Taiwan Weighted Index Project

Jason

Monday, May 04, 2015

```
set.seed(123)
```

```
Taiwan <- read.csv("C:/Users/jason/Desktop/R/CLW_R_Learning/Jason Wang/Taiwan_weighted.csv")
```

```
library(dplyr)
```

```
Taiwan <- tbl_df(Taiwan)
#Get the change first, up or down
#Also add a new variable Month
Taiwan <- Taiwan %>%
            mutate(Change=(Close - c(Close[-1], NA))) %>%
            mutate(UpDowns=(Change > 0) * 1) %>%
            mutate(Index=seq(along=Date)) %>%
            mutate(Year=substr(Date, 1, 4)) %>%
            mutate(Month=months(as.Date(Date))) %>%
            mutate(Quarter=quarters(as.Date(Date)))
```

```
#Remove the NA
Taiwan_1 <- Taiwan[-4395, ]
```

```
#According to each year
#Divided data into trainin and testing, 8:2
train_I <- Taiwan_1 %>%
             group_by(Year) %>%
             sample_frac(0.8, replace=F) %>%
             ungroup() %>%
             select(Index)
#Transform the class
train_I <- as.matrix(train_I)
#Train & Test
train <- Taiwan_1[train_I, ]
test <- Taiwan_1[-train_I, ]
```

```
train %>%
  group_by(Year) %>%
  select(UpDowns) %>%
  table()
```

```
##       UpDowns
## Year     0   1
##   1997  53  45
##   1998 117  80
##   1999  93 100
##   2000 112  85
##   2001 100  96
```

```
##   2002 111  87
##   2003  98 101
##   2004  94 106
##   2005  97 101
##   2006  93 105
##   2007  90 104
##   2008 110  89
##   2009  71 127
##   2010  92 108
##   2011  96 102
##   2012  95 103
##   2013  85 111
##   2014  90 108
##   2015  32  30
```

```
train %>%
  group_by(Year) %>%
  summarise(n())
```

```
## Source: local data frame [19 x 2]
##
##    Year n()
## 1  1997  98
## 2  1998 197
## 3  1999 193
## 4  2000 197
## 5  2001 196
## 6  2002 198
## 7  2003 199
## 8  2004 200
## 9  2005 198
## 10 2006 198
## 11 2007 194
## 12 2008 199
## 13 2009 198
## 14 2010 200
## 15 2011 198
## 16 2012 198
## 17 2013 196
## 18 2014 198
## 19 2015  62
```

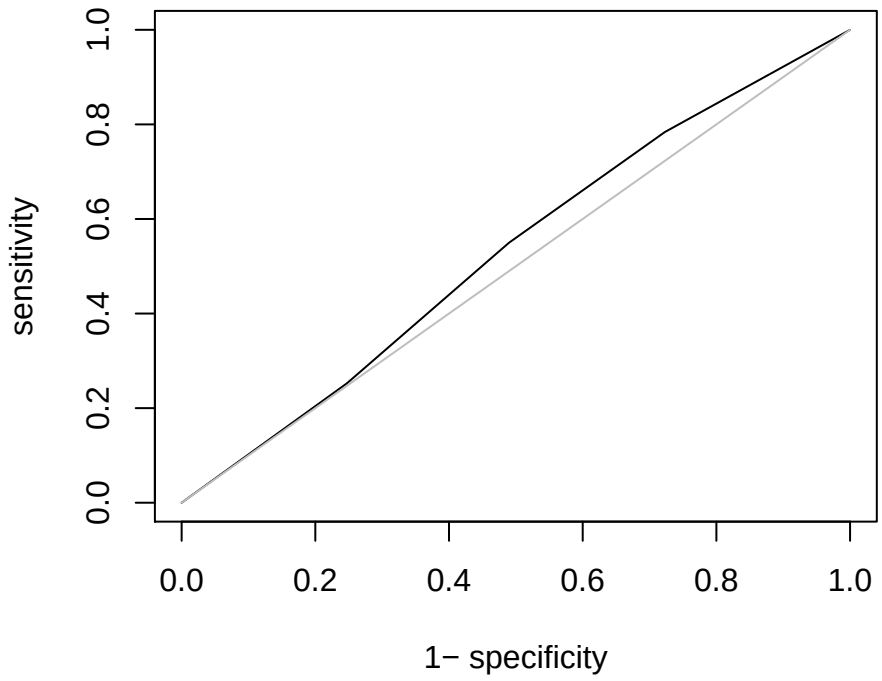
Model1 Quarter

```
#Model 1
model1 <- glm(UpDowns ~ Quarter, data = train, family=binomial)
```

```
#Prediction1
predict1 <- predict(model1, test, type="response")
table(Predict=round(predict1), True=test$UpDowns)
```

```
##         True
## Predict   0   1
##       0 120  96
##       1 313 348
```

```
library(AUC)
plot(roc(predict1, as.factor(test$UpDowns)))
```




```
auc(roc(predict1, as.factor(test$UpDowns)))
```

```
## [1] 0.5310634
```

