MSAN 604: FINAL PROJECT
FORECASTING CANADIAN NATIONAL BANKRUPTCY RATES

Arpita Jena, Jose A. Rodilla, Zizhen Song, Kaya Tollas (Team 2.9)

The goal of this analysis is to model and forecast monthly bankruptcy rates for Canada. The dataset we will be working with includes monthly data on bankruptcy rates along with unemployment rate, housing price index and population. Bankruptcy rate data ranges from January 1987 to December 2010. Data on the remaining variables is also available for the period we want to predict; January 2011 to December 2012. This allows us to use the additional variables in order to aid our prediction of bankruptcy rates. Our goal, therefore, is to come up with an optimal predictive model that will allow us to forecast Canadian bankruptcy rates between January 2011 and December 2012.

MODELING APPROACHES

In order to model bankruptcy rates there are several approaches we can take. These can be divided into two main classes: univariate and multivariate approaches.

A **univariate** approach only takes into consideration the past history of the variable being modeled; it makes no use of external information. In our case, no data other than bankruptcy rates would be fed into our model. We will consider two main univariate approaches; *Box-Jenkins* and *Holt-Winters*. The most general version of the *Box-Jenkins* approach is known as *SARIMA*.
One advantage of *Holt-Winters over SARIMA* is that it does not rely on any statistical assumptions. Future observations are predicted by performing a smoothing on the previous observations in the time series.

Multivariate approaches, on the other hand, try to take advantage of external time series in order to model and forecast the time series of interest (the response time series). In our case, this would mean using any combination of the additional variables (unemployment rate, housing price index and population) in order to predict bankruptcy rates. These methods can be divided into two main types.
If we consider the external time series to be **exogenous** —that is, they influence the response but the response does not influence them— we should then employ a technique known as *SARIMAX*, which extends the *SARIMA* framework by adding the exogenous time series into the mix.
If, on the other hand, we consider all the time series variables to influence each other (multidirectional relationships, i.e., **endogenous** effects) we would use a technique known as *Vector Autoregression (VAR)*. The main difference between this approach and *SARIMAX* is that *VAR* treats all the time series equally, and attempts to model each of them simultaneously.

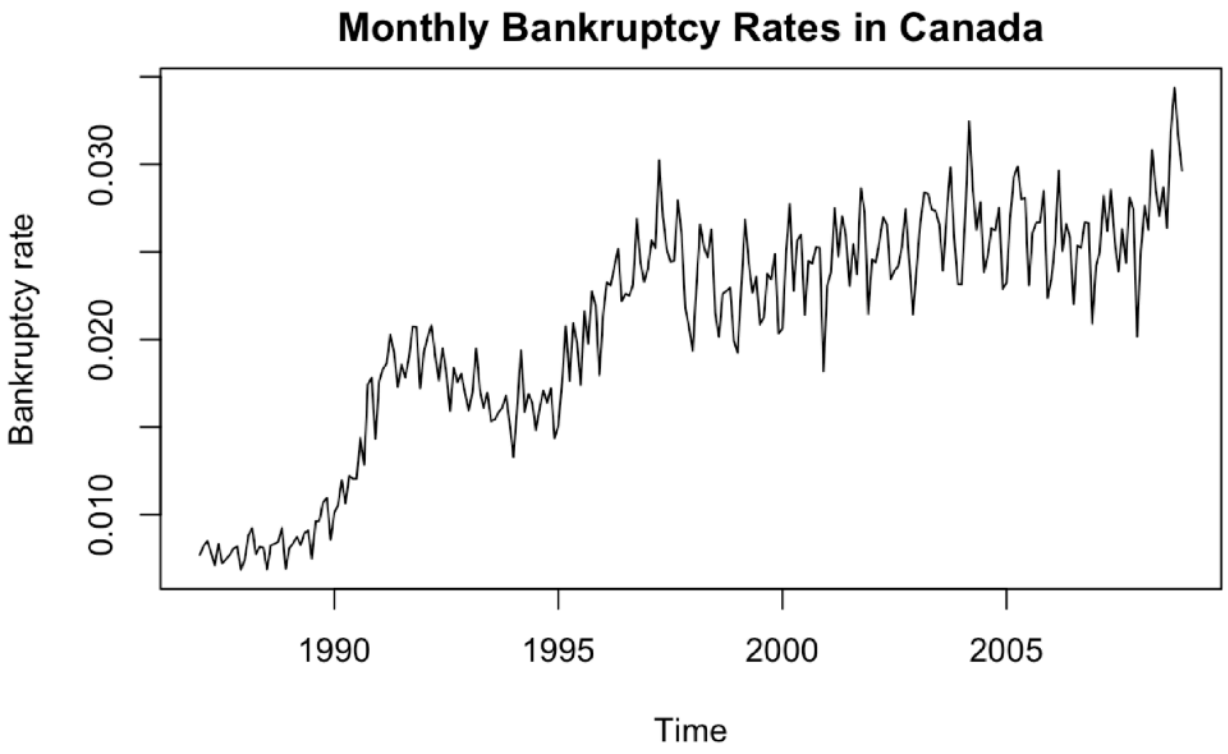
We will use all of the above mentioned methods in order to come up with our optimal model. Out of all these methods we will finally choose the one that yields the model with the highest predictive accuracy (as measured by RMSE; root mean squared error).

SPLITTING THE DATA

In order to asses the accuracy of our models we need to leave aside a portion of our data. This data must remain unseen to our models in the fitting process, and be used only as a measuring stick to compare predictive accuracy between models. In order to mimic the conditions of our actual goal (predict 24 months into the future) we have separated the last 24 observations of data as our validation set.

