PROJECT PROPOSAL

PROJECT DEFINITION: Weekly initial claims for unemployment in the US and google search data

OBJECTIVE: The objective is to estimate the rate of weekly initial claims for unemployment by google search keywords.

METHODOLOGY: For this project I used Bayesian structural time series to TOFIT THE MODEL. structural time series models WAS USED to show how Google search data can be used to improve short term forecasts of economic time series.

DATA: The federal reserve economic data set was obtained from economic research division of Federal Reserve Bank of St Louis, Link: https://fred.stlouisfed.org .

The data consist of the weekly initial claims for unemployment insurance in the US, as reported by the US Federal Reserve. For economic decisions based on these and similar numbers, it would help to have an early forecast of the current week's number as of the close of the week.

ANALYSIS: Bayesian structural time series METHOD WAS USED to fit time series models. Structural time series models are useful because they are flexible and modular.

For economic decisions based on these and similar numbers, it would help to have an early forecast of the current week's number as of the close of the week.

Methodology: The data was divided in to two parts (train, test). In the first model (Model 1), I tried to fit a bsts model with just the trend and seasonal components on the weekly claims without other components. Subsequently, I used to predict method to predict future the next 52 time points.

After that, test data was used for validation of the prediction. Finally, regression components (michigan unemployment, military bah, pennsylvania unemployment, unemployment offices, unemployment filing, pay chart) were added to the model to observe whether Google search data to improve the forecast.

MODEL 1

```
> library(readxl)
> iclaims <- read_excel("C:/Users/MFY/Desktop/data inc project/Bayesian Struc
tured Time Series Data.xlsx",
+     sheet = "Train")
> View(iclaims)
> library("bsts")
> data(iclaims)
> ss <- AddLocalLinearTrend(list(), iclaims$ICNSA)
> ss <- AddSeasonal(ss, iclaims$ICNSA, nseasons = 52)
> model1 <- bsts(iclaims$ICNSA,
+                state.specification = ss,
+                niter = 1000)
> plot(model1)
```