Model2 Quarter + BU + BD

```
#Add big ups and downs
Taiwan <- Taiwan %>%
            mutate(BU=c((Change[-1] > 100)*1, NA)) %>%
            mutate(BD=c((Change[-1] < -100)*1, NA))
```

```
#Remove the NA
Taiwan_2 <- Taiwan[-4395, ]
```

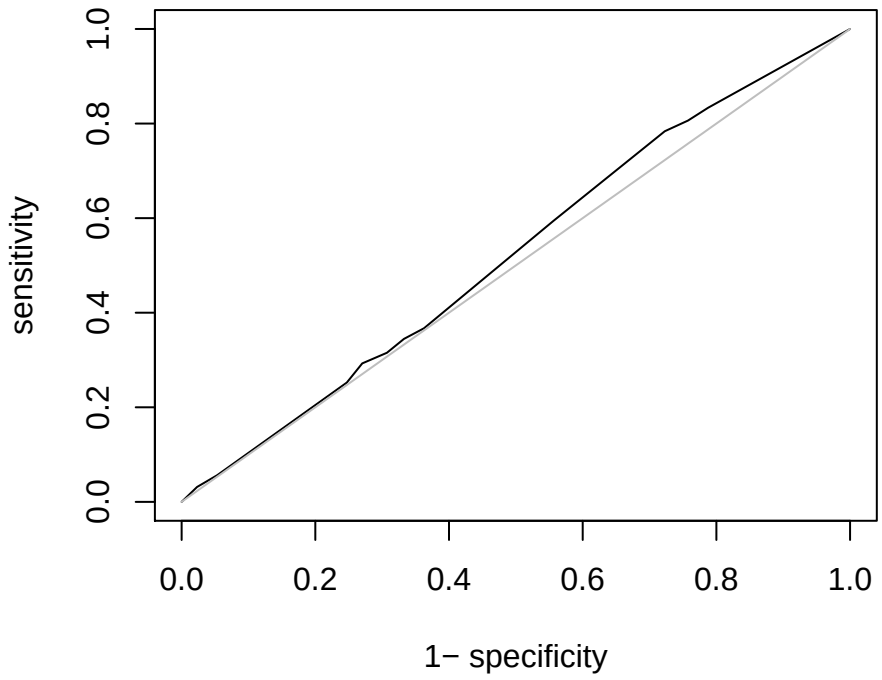
```r
#Train & Test
train <- Taiwan_2[train_I, ]
test <- Taiwan_2[-train_I, ]
```

```r
model2 <- glm(UpDowns ~ Quarter + BU + BD, data = train, family=binomial)
```

```r
#Prediction2
predict2 <- predict(model2, test, type="response")
table(Predict=round(predict2), True=test$UpDowns)
```

```
##        True
## Predict   0   1
##       0 120  96
##       1 313 348
```

```r
plot(roc(predict2, as.factor(test$UpDowns)))
```




```r
auc(roc(predict2, as.factor(test$UpDowns)))
```

```
## [1] 0.5230115
```

Model3 Month + BU + BD + SPUDLag1

```
#Add SP
SP <- read.csv("C:/Users/jason/Desktop/R/CLW_R_Learning/Jason Wang/S&P 500.csv")

SP <- tbl_df(SP)
SP <- SP %>%
            mutate(Change=(Close - c(Close[-1], NA))) %>%
            mutate(UpDowns=(Change > 0) * 1) %>%
            mutate(SPUDLag1=c(UpDowns[-1], NA))
```

```
all <- merge(Taiwan_2, SP, by="Date", all.x=TRUE)
all <- tbl_df(all)
Tai_SP <- all %>%
          select(Date, Year, Month, Close=Close.x, Change=Change.x, BU, BD,
                 UpDowns=UpDowns.x, SPUDLag1, Index) %>%
          arrange(Index)
```

