Ajjit Narayanan

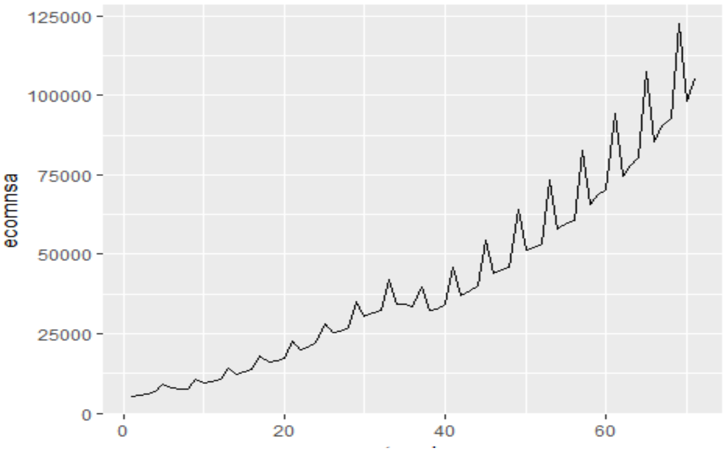
ECON 104 PS 3

Introduction

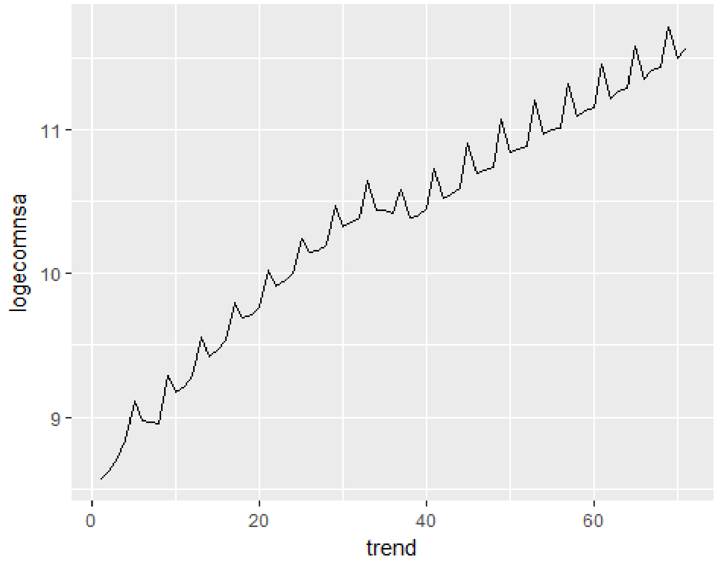
The purpose of this paper is to build a time series regression for U.S. quarterly e-commerce retail sales, NSA in millions of dollars. The data is called series ECOMNSA and is available via FRED.

Exploratory Data Analysis

The first thing to do is to plot the ecomnsa scores over time.



We can see that there is a clear positive upward trend associated with ecomnsa. In short e-commerce sales go up year by year. It is also notable that there is a clear quarterly pattern to the data. And that spike seems to be growing proportionally to the size of ecomnsa itself. To correct for the growing magnitude spike, we look at the log of ecomnsa. That plot is shown below



We see that seasonality is still present, but the spike does not grow as the value of log ecomnsa linearly increases. So for our dependent variable, we will be using log ecomnsa instead of ecomnsa as the seasonal fluctuations are roughly constant in size over time. There also does seem to be a roughly quadratic trend in the data, at least for the first 35 or so observations. There also seem to be some breakpoints around t=35 to t=40, around the time of the recession of 2008

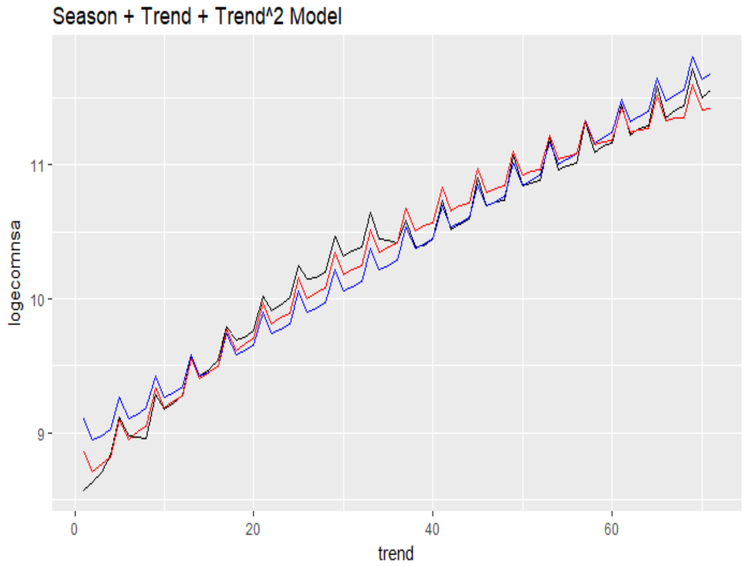
To further get a sense of our data, we provide some summary statistics

	Median	Mean	Var
logecomnsa	10.442	10.334	0.69834
ecomnsa	34269.12	41751	2.010407

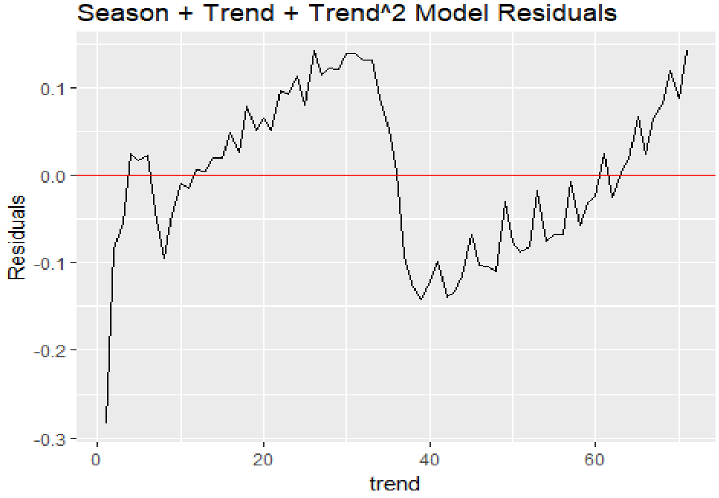
So it is clear that our data has a trend component, a seasonal component, and possible breakpoints

Trend + Seasonality

To break down the effects of trend and seasonality, we generate dummy variables for Q2, Q3 and Q4. We keep the constant as a baseline constant for Q1. For our first model, we include $trend + Q1 + Q2 + Q3 + intercept$. In our second model we include all the same intercepts with an addition of a $trend^2$ term. The fitted values from those models are plotted below. The red line is the model with just $trend$, the blue line is the model with $trend^2$ and the black line is the actual value.



While the model fits are enticing, the models themselves seems to systematically under forecast in the beginning and systematically over forecast in the end. This seems to imply some sort of auto correlation among the model residuals. In order to confirm this, we generate residual plots and run some formal tests for autocorrelation below. We do this for the model with $trend^2$ as a regressor but the results hold for either model.

