combi$Embarked)

combi$Survived <- factor(combi$Survived)

# ทำการแยกชุดข้อมูลฝึกสอนและข้อมูลทดสอบ ( combi)ออกจากกัน ถูกกำหนดไว้ในตัวแปรออบเจ็ค train กับ test

train <- combi[1:891,]

test <- combi[892:1309,]

# สร้างโมเดลในการพยากรณ์หาผู้รอดชีวิต จากข้อมูลฝึกสอน train โดยมีตัวแปรเป้าหมายคือ Survived และมีตัวแปรใช้เพื่อหาตัวแปรเป้าหมาย คือ

#  Pclass , Sex ,  Age ,  SibSp ,  Parch, Fare, Embarked , Title , FamilySize , FamilyID

# โดยใช้ฟังก์ชัน cforest กำหนดให้ จำนวนต้นไม้(ntree) =5000 กับ จำนวนของตัวแปรสุ่มที่ใช้ในการแบ่ง(mtry) =2

# ฟังก์ชัน cforest มีลักษณะคล้ายกับ Random forest แต่ กำหนดจำนวนของตัวแปรสุ่มที่ใช้ในการแบ่งได้

# กำหนด seed เท่ากับ 415 (ตัวเลขมีผลกับการพยากรณ์)

# *ผลลัพธ์ที่ได้จากฟังก์ชันcforest คือ โมเดลการพยากรณ์ที่อยู่ในรูปแบบของออบเจ็ค และถูกกำหนดไว้ในตัวแปร Model

set.seed(415)

Model <- cforest(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked + Title + FamilySize + FamilyID, data = train,controls=cforest_unbiased(ntree=5000,
mtry=2))
```

```
# ทำการพยากรณ์ โดยนำข้อมูลทดสอบ(test) มาทดสอบการพยากรณ์กับโมเดลที่สร้างไว้ Model

# *ผลลัพธ์ที่ได้จากฟังก์ชันpredict คือ ผลของการพยากรณ์อยู่ในรูปแบบของออบเจ็ค กำหนดให้เก็บไว้ในตัวแปรออบเจ็ค Prediction

Prediction <- predict(Model, test, OOB=TRUE, type = "response")

# ทำการจัดข้อมูลให้อยู่รูปแบบของตารางโดยมี 2 colum คือ

# colum แรกชื่อ PassengerId และใส่ค่า PassengerId ของข้อมูลทดสอบลงไป

# colum สองชื่อ Survived และใส่ค่า Prediction ที่ได้จากการพยาการณ์ลงไป

# *ผลลัพธ์ที่ได้จากฟังก์ชันdata.frame  คือข้อมูลที่อยู่ในรูปแบบตารางโดยมี 2colum (PassengerId กับ Survived) กำหนดให้เก็บไว้ในตัวแปรออบเจ็ค submit

submit <- data.frame(PassengerId = test$PassengerId, Survived = Prediction)

# กำหนดให้ทำการเขียนไฟล์การนำข้อมูลจากตัวแปรออบเจ็ค submit เขียนข้อมูลลงไปใน Path “ Data/CSV/ “ ไฟล์ชื่อCF.csv

#ทำการนำข้อมูลจากตัวแปร train เขียนข้อมูลลงไปใน Path “ Data/CSV/ “ ไฟล์ชื่อ Clean_train .csv

#ทำการนำข้อมูลจากตัวแปร test เขียนข้อมูลลงไปใน Path “ Data/CSV/ “ ไฟล์ชื่อ Clean_test.csv

write.file("Data/CSV/Clean_train.csv")

write.file("Data/CSV/Clean_test.csv")

write.file("Data/CSV/CF.csv")

write.csv(train,"Data/CSV/Clean_train.csv", row.names = FALSE)

write.csv(test,"Data/CSV/Clean_test.csv", row.names = FALSE)

write.csv(submit,"Data/CSV/CF.csv", row.names = FALSE)
```


Public Leaderboard
https://www.kaggle.com/c/titanic/leaderboard?submissionId=2762439
48 ↓13 JonathanSlomka 0.82775 1 Fri, 19 Feb 2016 15:07:04
49 ↓13 Sonia B 0.82775 13 Thu, 25 Feb 2016 07:38:57 (-3.5d)
50 ↓13 VeronicaD 0.82775 22 Fri, 26 Feb 2016 22:50:57
51 ↓13 Bryan Balajadia 0.82775 6 Wed, 09 Mar 2016 03:54:18 (-8.9d)
52 ↓13 Samuel 4 0.82775 20 Fri, 18 Mar 2016 15:19:52 (-13.6h)
53 new Wisit Lertsakwimarn 0.82775 16 Sun, 27 Mar 2016 11:15:40
Your Best Entry ↑
You improved on your best score by 0.00957.
You just moved up 93 positions on the leaderboard.
Tweet this!
54 ↓14 MVP24 0.82297 2 Thu, 28 Jan 2016 01:44:24
55 ↓14 ferranarroyo 0.82297 1 Sat, 30 Jan 2016 17:51:17
56 ↓14 akmnko 0.82297 1 Tue, 02 Feb 2016 03:37:51
57 ↓14 zamp 0.82297 1 Tue, 02 Feb 2016 13:58:33
58 ↓14 sravanthi P 0.82297 15 Sun, 21 Feb 2016 23:31:20 (-15.9d)
59 ↓14 nkelly13 0.82297 9 Tue, 09 Feb 2016 19:51:27 (-20.6h)
60 ↓14 mayhem! 0.82297 1 Fri, 12 Feb 2016 19:54:35
Activate Windows
Go to Settings to activate Windows.
ENG
18:16
27/3/2559