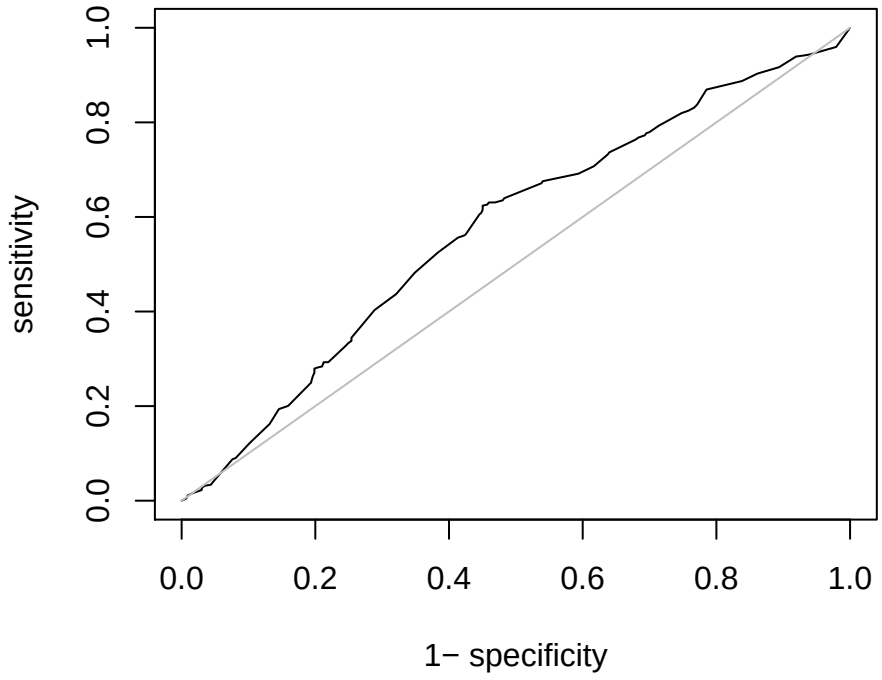
```
Tai_SP1 <- Tai_SP[-c(4393, 4394), ]
```

```
#Train & Test
train <- Tai_SP1[train_I, ]
test <- Tai_SP1[-train_I, ]
#Model 3
model3 <- glm(UpDowns ~ Month + BU + BD + SPUDLag1, data=train, family=binomial)
```

```
#Prediction3
predict3 <- predict(model3, test, type="response")
table(Predict=round(predict3), True=test$UpDowns)
```

```
##        True
## Predict   0   1
##       0 231 157
##       1 193 269
```

```
plot(roc(predict3, as.factor(test$UpDowns)))
```




```
auc(roc(predict3, as.factor(test$UpDowns)))
```

```
## [1] 0.5753568
```

Model4 LDA

```
library(MASS)
#Linear Discriminat Analysis
model4 <- lda(UpDowns ~ Month + BU + BD + SPUDLag1, data=train)
```

```
#Prediction4
predict4 <- predict(model4, test)$posterior[, 2]
```

```
## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf
```

```
## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf
```

```
## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf
```

```
## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf
```

```
## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf

## Warning in FUN(newX[, i], ...): min 中沒有無漏失的引數; 回傳 Inf
```

```
table(Predict=round(predict4), True=test$UpDowns)
```

```
##        True
## Predict   0   1
##       0 231 157
##       1 193 269
```

Model5 Quarter + BU + BD + Lag1

```
Taiwan_2 <- Taiwan_2 %>%
              mutate(Lag1=c(UpDowns[-1], NA))
```