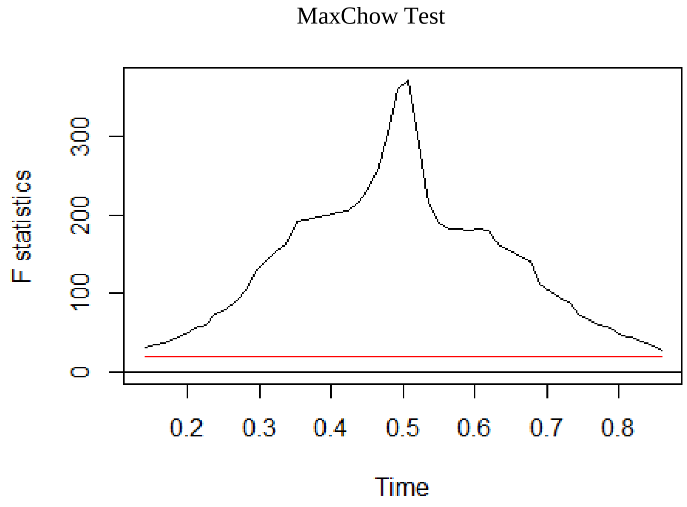


statistic	p.value	method	alternative
0.23861	0	Durbin-Watson test	true autocorrelation is greater than 0

So the residual plot has clear persistent excursions below and above the mean of 0, indicating serial autocorrelation. The Durban Watson p-value of effectively 0 confirms that there is at least first order serial autocorrelation present in the data.

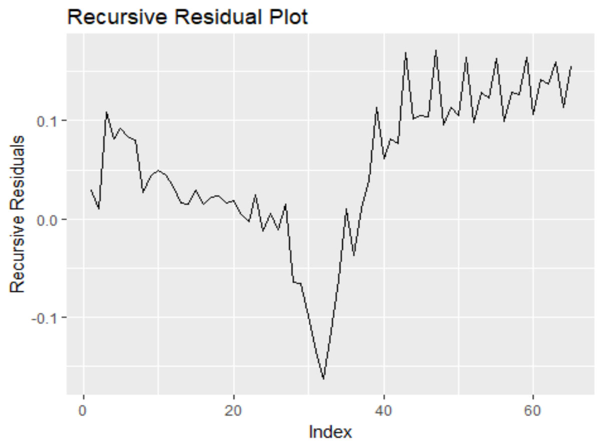
Structural Change

Before dealing with the dynamics of autocorrelation in our dataset, we test to see if there are any structural changes in our data. It does seem that our initial models might have a break somewhere around t=35 and t=40 as noted in the first section. This roughly corresponds to the time around the recession of 2008. We start by running a max Chow test on all time periods between 11 and 60. This leaves about 15 percent of the data at the beginning and end of the time series to use for comparison. The results of the Max Chow test are below



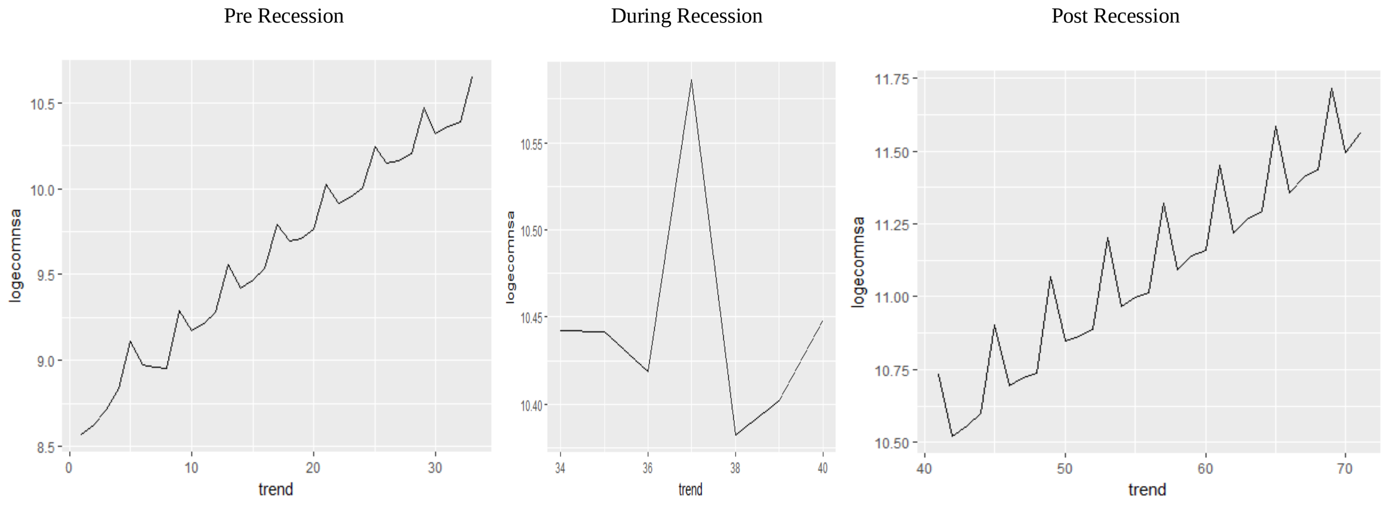
Max Position	F-Stat	p-value
36	372.79	2.20E-16

The plot of the f-statistics show us that right around the recession (ie t=35 to t=40), the F-statistic starts to balloon upwards. The Max value of the Chow test happens at t=36. The associated F-stat is 372.79 and the associated p-value is effectively 0. Thus we reject the null of no structural change. We also plot the recursive residuals obtained by doing 1 step ahead forecasts for every point.



We can clearly see that the recursive residuals plummet to below 0 around the time of the recession. There is a clear pattern to the recursive residuals.

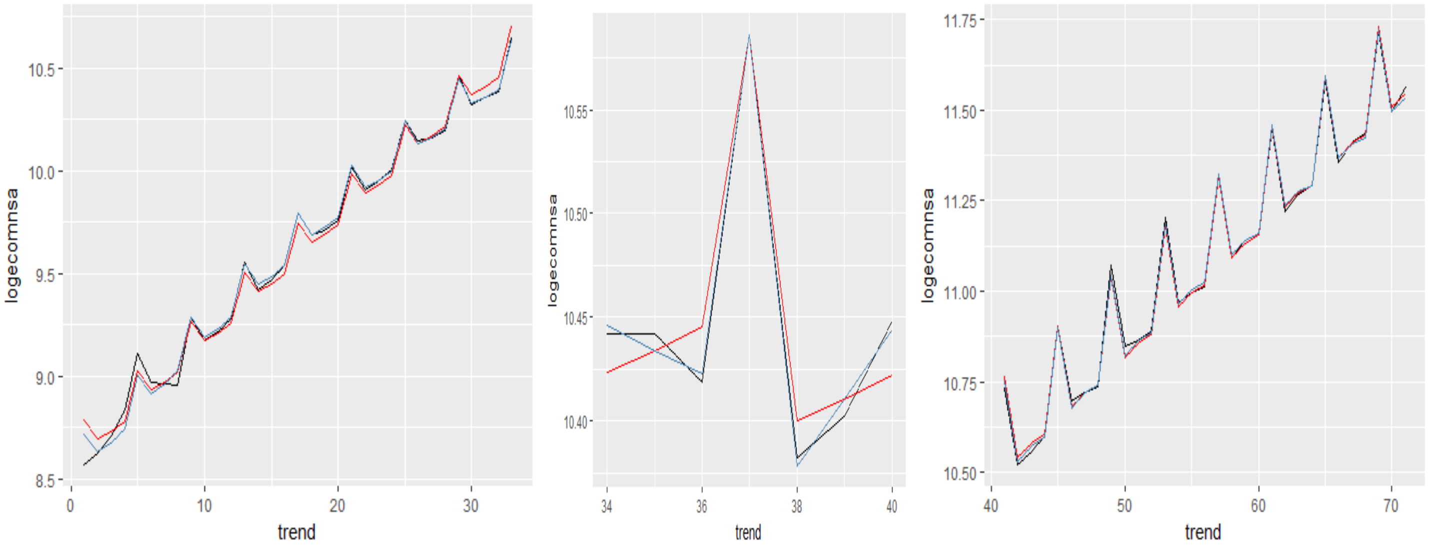
Now for the selection of the breakpoints, we decide to split the data into 3 parts: Pre recession, during recession, and post recession. This is because we believe that online retail sales probably faced a different data generating process during the recession and we don't want the data points from the recession time period to affect the OLS estimates. From FRED's website, we can see that the recession started in Q1 of 2008 and ended in Q3 of 2009. This corresponds to t=34 and t=40. Below we split up the data and plot the 3 separate datasets we now want to model.