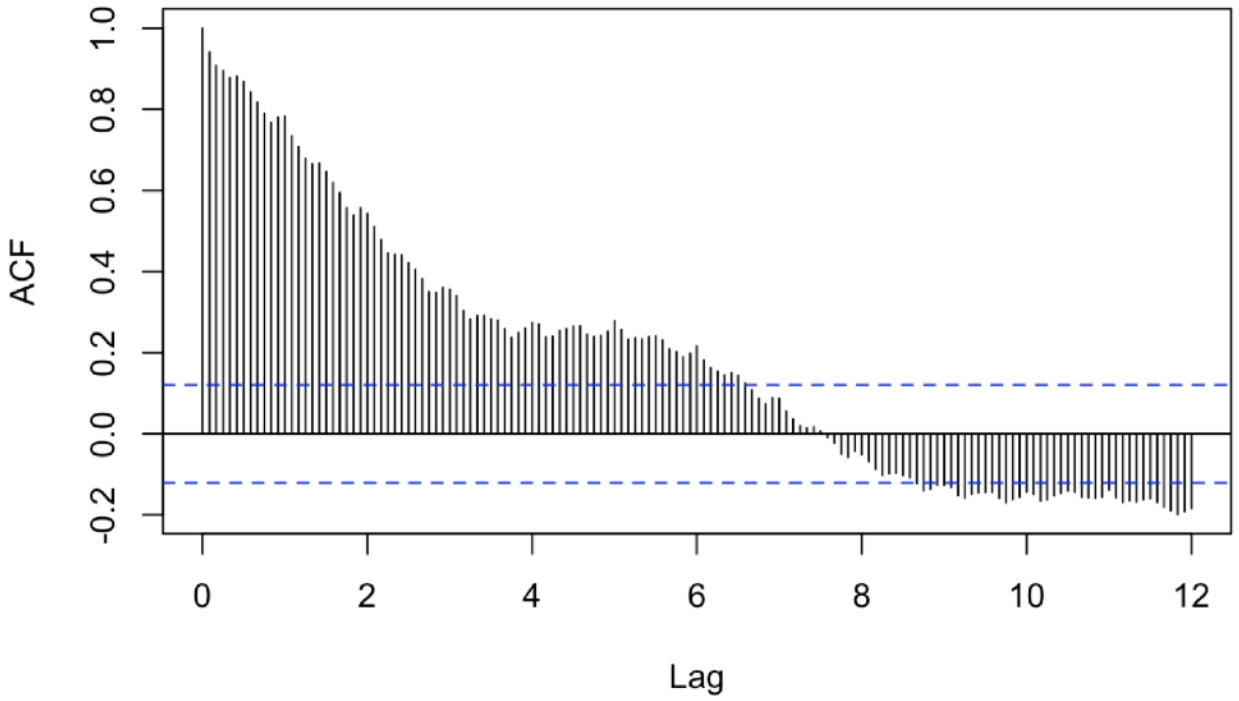
LOOKING INTO THE DATA

Before jumping into the modeling process we will take a first look into our training data.



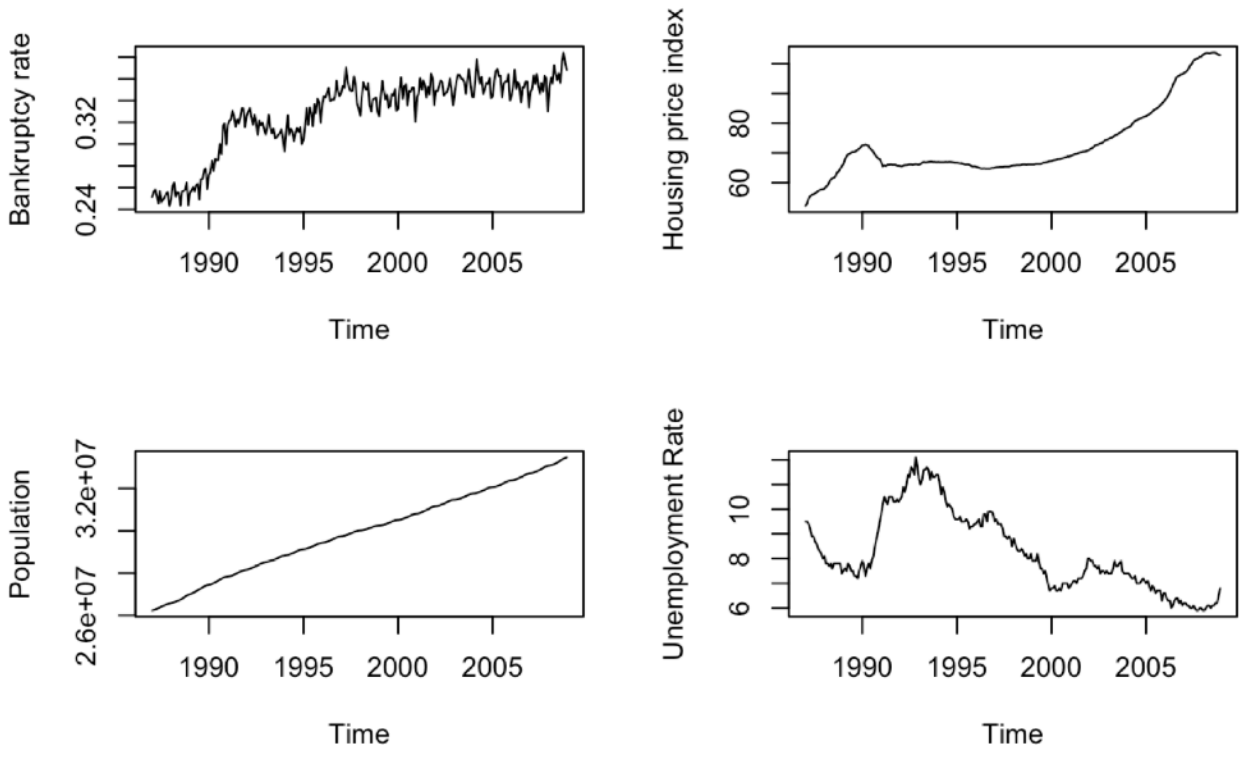
The first things to consider when looking at a time series is whether we are in presence of trend or seasonality, and whether the variability of the data seems constant (the magnitude of the ups and downs). In our case, it seems like there might be an upward trend in the bankruptcy rates. However, if we only consider data post-1995 the time series seems pretty stationary. The variability of the data also seems to increase with time.
We can also look at an ACF plot of the data to better asses the presence of trend/seasonality. An ACF plot shows the correlation between the time series and delayed copies of itself. The x-axis (the lag) represents the number of time points by which the series has been delayed. Each of the spikes represents the correlation between the time series and the series delayed by the corresponding lag.

