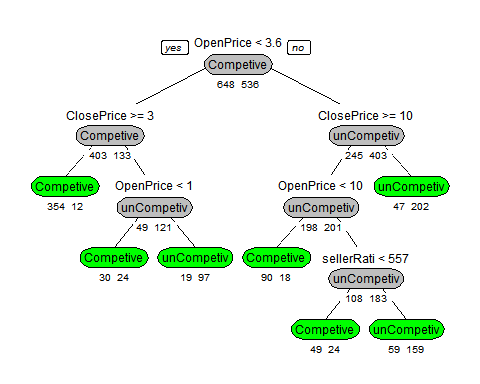
20181126 homework

## 'data.frame': 1972 obs. of 8 variables:  
## $ Category : Factor w/ 18 levels "Antique/Art/Craft",..: 14 14 14 14 14 14 14 14 14 14 ...  
## $ currency : Factor w/ 3 levels "EUR","GBP","US": 3 3 3 3 3 3 3 3 3 3 ...  
## $ sellerRating: int 3249 3249 3249 3249 3249 3249 3249 3249 3249 3249 ...  
## $ Duration : int 5 5 5 5 5 5 5 5 5 5 ...  
## $ endDay : Factor w/ 7 levels "Fri","Mon","Sat",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ ClosePrice : num 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...  
## $ OpenPrice : num 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...  
## $ Competitive.: int 0 0 0 0 0 0 0 0 0 0 ...

#### (a)



According to the result above ,the tree plot show us the decision rule in detial.And the rule is only that using variable *OpenPrice* ,*ClosePrice* and *sellerRating* as the node spliting criterion,and the *OpenPrice* as root node.It is resonable for us that most people only concern that the price of auction first.In common sense,the *OpencePrice* and *ClosePrice* actually have the main effect on the whether the auction is competive or not.So \*\*Price\* is the main consideration for auction decisions.

#### (b)

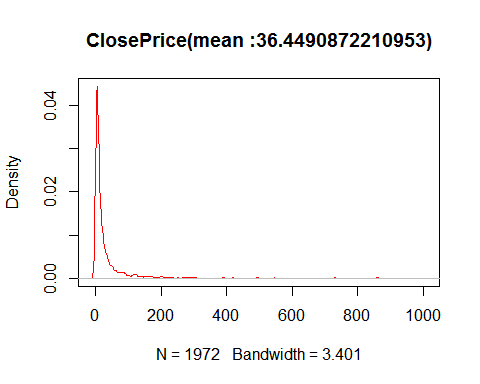
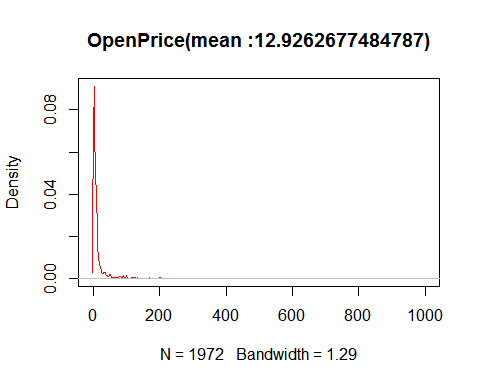
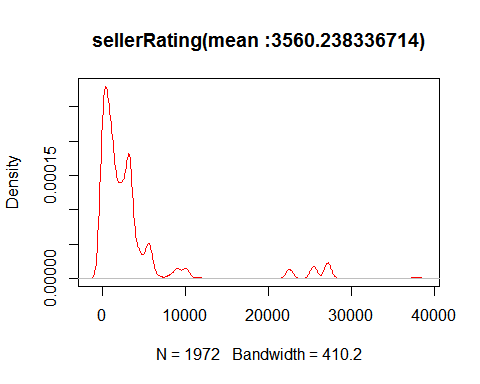
## The Accuracy of train dataset of the origin model is : 0.8285473