Pre Recession

During Recession

Post Recession

These somewhat arbitrary breakpoints are further supported by the fact that now the pre and post recession datasets look roughly linear. To test the robustness of our models, we recreate the models with trend, seasonality, and trend^2 for all 3 data groups and overlay the fitted values on the original data. The red lines are the model with just $trend$, and the blue lines are the models with $trend + trend^2$.



Pre Recession Models

Model	**r.squared**	**adj.r.squared**	**sigma**	**statistic**	**p.value**	**df**	**logLik**	**AIC**	**BIC**	**deviance**
Season + Trend	0.99111	0.98984	0.05907	781.0023	0	5	49.239	-86.4798	-77.500	0.097724
Season + Trend + Trend^2	0.99507	0.99416	0.04472	1091.024	0	6	58.971	-103.943	-93.468	0.0541808

Recession Models

Model	**r.squared**	**adj.r.squared**	**sigma**	**statistic**	**p.value**	**df**	**logLik**	**AIC**	**BIC**	**deviance**
Season + Trend	0.9911168	0.9898477	0.05907	781.0023	0	5	49.2399	-86.479	-77.500	0.09772
Season + Trend + Trend^2	0.9950749	0.9941628	0.04479	1091.024	0	6	58.9719	-103.943	-93.468	0.0541

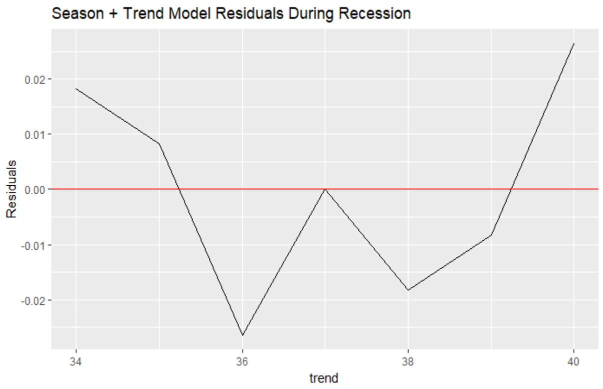
Post-Recession Models

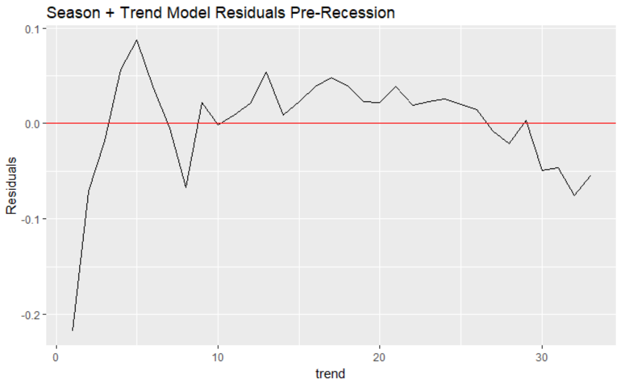
Model	**r.squared**	**adj.r.squared**	**sigma**	**statistic**	**p.value**	**df**	**logLik**	**AIC**	**BIC**	**deviance**
Season + Trend	0.991116	0.9898477	0.05907	781.002	0	5	49.2399	-86.479	-77.500	0.0977240
Season + Trend + Trend^2	0.995074	0.9941628	0.04479	1091.024	0	6	58.9719	-103.943	-93.468	0.0541808

We can see that the models with the $trend^2$. term, ie the blue line, seem to follow the data better. However, the tables, which are the model metrics of all the models, show us that the models with just a trend component have slightly lower BIC scores across the board. So this suggests that a trend plus seasonality model is a good starting point for all three of our datasets. The next question we have to ask is whether autocorrelation is present in the data.

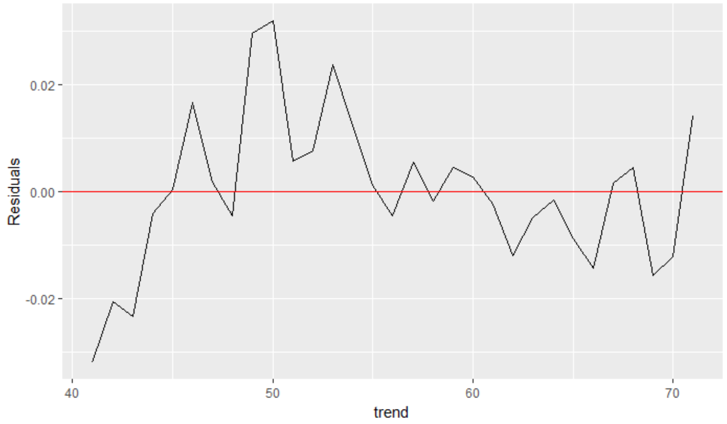
Autucorrelation

When we were looking at the full dataset, we had reason to believe that there was serial autocorrelation. The residual plot had persistent excursions from the mean and the Durbin Watson test rejected the null of no AR(1) autocorrelation. We now see if those results hold in each of our three separate datasets using our baseline model that includes $trend + Q2 + Q3 + Q4 + constant$. We plot the residual plots for all 3 datasets and the associated Durbin Watson tets





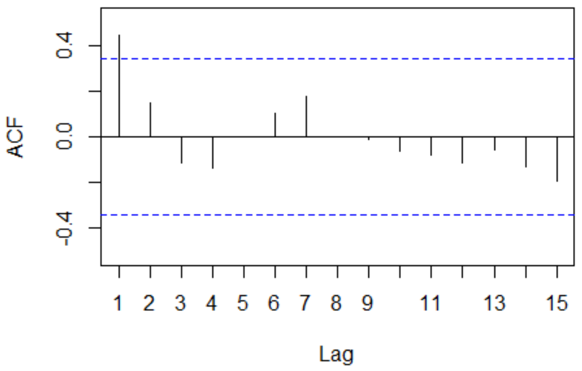
Season + Trend Model Residuals Post-Recession