ACF PLOT OF MONTHLY BANKRUPTCY RATES (TRAINING SET)



The slow decay in the magnitude of the spikes is a clear indication that we are in the presence of trend. We can also appreciate a periodic increase in the correlations at lags 5, 11 and so forth. This signals that we may also be in presence of seasonality. These are all things to keep in mind for the modeling stages.

ALL TIME SERIES (TRAINING SET)



MODELING PROCESS: GENERAL APPROACH AND MODEL SELECTION

For each of the modeling approaches described above we tried the following:

- Data:
  1) Use all of the training data
  2) Use only data post 1990
  3) Use only data post 1995

- Transformations:
  1) Use the data in its original state
  2) Use the logarithm of the original data (for constant variability)
  3) Use a box-cox transformation of the data (constant variability)

- Parameters: For each model we tried all possible combinations of parameters that fell within a reasonable range.

After going through all combinations of the above we chose, for each method, the models that yielded the lowest predictive rmse (as tested on the validation set). The results were as follows:

BEST MODELS FOR EACH APPROACH

Model type	Data	Transformation	Parameters	Additional regressors	RMSE
Holt-Winters	1987-2008	None	alpha=0.3, beta=0.9, gamma=0.15	—	0.00383
Sarima	1987-2008	Log	(p=1,d=0,q=0)(P=5,D=1,Q=1)[m=12]	—	0.00313
Sarima	1990-2008	Log	(p=2,d=0,q=4)(P=3,D=1,Q=5)[m=12]	—	0.00277
Elastic Net	1995-2008	Log	lambda=1.92e-06, alpha=0.5	—	0.00472
SarimaX	1987-2008	Box-Cox(^0.704)	(p=2,d=1,q=3)(P=1,D=0,Q=2)[m=12]	hpi	1.44944
VAR	1990-2008	Log	p=8	log_hpi, log_pop	0.00291
VARX	1990-2008	Log	p=8	log_hpi(en), log_pop(en), log_unemp(ex)	0.00362

The above table shows that our best candidates are the VAR and SARIMA models. We decided to take simple arithmetic averages of the highlighted models in order to see if we could come up with even better results.

VAR AND SARIMA AVERAGES

Models	RMSE
VAR, S1 and S2	0.002626013
VAR and S1	0.002599347
VAR and S2	0.002942064
S2 and S3	0.00256647

Combining the two SARIMA models yielded the best result. We thus decided to explore if we could find a weighted average between these two models that could produce even better results.