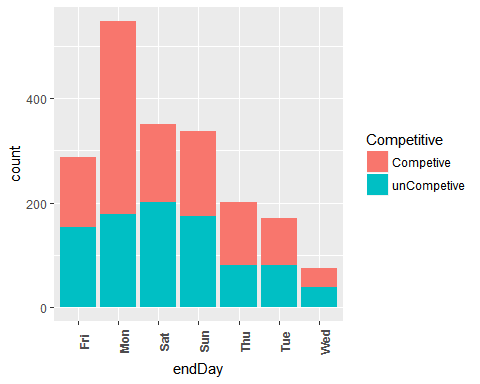
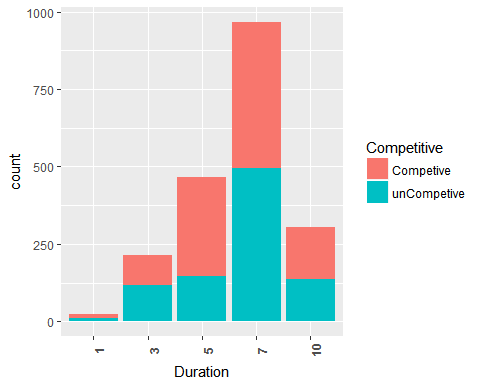
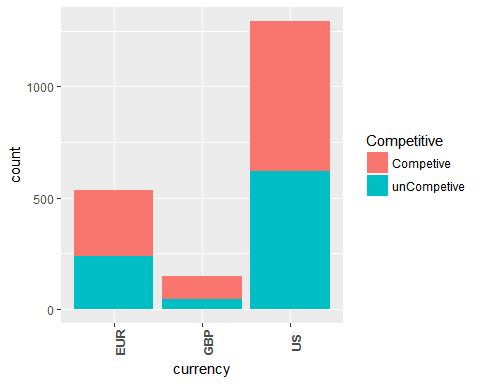
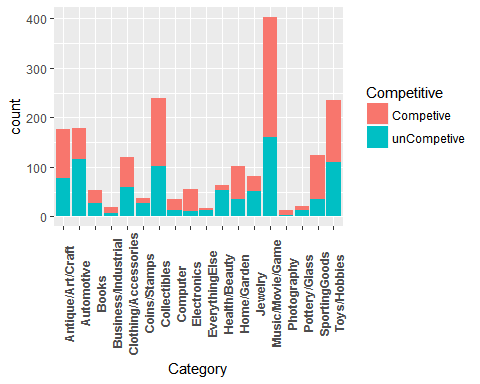
## The Accuracy of test dataset of the origin model is : 0.819797

The model uses *OpenPrice* ,*ClosePrice* and *sellerRating* as the predictors,that is said that the competition of *auction* is mainly related with these variables.And according to what we know in reality,this rule is reasonable.And the classification's accuracy of the train and test dataset is almost similiar,which mean the model is stable.So this model is practical for predicting the outcome of a new auction.

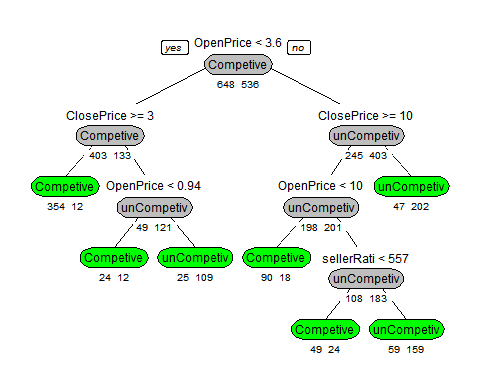
#### (c)

As we know ,the model used *OpenPrice,ClosePrice,sellerRati* as the predictors and removed the other variable(*Category,currency,Duration,endDay*),that is,the predictors are the intersting variable and the other variables are not intersting.