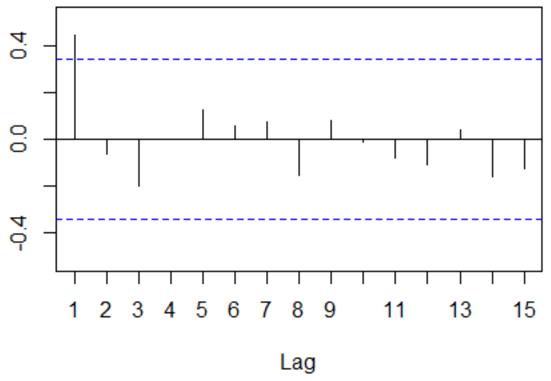
statistic	p.value	method	alternative	dataset
0.5895758	1.7e-06	Durbin-Watson test	true autocorrelation is greater than 0	Pre-Recession
0.8580143	0.0003569	Durbin-Watson test	true autocorrelation is greater than 0	Post-Recession
1.654981	0.1963555	Durbin-Watson test	true autocorrelation is greater than 0	During Recession

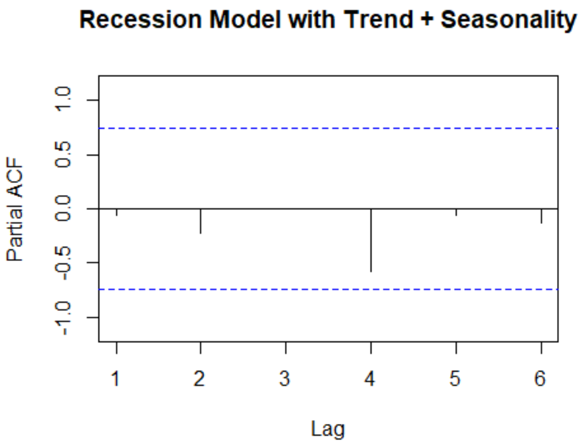
Looking at the data pre and post recession, we see that there are still some, albeit smaller, persistent excursions above or below the mean. This is confirmed by the results of the Durbin Watson test for both of these datasets, which returns an associated p-value of 0.0000017 and 0.0003 respectively. Thus, there is at least first order serial autocorrelation present. For the recession dataset, the DW returns a p-value of 0.19 and we accept the null of no first order serial autocorrelation. However, given the small size of this dataset, we should take these results with a grain of salt. To further analyze the autocorrelation in the pre and post recession datasets, we look at the autocorrelation and partial autocorrelation functions for all 3 datasets.

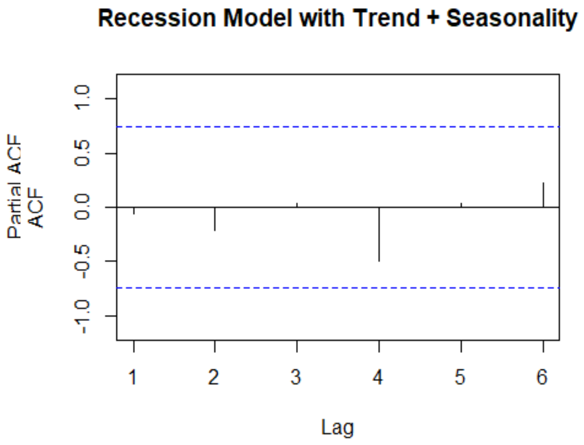
Pre-Recession Model with Trend + Seasonalit

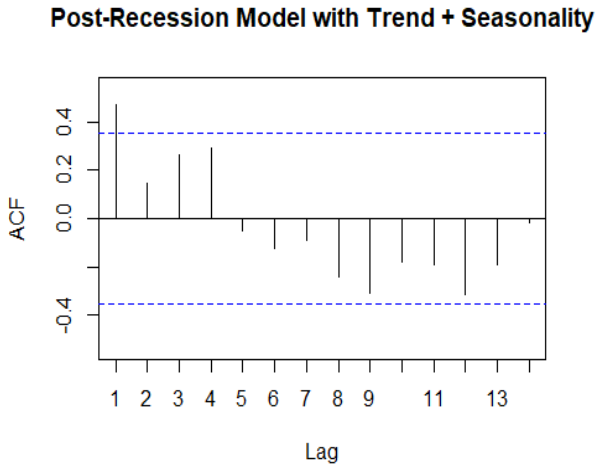


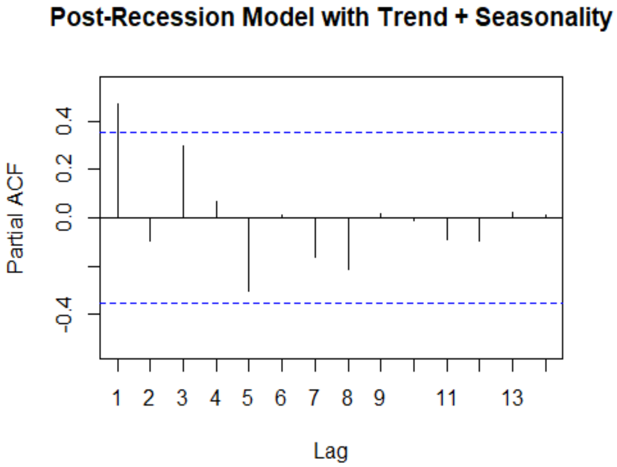
Pre-Recession Model with Trend + Seasonality











So we see that for the pre and post recession datasets, the ACF plots all have significant and positive first order autocorrelation. And it seems like they mostly exponentially decay smoothly to 0 and then exhibit non-significant negative autocorrelation at higher lags. And when looking at the partial autocorrelation functions, the same story unfold as both pre and post recession data have significant first order autocorrelation that then cuts off quickly to 0. In the case of post recession data, there is some high (but insignificant) autocorrelation at lags of 3 and 5 respectively that don't perfectly align with the sharp cut-off story. However in general, these plots give support to some degree of first order autocorrelation for the pre and post recession data. The Durban Watson test statistics from above also confirm this. So it is clear that an AR(1) model would be most appropriate for the pre and post recession data. For the recession data, there doesn't seem to be any significant autocorrelation or partial autocorrelation.

Final AR(1) Models

It's clear that the regression data doesn't exhibit any autocorrelation, but it is a significant problem for the pre and post recession data. So for the pre and post recession data, we run two AR(1) models. One of the models has the following regressors: $trend + Q2 + Q3 + Q4 +$

$intercept$. The other model has the same regressors and an additional $trend^2$ regressor. Below are the 2 model outputs for the pre-recession data

Pre-Recession Data

Trend Model

term	estimate	std.error
ar1	0.886325	0.099879
intercept	8.469131	0.09747
trend	0.063344	0.003873
Q2	-0.02948	0.012686
Q3	-0.04124	0.014632
Q4	0.163163	0.012729

Trend^2 Model

term	estimate	std.error
ar1	0.710878	0.251706
intercept	8.405565	0.090219
trend	0.084674	0.014187
trend2	-0.00068	0.000217
Q2	-0.03003	0.012175
Q3	-0.04089	0.013976
Q4	0.165485	0.012213

Model Metrics

Pre_model	BIC	sigma	loglik	RMSE
AR(1) + Trend + Seasonal	-94.7225	0.001529	59.38019	0.03909862
AR(1) + Trend + Trend^2+ Seasonal	-100.376	0.001173	64.17426	0.03424283

We see that both models have very similar beta coefficients on the intercept and quarters. However the trend^2 model has a slightly higher beta coefficient on $trend$ and a very small negative coefficient on $trend^2$. The coefficient on the first lag also drops slightly from 0.88 to 0.71. The BIC is also marginally lower in the first model with only trend. However the Root Mean Squared Error is slightly lower in the second model with $trend^2$. Next we look at the two model outputs of the post-recession data.