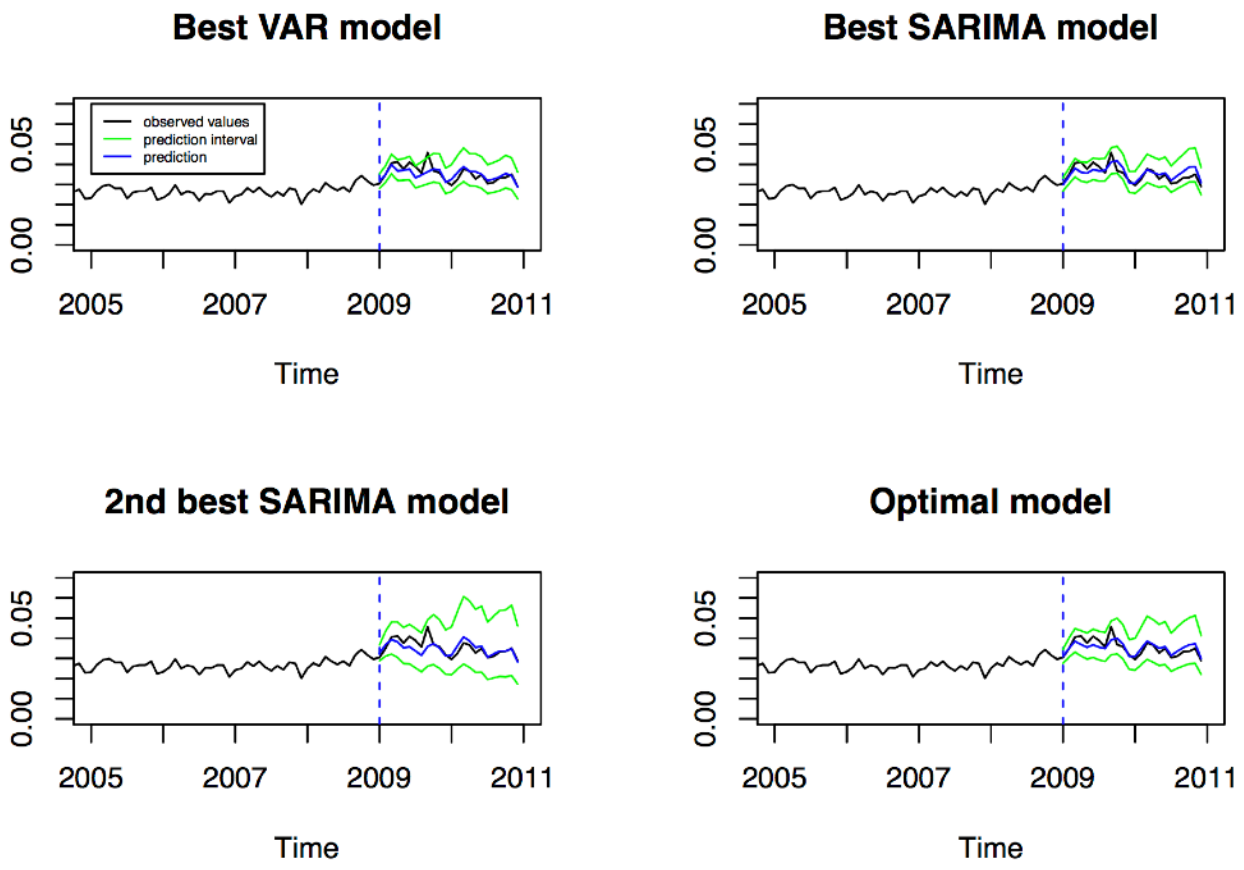
We found our optimal model for weights 0.6224 and 0.3776:

OPTIMAL MODEL

Model	RMSE
0.6224xS1 + **0.3776**xS2	0.002541158

A visual comparison of the performance of all the considered models can be seen in the following plot:

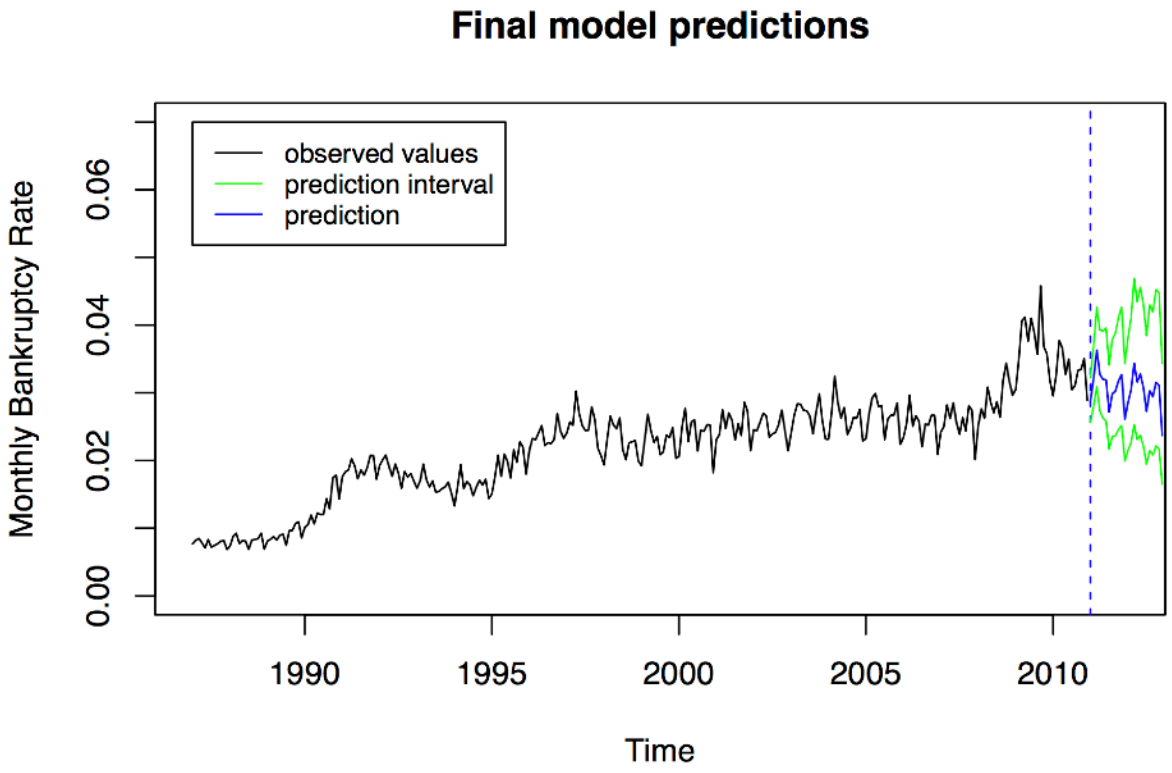
MODEL COMPARISON



Finally, we trained our optimal model on the entirety of the data (up to 2010) in order to come up with our final prediction results.

