Assignment 5 - Evaluating Model Performances

Matthew Yu

2017/5

1. 補強上課的範例程式，可以算出以上課講義「Lecture 8 Visualization Model Performance」的第 52 頁上，「Precision = 1.0, Recall = 0.2」兩個數值。

針對此題，我設計出能夠比較不同種方法 (共有 Classification Tree, kNN 以及 SVM) 的函數，結果除了畫出含有三條 Lift Curve 的圖以外，也將三種方法各自的 Precision 以及 Recall 計算出，列在結果。

函數的第一部份 `Pre_Re`，在讀入任一方法所得之 Confusion Matrix 後將兩個值算出。

```r
library(ROSE)
```

```
## Loaded ROSE 0.0-3
```

```r
library(rpart)
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 3.4.4
```

```
## Loading required package: gplots
```

```
## Warning: package 'gplots' was built under R version 3.4.4
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```r
library(class)
library(e1071)
data(hacide)
Pre_Re <- function(pred, table, test, response, Positive){
  if(ncol(table)==2){
    Precision <- sum(pred==test[,which(colnames(test)==response)]&pred==Positive)/(sum(pred==test[,which
    Recall <- sum(pred==test[,which(colnames(test)==response)]&pred==Positive)/(sum(pred==test[,which(co
    Answer <- c(Precision, Recall)
  }else{
    if(colnames(table)[1]!=Positive){
      Answer <- c(0,0)
    }
  }
  return(Answer)
}
```