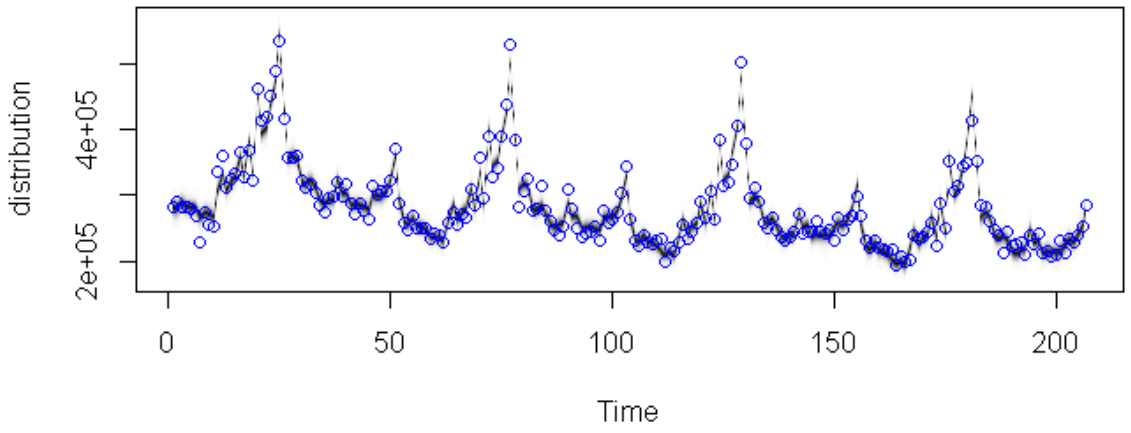

Figure 1: Distribution of train data

```
> plot(model1, "components")
```

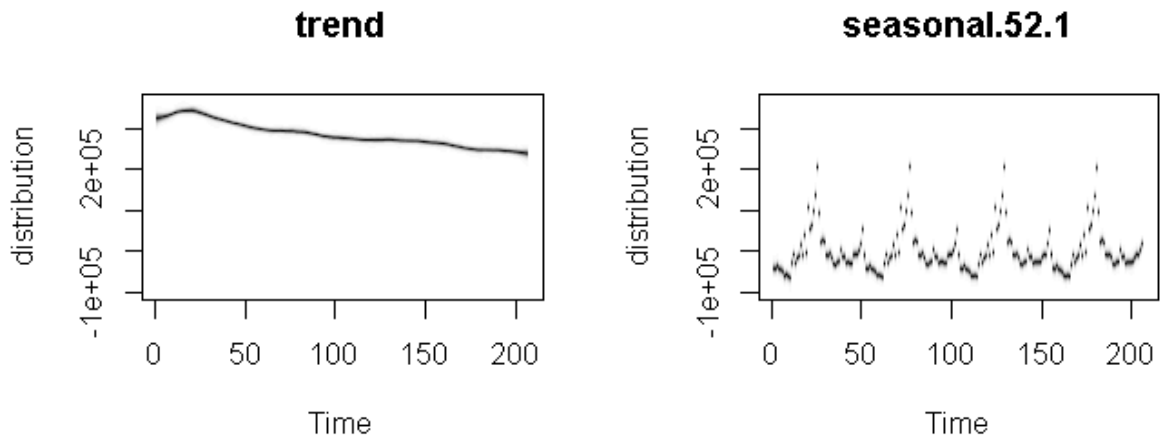

Figure 2: Trend and seasonality.

```
> pred1 <- predict(model1, horizon = 52)
> plot(pred1, plot.original = 156)
```

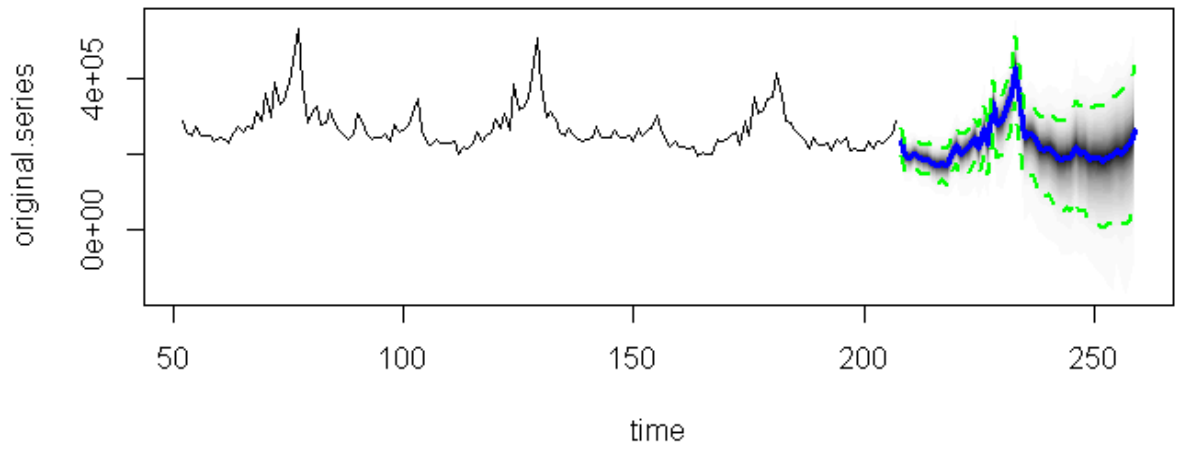

Figure 3: Predictive distribution for the next 52 weeks of initial claims.

Model 2

```
library(bsts)
> data(iclaims)
> ss <- AddLocalLinearTrend(list(), iclaims$ICNSA)
> ss <- AddSeasonal(ss, iclaims$ICNSA, nseasons = 52)
> model2 <- bsts(ICNSA ~ .,
+                state.specification = ss,
+                niter = 1000,
+                data = iclaims)
plot(model2)
```