PREDICTION RESULTS

Note: *Since our focus has been on predictive accuracy, we paid less attention to the fulfillment of the underlying theoretical assumptions in our model. In practice, this means that our prediction intervals should be taken as an approximation.*



Month-Year	Prediction	Lower Bound	Upper Bound
Jan 2011	0.0287319	0.0255982	0.0322503
Feb 2011	0.0321558	0.0279796	0.0369759
Mar 2011	0.0363075	0.0309315	0.0426652
Apr 2011	0.0327056	0.0272849	0.0392597
May 2011	0.0320044	0.0262465	0.0390938
Jun 2011	0.0319183	0.0257998	0.0395684
Jul 2011	0.0271549	0.0216798	0.0340909
Aug 2011	0.0298201	0.0235396	0.0378727
Sep 2011	0.0302118	0.0236276	0.0387380
Oct 2011	0.0317459	0.0245935	0.0411021
Nov 2011	0.0326979	0.0251504	0.0426465
Dec 2011	0.0260903	0.0199260	0.0342773
Jan 2012	0.0285338	0.0214587	0.0380827
Feb 2012	0.0304102	0.0226303	0.0410405
Mar 2012	0.0343446	0.0252842	0.0468766
Apr 2012	0.0315999	0.0230948	0.0434587
May 2012	0.0328115	0.0237452	0.0455913
Jun 2012	0.0307917	0.0221253	0.0431045
Jul 2012	0.0272669	0.0194725	0.0384163
Aug 2012	0.0302732	0.0214663	0.0429702
Sep 2012	0.0294183	0.0207802	0.0419252
Oct 2012	0.0315459	0.0221422	0.0452573
Nov 2012	0.0310687	0.0217007	0.0448027
Dec 2012	0.0237123	0.0165139	0.0343000

Appendix

Arpita Jena, Jose A. Rodilla, Zizhen Song, Kaya Tollas

12/3/2017

```
#load train data
setwd('/Users/ds-lorean/Documents/USF/604_TimeSeries/Project')
data <- read.csv('train.csv', sep = ',')
```

```
#Train 1990-2008, val 2009-2010
train <- data[37:264, ]
val <- data[265:288, ]

train_bank <- ts(train$Bankruptcy_Rate, start = c(1990, 1), end = c(2008, 12), frequency = 12)
train_pop <- ts(train$Population, start = c(1990, 1), end = c(2008, 12), frequency = 12)
train_unemp <- ts(train$Unemployment_Rate, start = c(1990, 1), end = c(2008, 12), frequency = 12)
train_hpi <- ts(train$House_Price_Index, start = c(1990, 1), end = c(2008, 12), frequency = 12)
valid_bank <- ts(val$Bankruptcy_Rate, start = c(2009, 1), end = c(2010, 12), frequency = 12)
valid_pop <- ts(val$Population, start = c(2009, 1), end = c(2010, 12), frequency = 12)
valid_unemp <- ts(val$Unemployment_Rate, start = c(2009, 1), end = c(2010, 12), frequency = 12)
valid_hpi <- ts(val$House_Price_Index, start = c(2009, 1), end = c(2010, 12), frequency = 12)
```

BEST VAR Model (1990-2008, log(bankruptcy) and log(hpi) RMSE:0.002906416)

```
m <- VAR(y = data.frame(log(train_bank), log(train_hpi), log(train_pop)), p = 8, season = 12)
# prediction:
f <- exp(predict(m, n.ahead=24, ci = 0.95)$fcst$log.train_bank[,1])
f_comp <- predict(m, n.ahead=24, ci = 0.95)
f_comp <- exp(f_comp$fcst$log.train_bank.)
f_lower <- f_comp[,2]
f_upper <- f_comp[,3]
cat ("RMSE =", sqrt(mean((f - valid_bank )^2)), "\n" )
```

```
## RMSE = 0.002906416
```

for data 1990-2008: best model is SARIMA(2,0,4)(3,1,5)[12] RMSE:0.002772531

```
m2 <- arima(log(train_bank), order = c(2,0,4),seasonal = list(order = c(3,1,5), period = 12), method =
f2 <- exp(forecast(m2, h = 24, level=c(95))$mean)
f2_comp <- forecast(m2, h = 24, level=c(95))
f2_lower <- exp(f2_comp$lower)
f2_upper <- exp(f2_comp$upper)
cat ("RMSE =", sqrt(mean((f2 - valid_bank )^2)), "\n" )
```

```
## RMSE = 0.002772531
```

for data 1987-2008: best model is SARIMA(1,0,0)(5,1,1)[12] RMSE:0.003130299

```
train <- data[1:264, ]
val <- data[265:288, ]

train_bank <- ts(train$Bankruptcy_Rate, start = c(1987, 1), end = c(2008, 12), frequency = 12)
train_pop <- ts(train$Population, start = c(1987, 1), end = c(2008, 12), frequency = 12)
train_unemp <- ts(train$Unemployment_Rate, start = c(1987, 1), end = c(2008, 12), frequency = 12)
train_hpi <- ts(train$House_Price_Index, start = c(1987, 1), end = c(2008, 12), frequency = 12)
```

```
m3 <- arima(log(train_bank), order = c(1,0,0),seasonal = list(order = c(5,1,1), period = 12), method =
f3 <- exp(forecast(m3, h = 24, level=c(95))$mean)
f3_comp <- forecast(m3, h = 24, level=c(95))
f3_lower <- exp(f3_comp$lower)
f3_upper <- exp(f3_comp$upper)
cat ("RMSE =", sqrt(mean((f3 - valid_bank )^2)), "\n" )
```

```
## RMSE = 0.003130299
```

