Forecasting Bankruptcy Rates

Chris Dong

November 28, 2017

Libaries

```
library(tseries)
library(car)
library(forecast)
library(tidyverse)
library(magrittr)
```

Loading Data

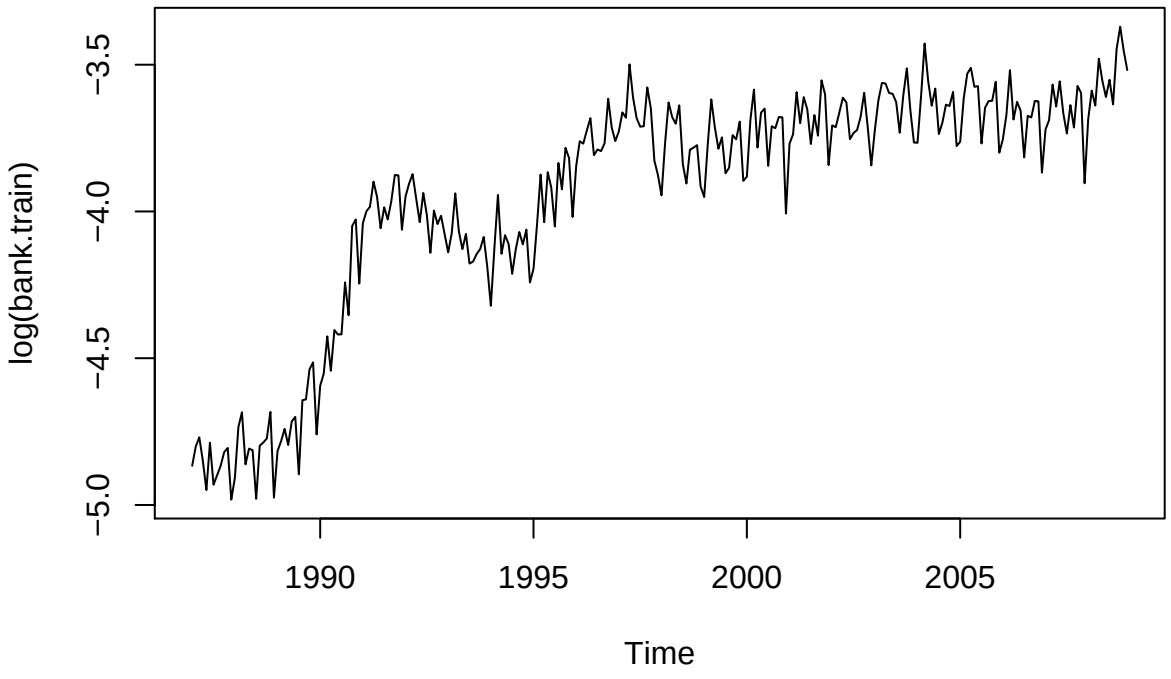
```
train <- read_csv("train.csv")
test <- read_csv("test.csv")

train %<>% na.omit()
bank <-ts(train$Bankruptcy_Rate, start = c(1987, 1), end =c(2010, 12), frequency = 12)
```

Create Training and Validation Set

```
bank.train <- window(bank, start = c(1987,1), end = c(2008,12))
bank.test <- window(bank, start = c(2009,1), end = c(2010,12))
```

```
plot(log(bank.train))
```



```
adf.test(bank.train)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  bank.train
## Dickey-Fuller = -2.0486, Lag order = 6, p-value = 0.5554
## alternative hypothesis: stationary
```

```
bank.train1 <- diff(bank.train)
```

```
adf.test(bank.train1)$p.value
```

```
## [1] 0.01
```

```
bank.train2 <- diff(bank.train1, lag = 12)
```

