1. Association rule mining

List<-read.csv("./data/titanic.csv")

List

summary(List)

str(List)

List$name<-NULL

List$fare<-NULL

List<-na.omit(List)

List$age<-ifelse(List$age>18,List$age<-"adult",List$age<-"child")

List$survived<-ifelse(List$survived,List$survived<-"yes",List$survived<-"no")

List$age<-factor(List$age)

List$survived<-factor(List$survived)

List$pclass<-factor(List$pclass)

List$sibsp<-factor(List$sibsp)

List$parch<-factor(List$parch)

List$embarked<-factor(List$embarked)

str(List)

summary(List)

library(arules)

rules<-apriori(List,control=list(verbose=F),parameter=list(minlen=2,supp=0.4,conf=0.5),appearance=list(rhs=c("survived=no","survived=yes"),default="lhs"))

quality(rules)<-round(quality(rules),digits=3)

inspect(rules)

install.packages("arulesViz")

library(arulesViz)

plot(rules)

plot(rules,method="grouped")

rules.sorted<-sort(rules,by="support")

subset.matrix<-is.subset(rules.sorted,rules.sorted)

subset.matrix

subset.matrix[lower.tri(subset.matrix,diag=T)]<-0

subset.matrix

redundant<-colSums(subset.matrix)>0

which(redundant)

rules.pruned<-rules.sorted[!redundant]

inspect(rules.pruned)

1. Regression analysis

List<-read.csv("./data/titanic\_full.csv")

List$name<-NULL

List<-na.omit(List)

List$age<-ifelse(List$age>18,List$age<-"adult",List$age<-"child")

classfare<-List[c("pclass","fare")]

cor(classfare$pclass,classfare$fare)

lm(pclass~fare,data=classfare)

cld <-lm(pclass~fare,data=classfare)

plot(pclass~fare,data=List,pch=20,col='red')

abline(cld)

anova(cld)

1. Decision tree

**List<-read.csv("./data/titanic\_full.csv")**

**List$name<-NULL**

**List<-na.omit(List)**

**List$survived<-ifelse(List$survived,List$survived<-"yes",List$survived<-"no")**

**List$survived<-factor(List$survived)**

**List$pclass<-factor(List$pclass)**

**List$sibsp<-factor(List$sibsp)**

**List$parch<-factor(List$parch)**

**List$embarked<-factor(List$embarked)**

**List$sex<-factor(List$sex)**

**install.packages("caret")**

**install.packages("rpart")**

**install.packages("rpart.plot")**

**install.packages("tree")**

**install.packages('C50')**

**library(C50)**

**library(caret)**

**library(rpart)**

**library(rpart.plot)**

**library(tree)**

**set.seed(1000)**

**intrain<-createDataPartition(y=List$survived, p=0.7, list=FALSE)**

**train<-List[intrain, ]**

**test<-List[-intrain, ]**

**treemod<-tree(survived~. , data=train)**

**plot(treemod)**

**text(treemod)**

**cv.trees<-cv.tree(treemod, FUN=prune.misclass )**

**plot(cv.trees)**

**prune.trees <- prune.misclass(treemod, best=5,6)**

**plot(prune.trees)**

**text(prune.trees, pretty=0)**