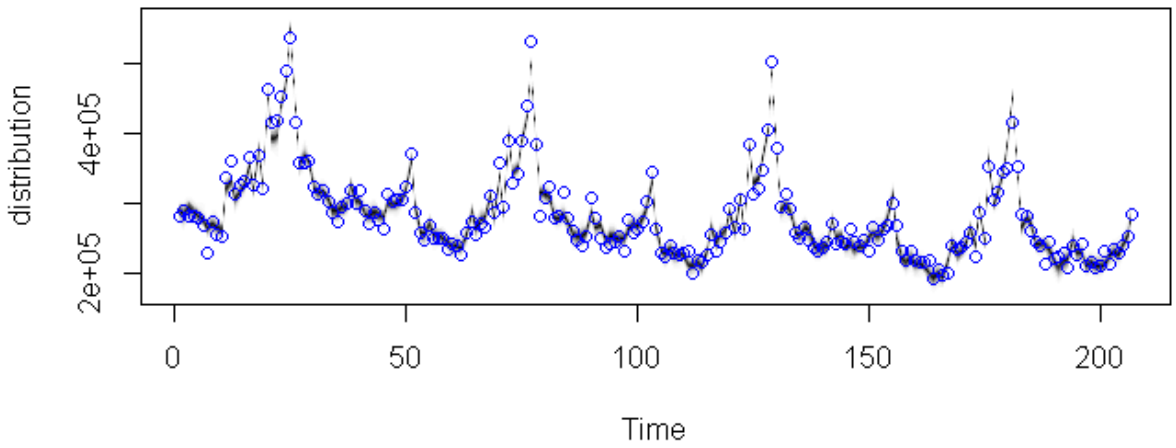

Figure 4: Distribution of the data with regression components

```
plot(model2,"comp")
```

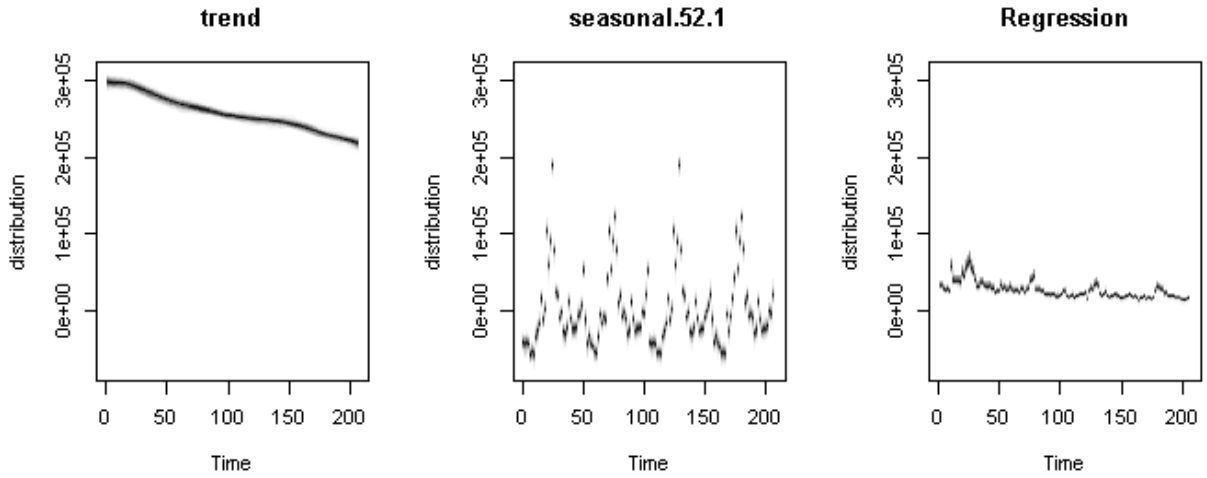

Figure 5: Contribution of each state component to the initial claims data, assuming a regression component with default prior. Compare to Figure 2.