Post-Recession Data

Trend Model

term	**estimate**	**std.error**
ar1	0.5984078	0.1649973
intercept	9.0670612	0.0324010
trend	0.0349400	0.0005867
Q2	0.0050470	0.0044922
Q3	-0.0079584	0.0052635
Q4	0.2591849	0.0044935

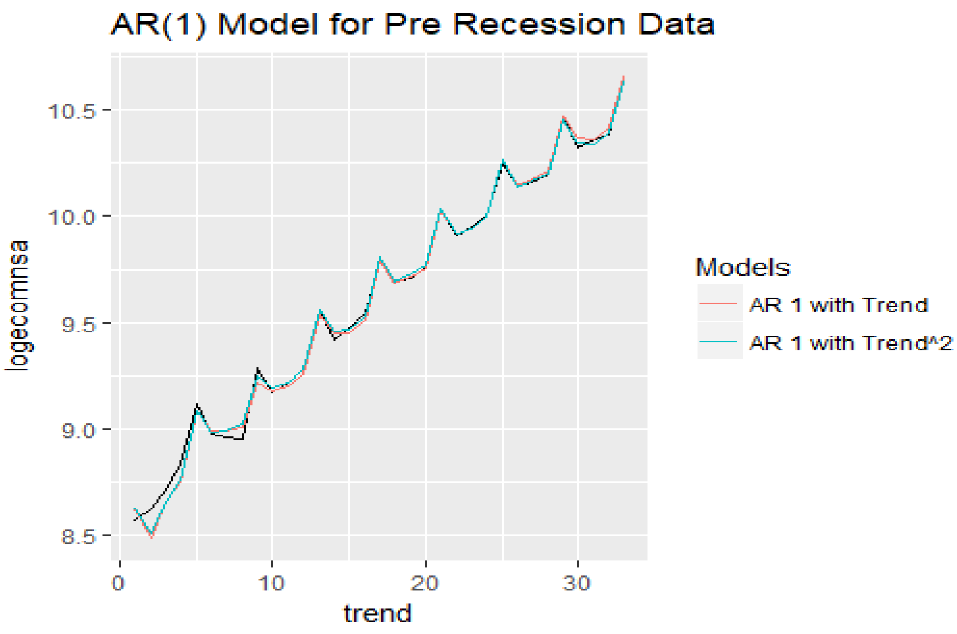
Trend^2 Model

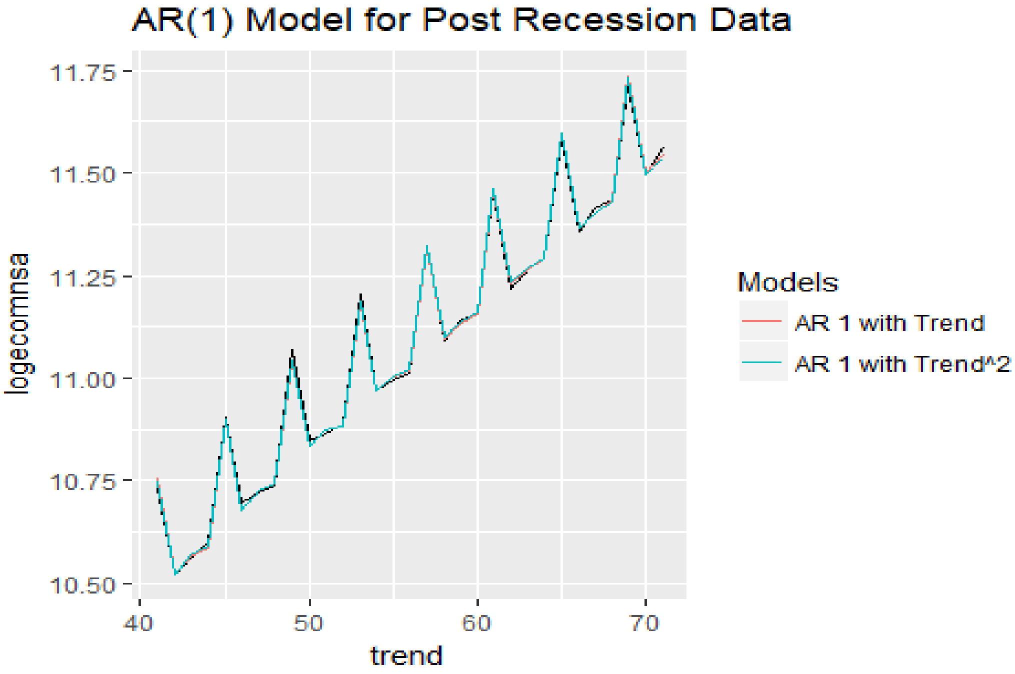
term	**estimate**	**std.error**
ar1	0.44062	0.10323
intercept	8.8042	0.12542
trend	0.04478	0.00461
trend2	-9E-05	4.5E-05
Q2	0.00559	0.00457
Q3	-0.0082	0.00539
Q4	0.25902	0.00464

Model Metrics

Post_Recession_model	BIC	sigma	loglik	RMSE
AR(1) + Trend + Seasonal	-162.606	0.0001401	93.32193	0.01183761
AR(1) + Trend + Trend^2+ Seasonal	-161.969	0.0001290	94.72043	0.01135704

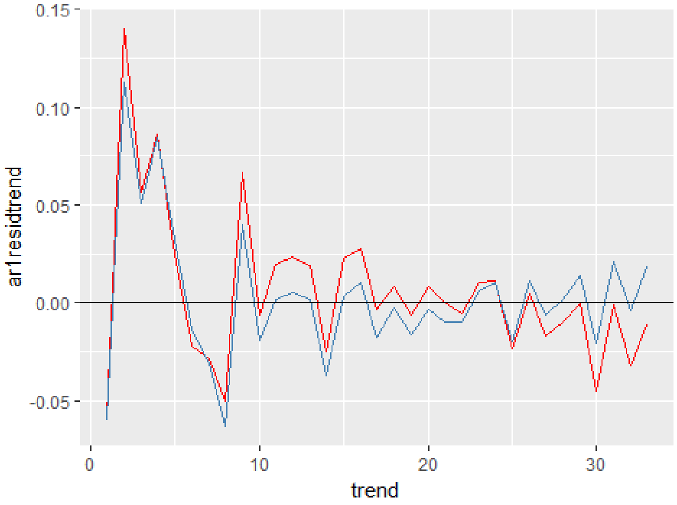
Here we see that the quarterly effects remain the same across both models. Again, the model with $trend^2$ has a small negative coefficient on $trend^2$ and a slightly higher $trend$ coefficient as compared to the first model with only $trend$. The coefficient on the lagged term also decreases. Here we see that the BIC is marginally lower for the second model with the $trend^2$ term. However, the difference in AIC is less than 3, so these results are not really significant. And finally, the RMSE scores are slightly lower in the trend^2 model. To further evaluate these models, we plot both models fitted values against the ‘real’ values of the data. Those graphs are below


