```
trainset <- read.csv("/Users/byu/Desktop/Data/titanic-train.csv")
trainset$Survived=factor(trainset$Survived)
trainset$Pclass=ordered(trainset$Pclass)

testset <- read.csv("/Users/byu/Desktop/Data/titanic-test.csv")
testset$Survived=factor(testset$Survived)
testset$Pclass=ordered(testset$Pclass)

myVars=c("Pclass", "Sex", "Age", "SibSp", "Fare", "Survived")
newtrain=trainset[myVars]
newtest=testset[myVars]

# replace missing value with mean and mode
MS <- make_Weka_filter("weka/filters/unsupervised/attribute/ReplaceMissingValues")
newtrain <-MS(data=newtrain, na.action = NULL)
newtest <-MS(data=newtest, na.action = NULL)

# train J48 model using RWeka
library("RWeka")
m=J48(Survived~., data = newtrain)
m=J48(Survived~., data = newtrain, control=Weka_control(U=FALSE, M=2, C=0.5))
e=evaluate_Weka_classifier(m, seed=1, numFolds=10)
pred=predict (m, newdata = newtest, type = c("class"))
myids=c("PassengerId")
id_col=testset[myids]
newpred=cbind(id_col, pred)
colnames(newpred)=c("Passengerid", "Survived")
write.csv(newpred, file="titanic-J48-pred.csv", row.names=FALSE)
```

```
InfoGainAttributeEval(Survived ~ . , data = trainset)


library(e1071)
# https://cran.r-project.org/web/packages/e1071/e1071.pdf
# some variables are numeric, some are nominal
# this algorithm uses normal distribution to estimate prob for numeric variables
nb=naiveBayes(Survived~., data = newtrain, laplace = 1, na.action = na.pass)
pred=predict(nb, newdata=newtest, type=c("class"))
myids=c("PassengerId")
id_col=testset[myids]
newpred=cbind(id_col, pred)
colnames(newpred)=c("Passengerid", "Survived")
write.csv(newpred, file="titanic-NB-pred.csv", row.names=FALSE)

# since the numeric variables may not follow normal distribution
# test if discretization would improve the performance
# use infotheo package for discretization
# faster than RWeka discretization filter
# Kaggle returned lower accuracy .727
library(infotheo)
#combine train and test data for unified discretization
data <- rbind(newtrain, newtest)
dData <- discretize(data[, 2:4], disc = "equalwidth", nbins=10)
dData <- lapply(dData, as.factor)
dData <- cbind(data[, c(1,6)], dData)
dlabel <- data$Survived
dData <- cbind(dData, dlabel)
# separate train (1-891) and test
```

```
train_index <- 1:891
train1<- dData[train_index,]
test1<- dData[-train_index,]

nb=naiveBayes(Survived~., data = train1, laplace = 1, na.action = na.pass)
pred=predict(nb, newdata=test1, type=c("class"))
myids=c("PassengerId")
id_col=testset[myids]
newpred=cbind(id_col, pred)
colnames(newpred)=c("Passengerid", "Survived")
write.csv(newpred, file="titanic-binned-NB-pred.csv", row.names=FALSE)

# kNN in the "class" package
# no missing values are allowed
# no nominal values are allowed
# labels should be separated from train and test data
# Kaggle returned accuracy .617
# install.packages("class")
library("class")
train_labels = newtrain$Survived
sex=as.numeric(newtrain$Sex)
pclass=as.numeric(newtrain$Pclass)
dtrain=cbind(sex, newtrain[, c(2,3,4)] )
dtrain=cbind(dtrain, pclass)

sex=as.numeric(newtest$Sex)
pclass=as.numeric(newtest$Pclass)
dtest=cbind(sex, newtest[, c(2,3,4)] )
dtest=cbind(dtest, pclass)
```

```
predKNN <- knn(train=dtrain, test=dtest, cl=train_labels, k=3)
myids=c("PassengerId")
id_col=testset[myids]
newpred=cbind(id_col, predKNN)
colnames(newpred)=c("Passengerid", "Survived")
write.csv(newpred, file="titanic-kNN-pred.csv", row.names=FALSE)



# SVM, acc .77990
library(e1071)
svm<- svm(Survived~., data = newtrain)
pred=predict(svm, newdata=newtest, type=c("class"))
myids=c("PassengerId")
id_col=testset[myids]
newpred=cbind(id_col, pred)
colnames(newpred)=c("Passengerid", "Survived")
write.csv(newpred, file="titanic-SVM-pred.csv", row.names=FALSE)

# random forest on non-discretized data
# Kaggle returned accuracy .727
install.packages("randomForest")
library(randomForest)
rfm <- randomForest(Survived~., data=newtrain, ntree=10)
print(rfm)
predRF <- predict(rfm, newtest, type=c("class"))
myids=c("PassengerId")
id_col=testset[myids]
newpred=cbind(id_col, pred)
```

```
colnames(newpred)=c("Passengerid", "Survived")
write.csv(newpred, file="titanic-RF-pred.csv", row.names=FALSE)
```