In this part I made another prediction to compare the predicted values and the actual values.

```
library(readxl)
> iclaimstest <- read_excel("C:/Users/MFY/Desktop/data inc project/Bayesian S
tructured Time Series Data.xlsx",
+     sheet = "Test")
> View(iclaimstest)
> newdata<-iclaimstest
> pred2 <- predict(bsts.model2,
+                  newdata=newdata)
pred2 <- predict(model2,
+                  newdata=newdata)
> plot(model2)
```

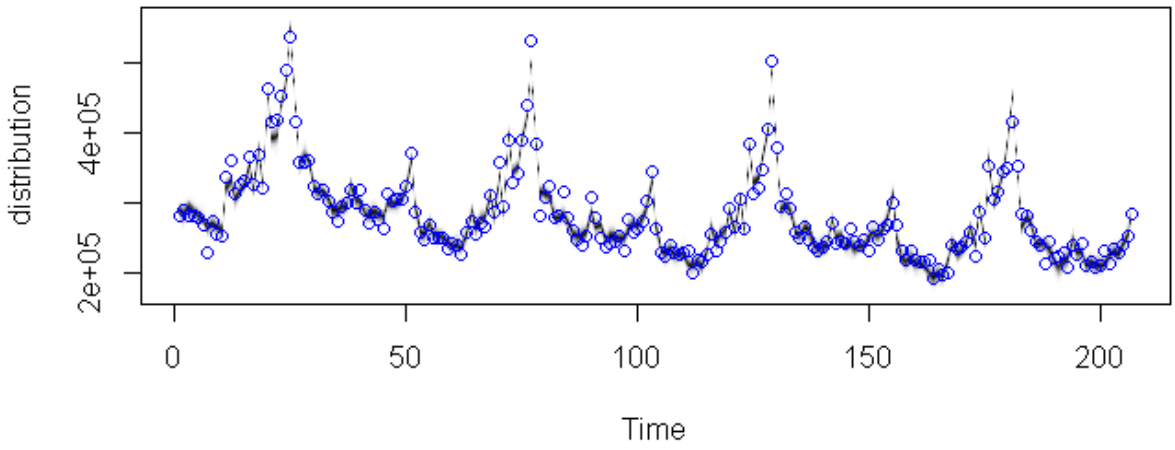

Figure 6: Distribution of test data

```
> plot(pred2,
+      plot.original=156)
```

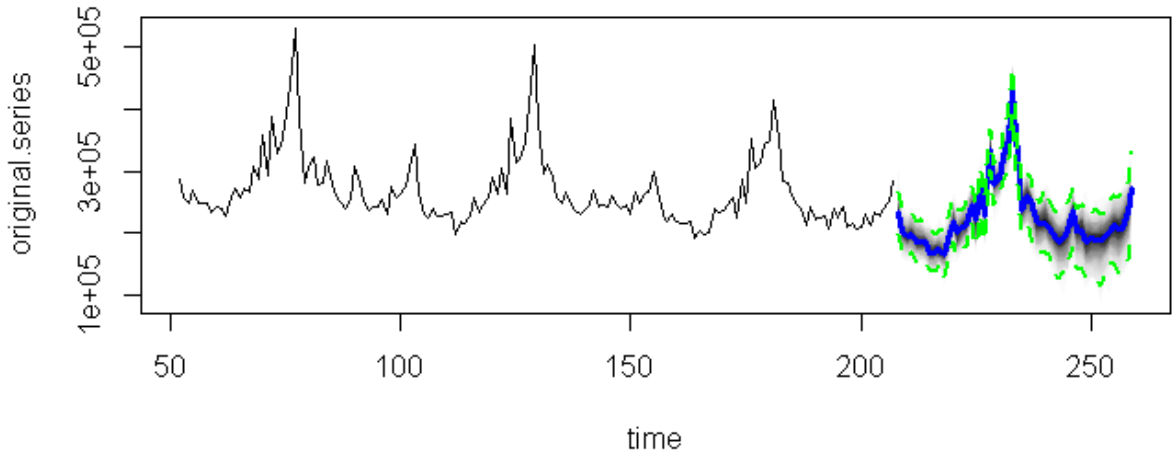

Figure 7: Predictive distribution for the next 52 weeks of initial claims with regression coefficient.

```
> plot(model2, "coef")
```

```
> print(pred2)
```