AR(1) Model for Pre Recession Data
logecomnsa
10.5
10.0
9.5
9.0
8.5
0
10
20
30
trend
Models
AR 1 with Trend
AR 1 with Trend^2

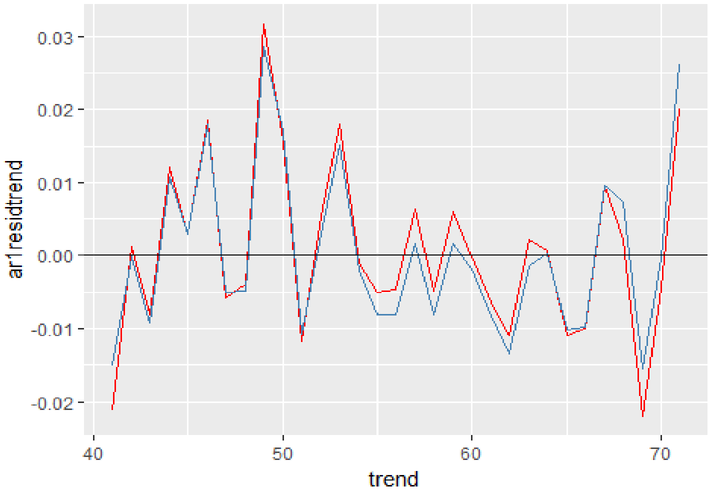

AR(1) Model for Post Recession Data
logecomnsa
11.75
11.50
11.25
11.00
10.75
10.50
40
50
60
70
trend
Models
AR 1 with Trend
AR 1 with Trend^2

In short, both models capture the real data pretty well. To further evaluate these two models, we take a look at the residual plot and partial autocorrelation function plots of both of these models to rule out any further autocorrelation. The blue line connotes the residuals of the model with a $trend^2$ term and the red line is the residuals of the model that doesn't have that term.

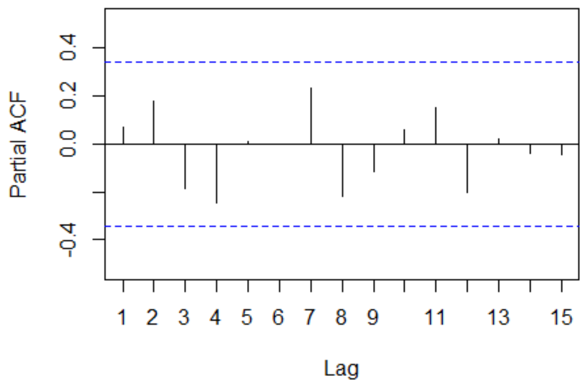
Pre-recession Data Model Residuals



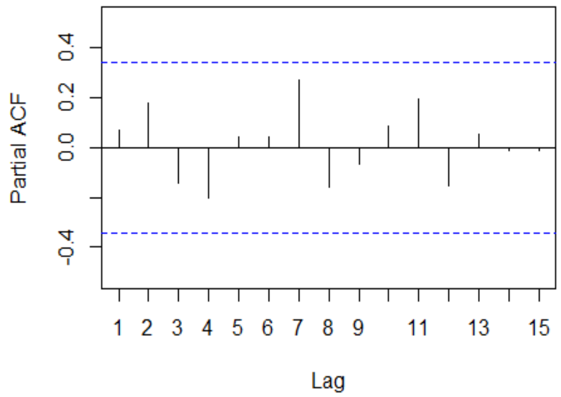
Post-recession Data Model Residuals



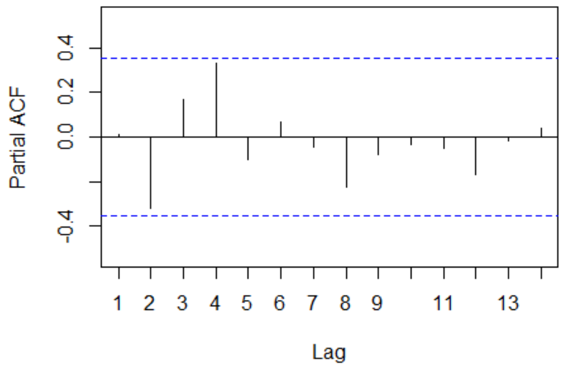
Pre Recession Trend^2 Model



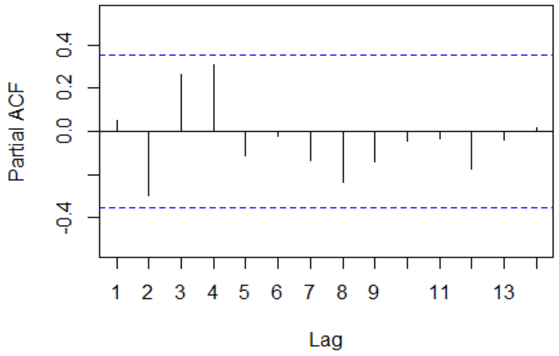
Pre Recession Trend Model



Post Recession Trend Model



Post Recession Trend^2 Model



The residual plots seem to be have much fewer persistent excursions from the mean. The Partial Autocorrelation functions all show non significant partial autocorrelation for all lags, suggesting that there is no longer significant autocorrelation present in the data for either model. If anything, the Trend^2 model seem to have slightly better behaved partial autocorrelation functions. This combined with the fact that the RMSE scores are lower for models with the trend^2 regressor lead us to pick the model with the trend^2 regressor as the final model for both pre and post recession data. For the recession data, we choose the model with just trend and seasonality (and no autocorrelation) as the final model as that seems to fit the data pretty well. Below are the final models for pre-recession data, recession data, and post recession data

Pre-Recession Data

Trend^2 Model

term	estimate	std.error
ar1	0.710878	0.251706
intercept	8.405565	0.090219
trend	0.084674	0.014187
trend2	-0.00068	0.000217
Q2	-0.03003	0.012175
Q3	-0.04089	0.013976
Q4	0.165485	0.012213

Recession Data

Trend Model

term	estimate	std.error	statistic	p.value
(Intercept)	10.42958	0.031021	336.2113	8.8E-06
trend	-0.00582	0.006769	-0.85912	0.480804
Q2	0.01553	0.033847	0.45882	0.6914
Q3	0.033153	0.03582	0.925535	0.452395
Q4	0.179763	0.041176	4.365714	0.048669

Post Recession Data

Trend^2 Model

term	estimate	std.error
ar1	0.44062	0.10323
intercept	8.8042	0.12542

trend	0.04478	0.00461
trend2	-9E-05	4.5E-05
Q2	0.00559	0.00457
Q3	-0.0082	0.00539
Q4	0.25902	0.00464

We quickly interpret these results. Starting with the pre-recession data, we see that that there is a strong autoregressive component as the phi is equal to 0.71. The intercept, or equivalently the constant term for Q1 is 8.405565. There is a slight negative quadratic trend to the data. And finally, there is a strong Q4 effect that raises the intercept by 0.16 and some weak negative effects for Q2 and Q3. These results are basically identical for the post recession data. The only difference is that the phi coefficient has dropped to 0.44. Furthermore, Q2 now has a very weak positive effect and Q4 has an even more positive additive effect on the constant. As for the recession data, we see that all quarters seemed to have a positive effect (in relation to Q1) suggesting that the effects of the recession were most prominent in Q1. And finally the trend is a weakly negative linear one.