Find best weighted average:

```
for (step in seq(0.05, 0.95, 0.05) ){
  pred <- step*f2 + (1-step)*f3
  cat ("weight=", step,"RMSE =", sqrt(mean((pred - valid_bank )^2)), "\n" )
}
```

```
## weight= 0.05 RMSE = 0.003046887
## weight= 0.1 RMSE = 0.0029684
## weight= 0.15 RMSE = 0.002895236
## weight= 0.2 RMSE = 0.00282781
## weight= 0.25 RMSE = 0.002766541
## weight= 0.3 RMSE = 0.002711846
## weight= 0.35 RMSE = 0.002664131
## weight= 0.4 RMSE = 0.002623775
## weight= 0.45 RMSE = 0.002591123
## weight= 0.5 RMSE = 0.00256647
## weight= 0.55 RMSE = 0.002550046
## weight= 0.6 RMSE = 0.002542012
## weight= 0.65 RMSE = 0.002542448
## weight= 0.7 RMSE = 0.002551347
## weight= 0.75 RMSE = 0.002568624
## weight= 0.8 RMSE = 0.00259411
## weight= 0.85 RMSE = 0.002627566
## weight= 0.9 RMSE = 0.002668693
## weight= 0.95 RMSE = 0.002717143
```

Look between weights 0.55 and 0.65

```
rmses <- c()
weight <- c()
for (step in seq(0.55, 0.65, 0.0001) ){
  pred <- step*f2 + (1-step)*f3
  weight <- c(weight, step)
  rmse <- sqrt(mean((pred - valid_bank )^2))
  rmses <- c(rmses, rmse)
}

index <- which(rmses == min(rmses))
#optimal weight
ow <- weight[index]
ow
```

```
## [1] 0.6224
```

```
rmses[index]
```

```
## [1] 0.002541158
```

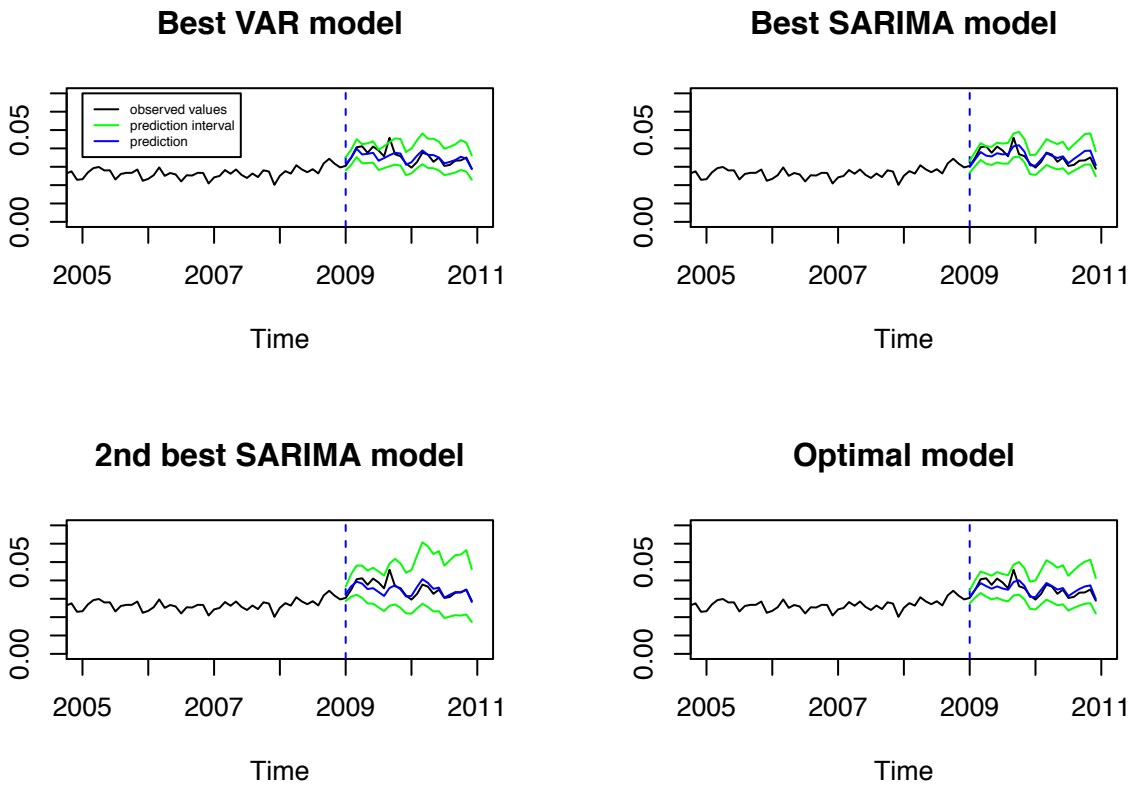
```r
#Optimal model
om <- 0.6224*f2 + 0.3776*f3
om_lower <- 0.6224*f2_lower + 0.3776*f3_lower
om_upper <- 0.6224*f2_upper + 0.3776*f3_upper
cat ("RMSE =", sqrt(mean((om - valid_bank )^2)), "\n" )
```

```
## RMSE = 0.002541158
```