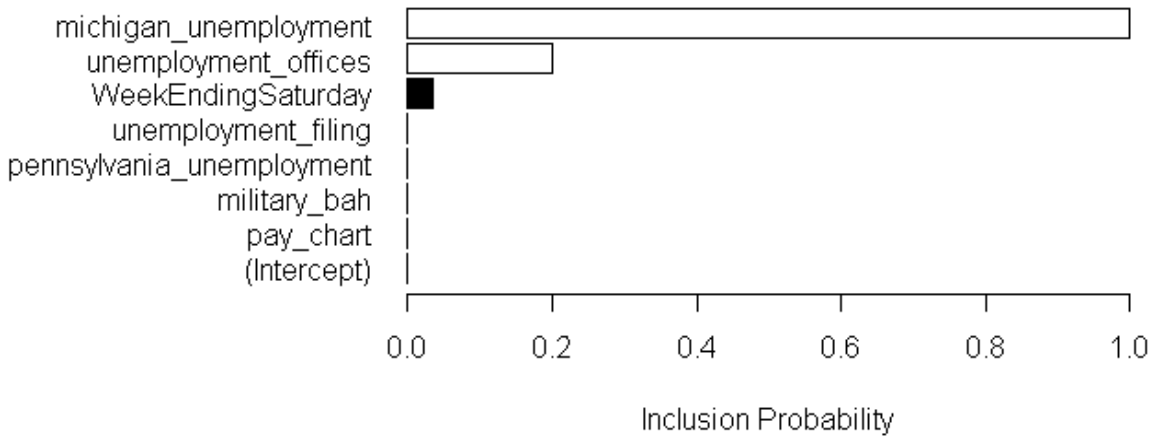

Figure 8: Inclusion probabilities for predictors in the "initial claims"

In Figure 5. The search term "michigan_unemployment" shows up with high probability in model

Conclusion

The comparison of the two-model with the actual values revealed that, there is an improvement in the model accuracy. For the model1 the mean absolute percentage errors (MAPE) scores were found 9.5%. Subsequently, MAPE2 scores for model2 was 5.7%. As a result, model2 was improved by 3.8% which indicates 40% percent improvement.

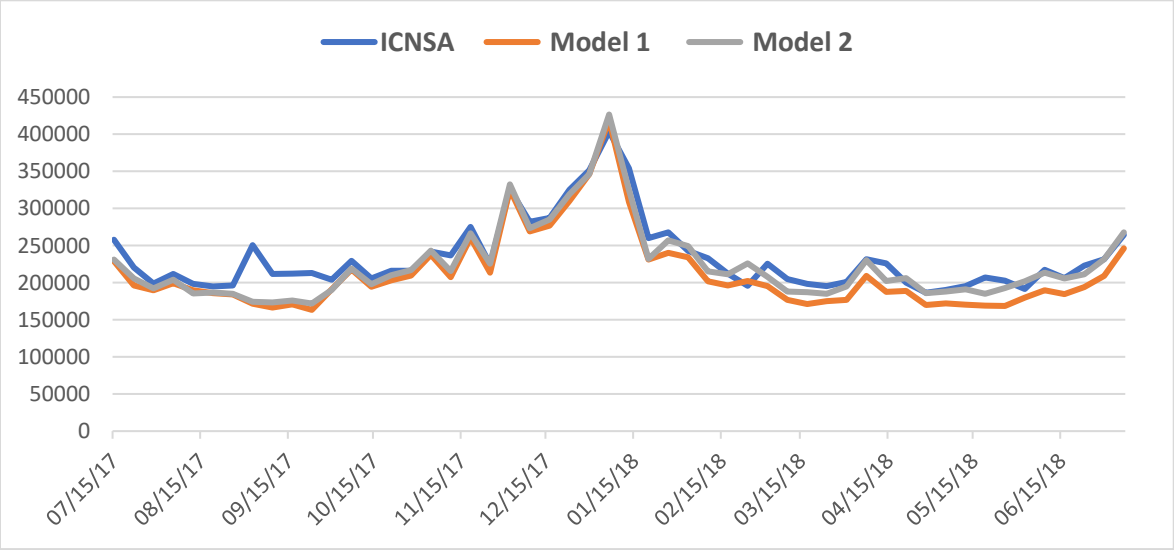

Figure 9: Initial claims, model1 and model2 comparison.