 According to the density plot above ,*sellerRating* has uneven distribution, large fluctuations.And they all are mainly distributed in lower level

 According to the plot above,In *Category*,Music/Movie/Game is most population in auction,and Collectibles secondly,and Photography is minimal;In *currency*,US is the most population,that is ,auction with US is main business channel;In *Duration*,7 is main effect on auction;In *endDay*,auctions on Monday are more dynamic.

#### (d)



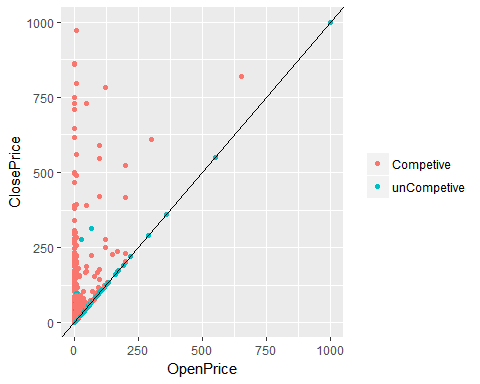
## The Accuracy of train dataset of the prune model is : 0.8336149

## The Accuracy of test dataset of the prune model is : 0.821066

Accordin to the conclusion in part(a) and part(b),the variable *OpenPrice,ClosePrice,sellerRating* are the predictors for predicting a new auction.So I only use these predictors to build the another tree.And the tree plot show us the decision rule in detial and that the acution decision rule is same as the result in part(a).But,the accuracy in train and test perform better than the tree classification in part(a),almost improve 1%.It is concluded that the smallest set of tree classification is *OpenPrice,ClosePrice,sellerRating*.

#### (e)

The best-pruned tree plot tell us that,the main spliting node is *OpenPrice* and *ClosePrice*,so using them for below plot.



According to the plot ,the auction which is competitive is mainly effected by *ClosePrice*.Acutually,it is reasonable So I think this splitting line is reasonable with respect to the meaning of the two predictors.Only Only a few are not split correctly by this line in graph.So it seems to do a good job of separating the two classes.

#### (f)

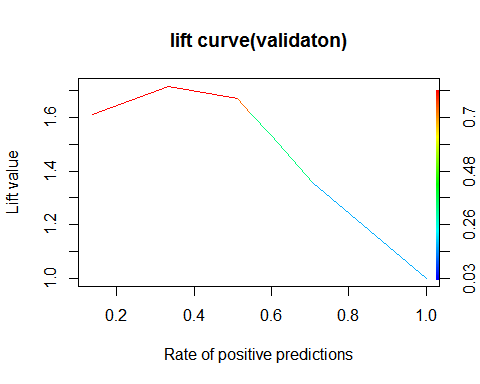
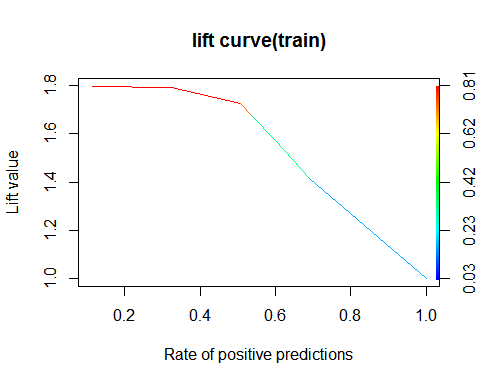
* ConfusionMatrix

## The train dataset confusionMatrix:

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Competive unCompetive  
## Competive 517 66  
## unCompetive 131 470  
##   
## Accuracy : 0.8336   
## 95% CI : (0.8112, 0.8544)  
## No Information Rate : 0.5473   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.6677   
## Mcnemar's Test P-Value : 5.12e-06   
##   
## Sensitivity : 0.7978   
## Specificity : 0.8769   
## Pos Pred Value : 0.8868   
## Neg Pred Value : 0.7820   
## Prevalence : 0.5473   
## Detection Rate : 0.4367   
## Detection Prevalence : 0.4924   
## Balanced Accuracy : 0.8374   
##   
## 'Positive' Class : Competive   
##