Visual comparison of the models

```r
train <- data[1:288, ]
par(mfrow=c(2,2))
#plot m
plot(ts(train[,4], start = c(1987, 1), frequency = 12), xlim=c(2005,2011), ylim=c(0,0.07), main = "Best
# adding a vertical line at the point where prediction starts
abline(v=2009,col='blue',lty=2)
# plotting the predict
points(ts(f, start = c(2009, 1), frequency = 12),type='l',col='blue')
# plotting lower limit of the prediction interval
points(ts(f_lower, start = c(2009, 1), frequency = 12),type='l', col='green')
# plotting upper limit of the  prediction interval
points(ts(f_upper, start = c(2009, 1), frequency = 12),type='l', col='green')
legend(2005, 0.07, legend =
         c("observed values", "prediction interval", "prediction"),
       col = c("black", "green", "blue"), lty = 1, cex = 0.5)

#plot m2
plot(ts(train[,4], start = c(1987, 1), frequency = 12), xlim=c(2005,2011), ylim=c(0,0.07), main = "Best
# adding a vertical line at the point where prediction starts
abline(v=2009,col='blue',lty=2)
# plotting the predict
points(f2,type='l',col='blue')
# plotting lower limit of the prediction interval
points(f2_lower,type='l', col='green')
# plotting upper limit of the  prediction interval
points(f2_upper,type='l', col='green')
legend(1987, 0.07, legend =
         c("observed values", "prediction interval", "prediction"),
       col = c("black", "green", "blue"), lty = 1, cex = 0.5)

#plot m3
plot(ts(train[,4], start = c(1987, 1), frequency = 12), xlim=c(2005,2011), ylim=c(0,0.07), main = "2nd
# adding a vertical line at the point where prediction starts
abline(v=2009,col='blue',lty=2)
# plotting the predict
points(f3,type='l',col='blue')
# plotting lower limit of the prediction interval
points(f3_lower,type='l', col='green')
# plotting upper limit of the  prediction interval
points(f3_upper,type='l', col='green')
legend(1987, 0.07, legend =
         c("observed values", "prediction interval", "prediction"),
       col = c("black", "green", "blue"), lty = 1, cex = 0.5)

#plot Optimal Model
```

```r
plot(ts(train[,4], start = c(1987, 1), frequency = 12), xlim=c(2005,2011), ylim=c(0,0.07), main = "Optim
# adding a vertical line at the point where prediction starts
abline(v=2009,col='blue',lty=2)
# plotting the predict
points(om,type='l',col='blue')
# plotting lower limit of the prediction interval
points(om_lower,type='l', col='green')
# plotting upper limit of the  prediction interval
points(om_upper,type='l', col='green')
legend(1987, 0.07, legend =
         c("observed values", "prediction interval", "prediction"),
       col = c("black", "green", "blue"), lty = 1, cex = 0.5)
```



Fitting the optimal model up to 2010

m2 up to 2010

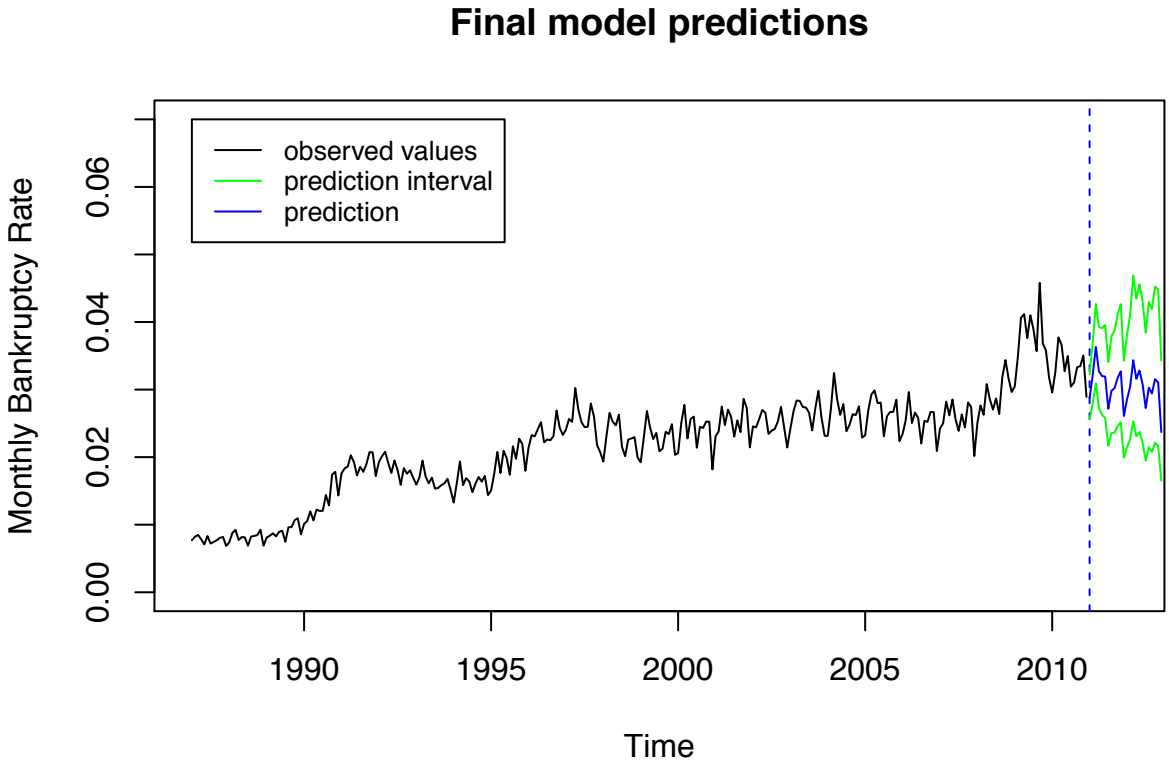
```r
train <- data[37:288, ]
train_bank <- ts(train$Bankruptcy_Rate, start = c(1990, 1), end = c(2010, 12), frequency = 12)
train_pop <- ts(train$Population, start = c(1990, 1), end = c(2010, 12), frequency = 12)
train_unemp <- ts(train$Unemployment_Rate, start = c(1990, 1), end = c(2010, 12), frequency = 12)
train_hpi <- ts(train$House_Price_Index, start = c(1990, 1), end = c(2010, 12), frequency = 12)

m2 <- arima(log(train_bank), order = c(2,0,4),seasonal = list(order = c(3,1,5), period = 12), method =
f2 <- exp(forecast(m2, h = 24, level=c(95))$mean)
f2_complete <- forecast(m2, h = 24, level=c(95))
# cat ("RMSE =", sqrt(mean((f2 - valid_bank )^2)), "\n" )
```