Forecasts

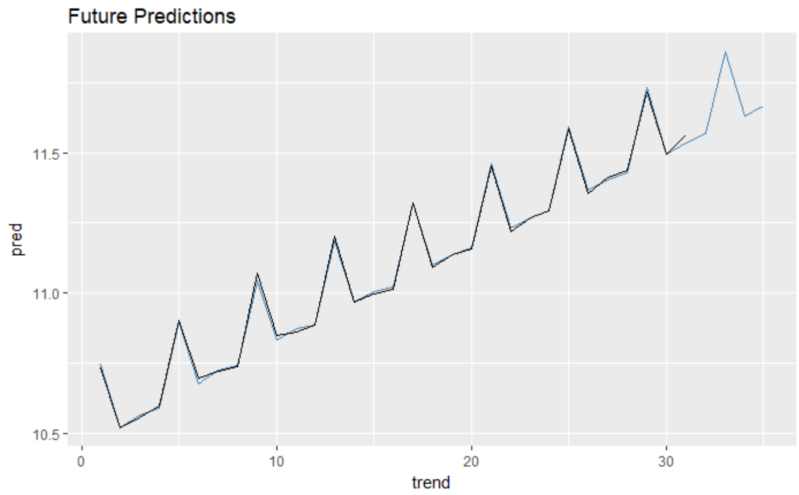
So now, we move onto forecasting the next 4 quarters worth of ecomnsa data, or in our case log ecomnsa data. Since we have 3 separate models, we only use our most recent post recession model to make predictions. Using that data, below are our predictions for the next four quarters:

- `11.56751, 11.86073, 11.63078, 11.66669`

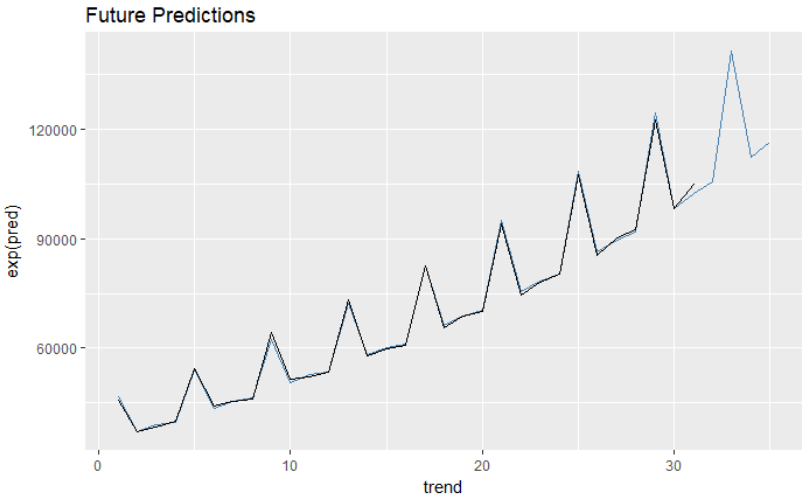
Or if you de-log transformed it, the predictions in terms of ECOMNSA in millions of dollars is

- `105609.9 141595.7 112507.7 116621.7`

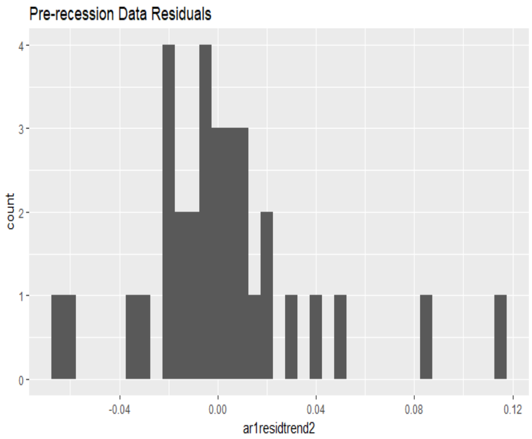
And here they are plotted out. The blue line is our model and the black line is the actual values of logecomnsa.

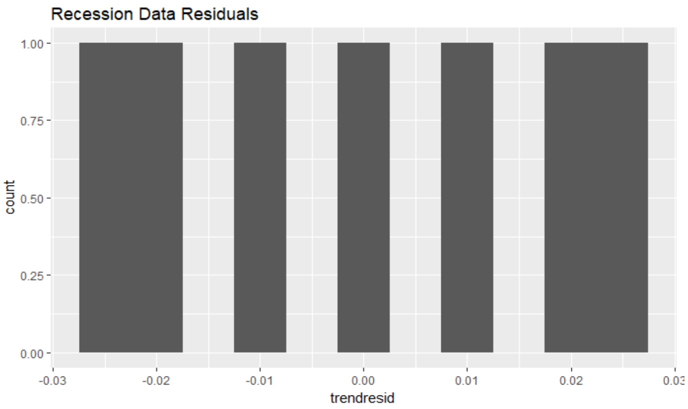


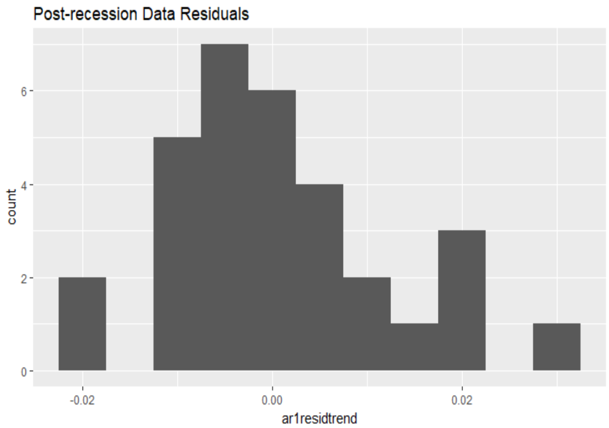
And in the untransformed ECOMNSA scale, the predictions look like this:



Before we make interval and density forecasts for the next period, we look at whether the residuals are normally distributed. And unfortunately, they are not. Here are the residual histograms for each of our models

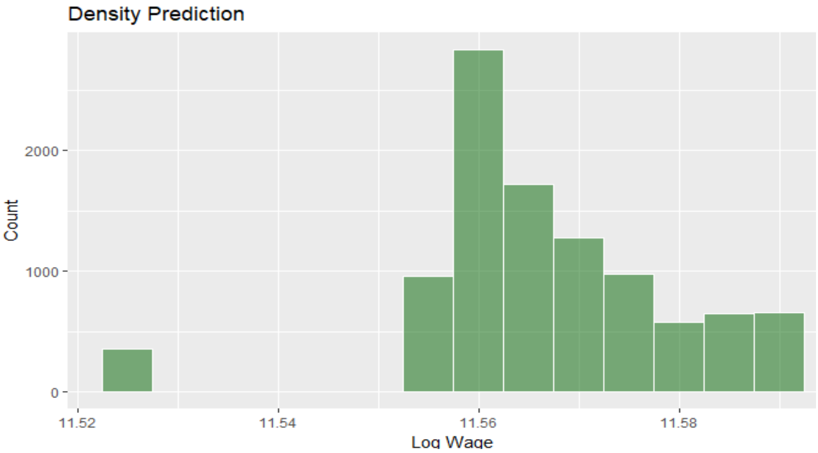






These non-normal looking histograms of residuals could be due to the smaller size of the each of the datasets. Promisingly, the last histogram of the post recession model (which we used for forecasts) looks the most normal. There just seems to be some excess values in the right tail.