## The validation dataset confusionMatrix:

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Competive unCompetive  
## Competive 331 54  
## unCompetive 87 316  
##   
## Accuracy : 0.8211   
## 95% CI : (0.7925, 0.8472)  
## No Information Rate : 0.5305   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.6426   
## Mcnemar's Test P-Value : 0.007041   
##   
## Sensitivity : 0.7919   
## Specificity : 0.8541   
## Pos Pred Value : 0.8597   
## Neg Pred Value : 0.7841   
## Prevalence : 0.5305   
## Detection Rate : 0.4201   
## Detection Prevalence : 0.4886   
## Balanced Accuracy : 0.8230   
##   
## 'Positive' Class : Competive   
##

* Lift Chart 

According to the confusionMatrix of train and test dataset,the result of them perform similiar,which indicateds that the model is stable.And Accuracy,Sensitivity,Specificity,etc criterion also have a nice score.The lift curve (train and validation) show us that the line hold a high boost value for a period, or slowly drop for a period, then quickly drop to 1,which mean that perform not bad.In conclusion,this model has a good performance.

#### (g)

Based on the last tree,we can clonclude that the chance of an auction is mainly effected by *OpenPrice*,*ClosePrice*,*sellerRating* as above what we said.The lower OpenPrice,ClosePrice or sellerRating always can get a competitive auction.On the contrary,it will get uncompetitive result.If sellerRating is greater than 557,auction is uncompetitive.So,the pruned-best tree recommends us that we should choose lower OpenPrice,ClosePrice or sellerRating as the main reference strategy for us,it will have bigger chance to perform competitively.

#### Code:

# load necessary packages and dataset

library(rpart)

library(rpart.plot)

library(caret)

library(ROCR)

library(party)

eBaysAuctions<-read.csv("./eBayAuctions.csv",header = TRUE)

#str(eBaysAuctions) # check struct of dataset

str(eBaysAuctions)

#### Data Preprocessing

eBaysAuctions$Duration<-as.factor(eBaysAuctions$Duration) # Convert Duration into catergorical variable

eBaysAuctions$Competitive<-as.factor(eBaysAuctions$Competitive.)

eBaysAuctions$Competitive.<-NULL

eBaysAuctions$Competitive<-ifelse(eBaysAuctions$Competitive==0,"unCompetive","Competive")

eBaysAuctions$Competitive<-as.factor(eBaysAuctions$Competitive)

# split dataset into train(60%) and test(40%)

set.seed(20181126)

n<-dim(eBaysAuctions)[1]

train\_idx<-sample(1:n,size = ceiling(n\*0.6),replace = FALSE)

test\_idx<-setdiff(1:n,train\_idx)

train<-eBaysAuctions[train\_idx,]

test<-eBaysAuctions[test\_idx,]

#### (a)

model<-rpart(Competitive~.,data = train,method = "class",

control = list(minbucket=50,maxdepth=7,minsplit=1)) #,xval=5 # xval=5,5 fold validation

# plot tree

prp(model, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10,

box.col=ifelse(model$frame$var == "<leaf>", 'green', "gray"))

#### (b)

train\_pred<-predict(model,newdata = train,type = "class")

train\_ConfusionMat<-table(train$Competitive,train\_pred)

train\_acc<-sum(diag(train\_ConfusionMat))/sum(train\_ConfusionMat)

cat("The Accuracy of train dataset of the origin model is :",train\_acc,"\n")

test\_pred<-predict(model,newdata = test,type = "class")

test\_ConfusionMat<-table(test$Competitive,test\_pred)

test\_acc<-sum(diag(test\_ConfusionMat))/sum(test\_ConfusionMat)

cat("The Accuracy of test dataset of the origin model is :",test\_acc,"\n")