Trying `auto.arima` as baseline

```
automl <- arima(log(bank.train), order = c(2,0,1),
      seasonal = list(order = c(0,0,2), method = "ML"))
sqrt(mean((exp(forecast(automl, level = 95, h = 24)$mean) - bank.test)^2))
```

```
## [1] 0.004791864
```

```
result <-c()
orderlist = list()

for(i in 0:3){
  for(j in 0:3){
    for(a in 0:3){
      for(b in 0:3){

        orderlist <- c(orderlist, paste(i,j,a,b))
        bankmodel <- tryCatch({expr = arima(log(bank.train), order = c(i,1,j),
          seasonal = list(order = c(a,1,b), period = 12), method = "ML")},
          error = function(cond) {return(NA)})

        rmse <- sqrt(mean((exp(forecast(bankmodel,
                level = 95, h = 24)$mean) - bank.test)^2))
        print(paste(i,j,a,b, ":", rmse))
        ifelse(!is.na(bankmodel), result <- c(result, rmse),
               result <- c(result, NA))
  }
  }
  }
}
save(result, orderlist, file = "bank.Rmd")
```

```
load(file = "bank.RData")
```

```
names(result) <- unlist(orderlist)
head(result[order(result)], n = 25)
```

```
##     0 3 3 3     0 3 2 3     0 2 3 3     0 2 2 3     2 0 3 3     1 1 3 2
## 0.003669244 0.003723792 0.003755809 0.003819419 0.003826357 0.003836567
##     0 0 2 3     1 0 3 3     0 1 3 3     3 1 3 3     2 1 3 3     1 2 3 3
## 0.003872421 0.003883214 0.003884174 0.003887526 0.003941891 0.003986701
```

```
##     1 1 2 2     1 0 2 3     0 1 2 2     0 1 3 2     2 3 2 2     3 2 3 3
## 0.003990225 0.003991392 0.003994789 0.003996011 0.004004227 0.004007126
##     2 0 2 2     0 2 2 2     2 2 2 2     3 0 3 3     3 0 3 2     2 0 1 3
## 0.004026622 0.004040001 0.004057212 0.004077495 0.004079149 0.004080659
##     2 1 2 2
## 0.004082576
```

```
m1 <- arima(log(bank.train), order = c(0,1,3), seasonal = list(order = c(2,1,3), period = 12), method =
m2 <- arima(log(bank.train), order = c(0,1,3), seasonal = list(order = c(3,1,3), period = 12), method =
D <- -2*(m1$loglik - m2$loglik)
pval <- 1-pchisq(D,length(m2$coef) - length(m1$coef))
print(c("Test Statistic:",round(D, 4),"P-value:", round(pval, 4)))
```

```
## [1] "Test Statistic:" "0.0246"          "P-value:"        "0.8754"
```

SARIMA(0,1,3)(2,1,3) better than SARIMA(0,1,3)(3,1,3)

```
m1 <- arima(log(bank.train), order = c(0,1,2), seasonal = list(order = c(2,1,3), period = 12), method =
m2 <- arima(log(bank.train), order = c(0,1,3), seasonal = list(order = c(2,1,3), period = 12), method =
D <- -2*(m1$loglik - m2$loglik)
pval <- 1-pchisq(D,length(m2$coef) - length(m1$coef))
print(c("Test Statistic:",round(D, 4),"P-value:", round(pval, 4)))
```

```
## [1] "Test Statistic:" "4.1163"          "P-value:"        "0.0425"
```

SARIMA(0,1,3)(2,1,3) better than SARIMA(0,1,2)(2,1,3)

```
rmse <- function(logmodel) sqrt(mean((exp(forecast(logmodel, level = 95, h = 24)$mean) - bank.test)^2))
```

```
model <-  arima(log(bank.train), order = c(0,1,3), seasonal = list(order = c(2,1,3), period = 12), metho
(score <- rmse(model))
```

```
## [1] 0.003723792
```

So far, an SARIMA(0,1,3)(2,1,3) gets a RMSE of 0.0037238 when forecasting from January 2009 to December 2010.