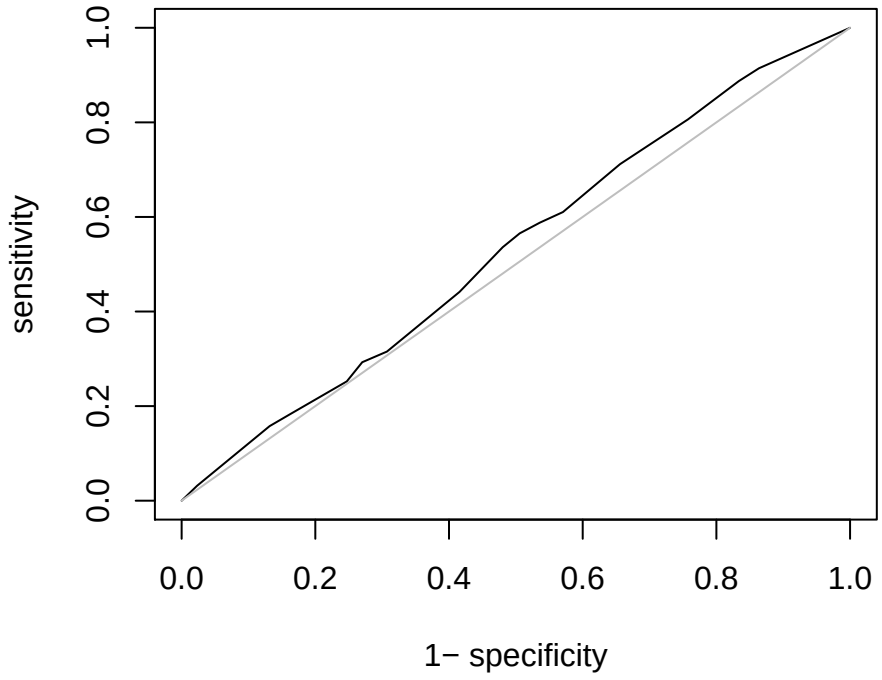
```
#Train & Test
train <- Taiwan_2[train_I, ]
test <- Taiwan_2[-train_I, ]
```

```
model5 <- glm(UpDowns ~ Quarter + BU + BD + Lag1, data = train, family=binomial)
```

```
#Prediction5
predict5 <- predict(model5, test, type="response")
table(Predict=round(predict5), True=test$UpDowns)
```

```
##        True
## Predict   0   1
##       0 201 183
##       1 232 261
```

```
plot(roc(predict5, as.factor(test$UpDowns)))
```




```
auc(roc(predict5, as.factor(test$UpDowns)))
```

```
## [1] 0.5321973
```

Model6

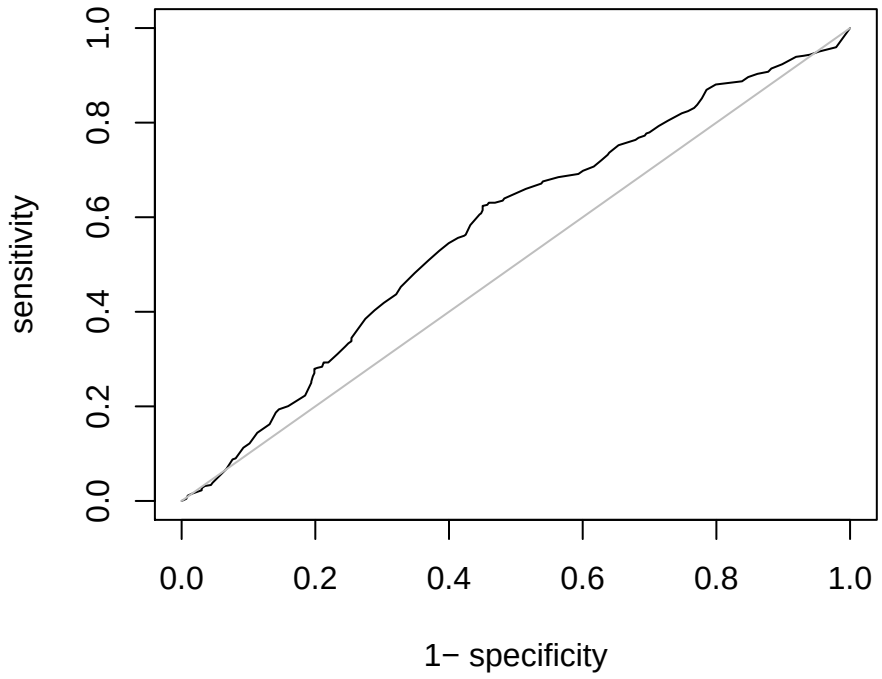
```
library(dplyr)
all <- merge(Taiwan_2, SP, by="Date", all.x=TRUE)
all <- tbl_df(all)
Tai_SP2 <- all %>%
          dplyr::select(Date, Year, Month, Close=Close.x, Change=Change.x, BU, BD,
                 UpDowns=UpDowns.x, SPUDLag1, Lag1, Index) %>%
          arrange(Index)
```

```
#Train & Test
train <- Tai_SP2[train_I, ]
test <- Tai_SP2[-train_I, ]
#Model 6
model6 <- glm(UpDowns ~ Month + BU + BD + SPUDLag1 + Lag1, data=train, family=binomial)
```

```r
#Prediction6
predict6 <- predict(model6, test, type="response")
table(Predict=round(predict6), True=test$UpDowns)
```

```
##        True
## Predict   0   1
##       0 231 157
##       1 193 269
```

```r
plot(roc(predict6, as.factor(test$UpDowns)))
```




```r
auc(roc(predict6, as.factor(test$UpDowns)))
```

```
## [1] 0.5760954
```

```r
Taiwan_3 <- Taiwan %>%
              mutate(Lag1=c(UpDowns[-1], NA)) %>%
              mutate(Lag2=c(UpDowns[-(1:2)], rep(NA, 2))) %>%
              mutate(Lag3=c(UpDowns[-(1:3)], rep(NA, 3))) %>%
              mutate(Lag4=c(UpDowns[-(1:4)], rep(NA, 4))) %>%
              mutate(Lag5=c(UpDowns[-(1:5)], rep(NA, 5)))

#Remove NA
Taiwan_3 <- Taiwan_3[1:4389, ]
```