m3 up to 2010

```r
train <- data[1:288, ]
train_bank <- ts(train$Bankruptcy_Rate, start = c(1987, 1), end = c(2010, 12), frequency = 12)
train_pop <- ts(train$Population, start = c(1987, 1), end = c(2010, 12), frequency = 12)
train_unemp <- ts(train$Unemployment_Rate, start = c(1987, 1), end = c(2010, 12), frequency = 12)
train_hpi <- ts(train$House_Price_Index, start = c(1987, 1), end = c(2010, 12), frequency = 12)

m3 <- arima(log(train_bank), order = c(1,0,0),seasonal = list(order = c(5,1,1), period = 12), method =
f3 <- exp(forecast(m3, h = 24, level=c(95))$mean)
f3_complete <- forecast(m3, h = 24, level=c(95))
# cat ("RMSE =", sqrt(mean((f3 - valid_bank )^2)), "\n" )
```

```r
#Final predictions
fp <- 0.6224*f2 + 0.3776*f3
# cat ("RMSE =", sqrt(mean((om - valid_bank )^2)), "\n" )
```

```r
#Plot of final predictions
fp_lower <- 0.6224*exp(f2_complete$lower) + 0.3776*exp(f3_complete$lower)

fp_upper <- 0.6224*exp(f2_complete$upper) + 0.3776*exp(f3_complete$upper)

par(mfrow=c(1,1))
plot(ts(train[,4], start = c(1987, 1), frequency = 12),
     xlim=c(1987,2012), ylim=c(0,0.07),
     main = "Final model predictions", ylab="Monthly Bankruptcy Rate") #plotting the data
# adding a vertical line at the point where prediction starts
abline(v=2011,col='blue',lty=2)
# plotting the predict
points(fp,type='l',col='blue')
# plotting lower limit of the prediction interval
points(fp_lower,type='l', col='green')
# plotting upper limit of the  prediction interval
points(fp_upper,type='l', col='green')
legend(1987, 0.07, legend =
         c("observed values", "prediction interval", "prediction"),
       col = c("black", "green", "blue"), lty = 1, cex = 0.8)
```


Final model predictions
observed values
prediction interval
prediction
Monthly Bankruptcy Rate
0.00
0.02
0.04
0.06
1990
1995
2000
2005
2010
Time

Predictions table

```
test <- read.csv('test.csv', sep = ',')
test <- ts(test, start = c(2011,1), frequency = 12)
test_ <- window(test, start=c(2011,1), end=c(2012,12))
t <- as.numeric(time(test_))
t2 <- zoo::yearmon(t)

table <- data.frame(t2, as.numeric(fp), as.numeric(fp_lower), as.numeric(fp_upper))
colnames(table) <- c("Month-Year","Prediction", "Lower Bound", "Upper Bound")
kable(table)
```

Month-Year	Prediction	Lower Bound	Upper Bound
Jan 2011	0.0287319	0.0255982	0.0322503
Feb 2011	0.0321558	0.0279796	0.0369759
Mar 2011	0.0363075	0.0309315	0.0426652
Apr 2011	0.0327056	0.0272849	0.0392597
May 2011	0.0320044	0.0262465	0.0390938
Jun 2011	0.0319183	0.0257998	0.0395684
Jul 2011	0.0271549	0.0216798	0.0340909
Aug 2011	0.0298201	0.0235396	0.0378727
Sep 2011	0.0302118	0.0236276	0.0387380
Oct 2011	0.0317459	0.0245935	0.0411021
Nov 2011	0.0326979	0.0251504	0.0426465
Dec 2011	0.0260903	0.0199260	0.0342773
Jan 2012	0.0285338	0.0214587	0.0380827
Feb 2012	0.0304102	0.0226303	0.0410405
Mar 2012	0.0343446	0.0252842	0.0468766
Apr 2012	0.0315999	0.0230948	0.0434587
May 2012	0.0328115	0.0237452	0.0455913
Jun 2012	0.0307917	0.0221253	0.0431045
Jul 2012	0.0272669	0.0194725	0.0384163
Aug 2012	0.0302732	0.0214663	0.0429702
Sep 2012	0.0294183	0.0207802	0.0419252
Oct 2012	0.0315459	0.0221422	0.0452573
Nov 2012	0.0310687	0.0217007	0.0448027
Dec 2012	0.0237123	0.0165139	0.0343000

```
#Write predictions to text file
write.table(as.numeric(fp), "/Users/ds-lorean/Documents/USF/604_TimeSeries/Project/Team2.9.txt")
```