#### (c)

#As we know ,the model used \*OpenPrice,ClosePrice,sellerRati\* as the predictors and removed the other variable(\*Category,currency,Duration,endDay\*),that is,the predictors are the intersting variable and the other variables are not intersting.

plot(density(eBaysAuctions$sellerRating),col="red",main=paste0("sellerRating","(mean :",mean(eBaysAuctions$sellerRating),")"))

plot(density(eBaysAuctions$OpenPrice),col="red",main=paste0("OpenPrice","(mean :",mean(eBaysAuctions$OpenPrice),")"))

plot(density(eBaysAuctions$ClosePrice),col="red",main=paste0("ClosePrice","(mean :",mean(eBaysAuctions$ClosePrice),")"))

unintersting<-eBaysAuctions[,c("Category","currency","Duration","endDay","Competitive")]

#Category<-eBaysAuctions[,"Category"]

#currency<-eBaysAuctions[,"currency"]

#Duration<-eBaysAuctions[,"Duration"]

#endDay<-eBaysAuctions[,"endDay"]

ggplot(data=unintersting)+geom\_bar(aes(x=Category,fill=Competitive),position = "stack")+theme(axis.text.x = element\_text(angle = 90,face = "bold"))

ggplot(data=unintersting)+geom\_bar(aes(x=currency,fill=Competitive),position = "stack")+theme(axis.text.x = element\_text(angle = 90,face = "bold"))

ggplot(data=unintersting)+geom\_bar(aes(x=Duration,fill=Competitive),position = "stack")+theme(axis.text.x = element\_text(angle = 90,face = "bold"))

ggplot(data=unintersting)+geom\_bar(aes(x=endDay,fill=Competitive),position = "stack")+theme(axis.text.x = element\_text(angle = 90,face = "bold"))

#### (d)

# prune tree

model.2<-rpart(Competitive~OpenPrice+ClosePrice+sellerRating,data = train,method = "class")

model.2<- prune(model.2, cp =model.2$cptable[which.min(model$cptable[,"xerror"]),"CP"])

prp(model.2, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10,

box.col=ifelse(model.2$frame$var == "<leaf>", 'green', "gray"))

train\_pred<-predict(model.2,newdata = train,type = "class")

train\_ConfusionMat<-table(train$Competitive,train\_pred)

train\_acc<-sum(diag(train\_ConfusionMat))/sum(train\_ConfusionMat)

cat("The Accuracy of train dataset of the prune model is :",train\_acc,"\n")

test\_pred<-predict(model.2,newdata = test,type = "class")

test\_ConfusionMat<-table(test$Competitive,test\_pred)

test\_acc<-sum(diag(test\_ConfusionMat))/sum(test\_ConfusionMat)

cat("The Accuracy of test dataset of the prune model is :",test\_acc,"\n")

#### (e)

library(ggplot2)

ggplot(data=eBaysAuctions,aes(x=OpenPrice,y=ClosePrice,color=Competitive))+

geom\_point()+geom\_abline(intercept = 0.0, slope = 1)+

theme(legend.title = element\_blank())

#### (f)

##### confusionMatrix

cat("The train dataset confusionMatrix:","\n")

confusionMatrix(train\_pred,train$Competitive)

cat("The validation dataset confusionMatrix:","\n")

confusionMatrix(test\_pred,test$Competitive)

##### lift chart

train\_prob<-predict(model.2,newdata = train,type = "prob")

train\_pred<-prediction(train\_prob[,2],train$Competitive)

train\_perf<-performance(train\_pred,"lift","rpp")

plot(train\_perf, main="lift curve(train)", colorize=T)

test\_prob<-predict(model.2,newdata = test,type = "prob")

test\_pred<-prediction(test\_prob[,2],test$Competitive)

test\_perf<-performance(test\_pred,"lift","rpp")

plot(test\_perf, main="lift curve(validaton)", colorize=T)