剩餘部份 `Generic_bin_plots` 在過程中呼叫 `Pre_Re`，並司職繪圖以及列出結果。

```r
Generic_bin_plots <- function(train, test, response, Positive){
  if(length(unique(train[,which(colnames(train)==response)]))>2){
    print("This function is only meant for binary classifications")
  }
  else{
```

```
    TREEmod <- rpart(train[,which(colnames(train)==response)]~.,data=train[,-which(colnames(train)==resp
    SVMmod <- svm(train[,which(colnames(train)==response)]~.,data=train[,-which(colnames(train)==respons
    TREEpred <- predict(TREEmod, newdata=test, type="class")
    KNNpred <- knn(train, test, cl=factor(train[,which(colnames(train)==response)]))
    SVMpred <- predict(SVMmod, newdata=test, type="class")
    TREEtable <- table(test[,which(colnames(test)==response)], TREEpred)
    KNNtable <- table(test[,which(colnames(test)==response)], KNNpred)
    SVMtable <- table(test[,which(colnames(test)==response)], SVMpred)
    T.Answer <- Pre_Re(TREEpred, TREEtable, test=test, response=response, Positive=Positive)
    K.Answer <- Pre_Re(KNNpred, KNNtable, test=test, response=response, Positive=Positive)
    S.Answer <- Pre_Re(SVMpred, SVMtable, test=test, response=response, Positive=Positive)
    pre.T <- prediction(as.numeric(TREEpred),
                        as.numeric(test[,which(colnames(test)==response)]))
    pre.K <- prediction(as.numeric(KNNpred),
                        as.numeric(test[,which(colnames(test)==response)]))
    pre.S <- prediction(as.numeric(SVMpred),
                        as.numeric(test[,which(colnames(test)==response)]))
    plot(performance(pre.T, "lift", "rpp"),lwd=2, col="tomato",xlim=c(0,1),ylim=c(0,50))
    par(new=T)
    plot(performance(pre.K, "lift", "rpp"),lwd=2, col="goldenrod1",xlim=c(0,1),ylim=c(0,50))
    par(new=T)
    plot(performance(pre.S, "lift", "rpp"),lwd=2, col="darkcyan",
         main = "Lift Curve of Model",xlim=c(0,1),ylim=c(0,50))
    abline(h=1, col="black", lwd=2)
    legend("topright",legend=c("rpart", "kNN", "SVM", "Random"),
           col=c("tomato", "goldenrod1", "darkcyan","black"),lwd=2, cex=0.7)
    cat("\n","\nPrecision and recall of Tree: \n")
    cat("Precision = ",T.Answer[1]," , Recall = ",T.Answer[2])
    cat("\n","\nPrecision and recall of kNN: \n")
    cat("Precision = ",K.Answer[1]," , Recall = ",K.Answer[2])
    cat("\n","\nPrecision and recall of SVM: \n")
    cat("Precision = ",S.Answer[1]," , Recall = ",S.Answer[2],"\n")
  }
}
```
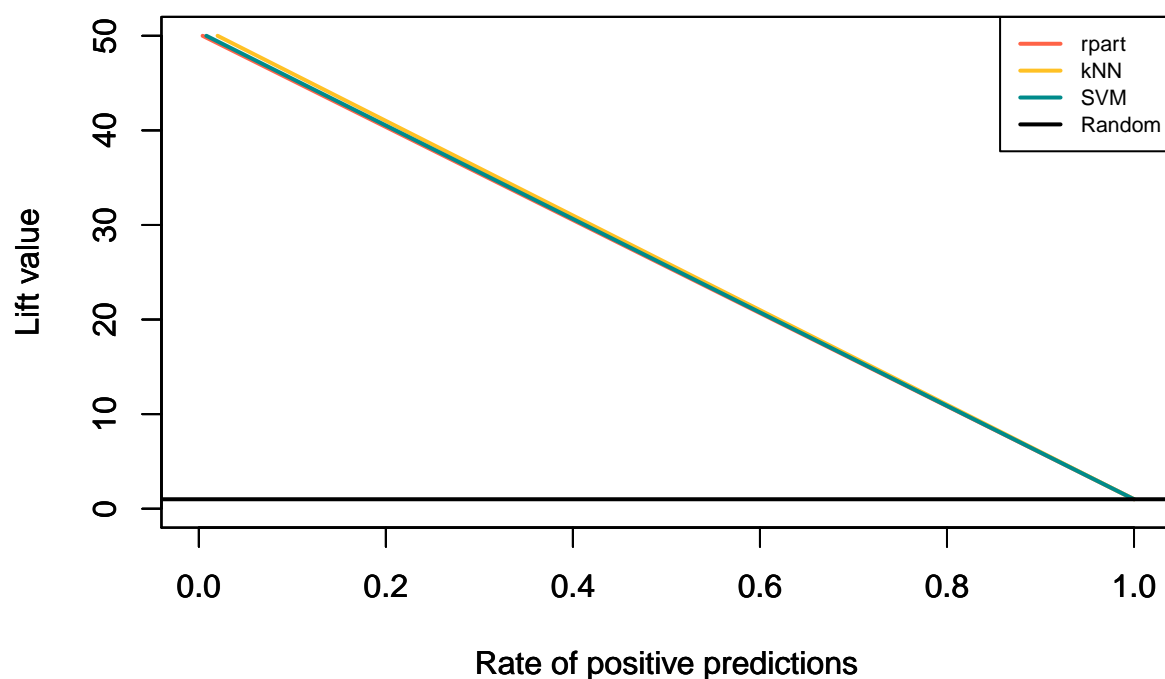
兩部份 arguments 都要求輸入目標變數之名稱 (response) 和感興趣類別 (Positive)。

以執行下列一行指令為例，可得結果報表及結果圖如下：

```
Generic_bin_plots(hacide.train, hacide.test, "cls", "1")
```

Lift Curve of Model


```
##
##
## Precision and recall of Tree:
## Precision =  1  , Recall =  0.2
##
## Precision and recall of kNN:
## Precision =  1  , Recall =  1
##
## Precision and recall of SVM:
## Precision =  1  , Recall =  0.4
```

另外值得一提的是，之所以將函數分成兩種除了外觀稍微簡潔以外，使用 table 函數計算可能會在某種方法全數分為同一類的情況時不能分辨偽陰或者偽陽。如果預測值只有一種，table 函數無法製作皆為 0 的一列或一行。