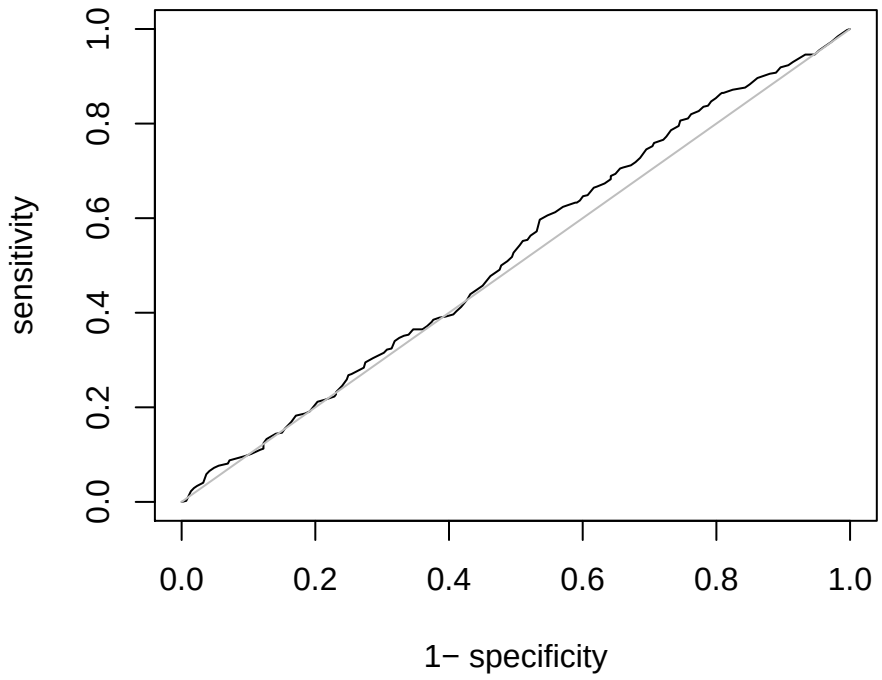
```r
#Train & Test
train <- Taiwan_3[train_I, ]
test <- Taiwan_3[-train_I, ]
```

```r
model7 <- glm(UpDowns ~ Quarter + Lag1 + Lag2 + Lag3 + Lag4 + Lag5, data = train, family=binomial)
```

```r
#Prediction7
predict7 <- predict(model7, test, type="response")
table(Predict=round(predict7), True=test$UpDowns)
```

```
##        True
## Predict   0   1
##       0 178 163
##       1 255 281
```

```r
plot(roc(predict7, as.factor(test$UpDowns)))
```




```r
auc(roc(predict7, as.factor(test$UpDowns)))
```

```
## [1] 0.5230037
```

```r
#Test
test <- read.csv("C:/Users/jason/Desktop/test.csv")
test <- rbind(test[, -6], Taiwan[c(1, 2), c(1:5)])
test <- tbl_df(test)

#Get the change first, up or down
#Also add a new variable Month
test <- test %>%
  mutate(Change=(Close - c(Close[-1], NA))) %>%
  mutate(UpDowns=(Change > 0) * 1) %>%
  mutate(Index=seq(along=Date)) %>%
  mutate(Year=substr(Date, 1, 4)) %>%
  mutate(Month=months(as.Date(Date))) %>%
  mutate(Quarter=quarters(as.Date(Date)))

test <- test %>%
  mutate(BU=c((Change[-1] > 100)*1, NA)) %>%
  mutate(BD=c((Change[-1] < -100)*1, NA))


test <- test %>%
  mutate(Lag1=c(UpDowns[-1], NA))

test <- test %>%
  mutate(Lag1=c(UpDowns[-1], NA)) %>%
  mutate(Lag2=c(UpDowns[-(1:2)], rep(NA, 2))) %>%
  mutate(Lag3=c(UpDowns[-(1:3)], rep(NA, 3))) %>%
  mutate(Lag4=c(UpDowns[-(1:4)], rep(NA, 4))) %>%
  mutate(Lag5=c(UpDowns[-(1:5)], rep(NA, 5)))
```