Given the non-normal nature of the residual, the interval and density forecasts have to be drawn from simulations. In particular, we run a BGP regression of the final model residuals on all the regressors. This BGP regression will itself be a function of the lagged residuals. So the setup of the regression will be residual today = residual yesterday + regressors. We will take the square root of the fitted values from this regression as the approximate sd of each residual. We will then divide the model residuals by the sd of the residuals to get the standardized residuals. We then assign probability 1/N to each of these standardized residuals and sample with replacement 10000 times. To each sample draw, we multiply the estimated sd of the upcoming quarter (as forecasted by the parameters from the BGP regression) and then add on our point estimate (already estimated above as 11.56751). This will help us come up with a simulation distribution that we can use as our density forecast and make interval estimates. Below are the density estimates using the method described above.



Putting it all tsogether, below is the point, and interval forecasts for our one period ahead forecast:

Point_estimate	2.5% Quantile	97.5 Quantile
11.56751	11.52749	11.58964

Code Appendix

```
knitr::opts_chunk$set(echo = FALSE)
library(tidyverse)
library(forecast)
library(lubridate)
library(broom)
library(strucchange)
library(lmtest)
library(knitr)
data = read.csv("ECOMNSA.csv")
data = data %>%
  mutate(date = as.Date(DATE),
         logecomnsa = log(ECOMNSA),
         ecomnsa = ECOMNSA,
         year = year(DATE),
         month = month(DATE),
         day = day(DATE),
         Q2 = ifelse(month == 4, 1, 0),
         Q3 = ifelse(month == 7, 1, 0),
         Q4 = ifelse(month == 10,1, 0),
         trend = 1:nrow(data),
         trend2 = trend^2) %>%
  select(month, year, logecomnsa, ecomnsa, trend,trend2, Q2, Q3, Q4)


ggplot(data, aes(x = trend)) +
  geom_line(aes(y = ecomnsa))
ggplot(data, aes(x = trend)) +
  geom_line(aes(y = logecomnsa))

#plot(data[, "logecomnsa"], ylab = "log ecomnsa)")
season_trendfit = lm(logecomnsa ~ trend+Q2+Q3+Q4, data = data)
season_trend2fit = lm(logecomnsa ~ trend+trend2+Q2+Q3+Q4, data = data)


summ_basemodels = rbind(glance(season_trendfit),(glance(season_trend2fit)))
summ_basemodels = cbind(Model = c("Season + Trend", "Season + Trend + Trend^2"),summ_basemodel
s)


data$seasontrendfitted = season_trendfit$fitted.values
data$seasontrendresid = season_trendfit$residuals
data$seasontrend2fitted = season_trend2fit$fitted.values
data$seasontrend2resid = season_trend2fit$residuals


ggplot(data, aes(x = trend)) +
  geom_line(aes(y = logecomnsa)) +
  geom_line(aes(y = seasontrendfitted), col = "blue") +
  geom_line(aes(y = seasontrend2fitted), col = "red") +
  ggtitle("Season + Trend + Trend^2 Model")

ggplot(data, aes(x = trend)) +
  geom_line(aes(y = seasontrend2resid)) +
  geom_hline(yintercept = 0, col = "red")+
  ggtitle("Season + Trend + Trend^2 Model Residuals")+
  ylab("Residuals")


kable(tidy(dwtest(season_trend2fit)))
```

```r
f_Stats = Fstats(logecomnsa ~ trend+trend2+Q2+Q3+Q4, data = data)
plot(f_Stats)

print(paste0("point of biggest F-stat is t=",f_Stats$breakpoint))
maxchowtest = sctest(f_Stats, type = c("supF"))
maxchowtest




rec_resid = data.frame(rec_residu = recresid(season_trend2fit))

ggplot(data = rec_resid) +
  geom_line(aes(x = 1:nrow(rec_resid), y =rec_residu))+
  xlab("Index") +
  ylab("Recursive Residuals") +
  ggtitle("Recursive Residual Plot")


data_pre = data %>%
  filter(trend %in% 1:33)
data_recession = data %>%
  filter(trend %in% 34:40)
data_post = data %>%
  filter(trend %in% 41:nrow(data))

data_post = data_post %>%
  mutate(trend = 1:nrow(data_post),
         trend2 = trend^2)
data_recession = data_recession %>%
  mutate(trend = 1:nrow(data_recession),
         trend2 = trend^2)


ggplot(aes(x=trend), data = data_pre)+
  geom_line(aes(y = logecomnsa))

ggplot(aes(x=trend), data = data_recession)+
  geom_line(aes(y = logecomnsa))

ggplot(aes(x=trend), data = data_post)+
  geom_line(aes(y = logecomnsa))


f_stats_pre = Fstats(logecomnsa ~ trend+trend2+Q2+Q3+Q4, data = data_pre)
plot(f_stats_pre)

f_stats_post = Fstats(logecomnsa ~ trend+Q2+Q3+Q4, data = data_post)
plot(f_stats_post)

print(paste0("point of biggest F-stat is t=",f_stats_post$breakpoint))
maxchowtest = sctest(f_stats_post, type = c("supF"))
maxchowtest



pre_trend = lm(logecomnsa ~ trend+Q2+Q3+Q4, data = data_pre)
pre_trend2= lm(logecomnsa ~ trend+trend2+Q2+Q3+Q4, data = data_pre)
data_pre$trendfitted = pre_trend$fitted.values
```

```
data_pre$trendresid  = pre_trend$residuals
data_pre$trend2fitted= pre_trend2$fitted.values
data_pre$trend2resid = pre_trend2$residuals

post_trend = lm(logecomnsa ~ trend+Q2+Q3+Q4, data = data_post)
post_trend2= lm(logecomnsa ~ trend+trend2+Q2+Q3+Q4, data = data_post)
data_post$trendfitted = post_trend$fitted.values
data_post$trendresid  = post_trend$residuals
data_post$trend2fitted= post_trend2$fitted.values
data_post$trend2resid = post_trend2$residuals


recession_trend = lm(logecomnsa ~ trend+Q2+Q3+Q4, data = data_recession)
recession_trend2= lm(logecomnsa ~ trend+trend2+Q2+Q3+Q4, data = data_recession)
data_recession$trendfitted = recession_trend$fitted.values
data_recession$trendresid  = recession_trend$residuals
data_recession$trend2fitted= recession_trend2$fitted.values
data_recession$trend2resid = recession_trend2$residuals

ggplot(aes(x=trend), data = data_pre)+
  geom_line(aes(y = logecomnsa)) +
  geom_line(aes(y = trendfitted), col = "red") +
  geom_line(aes(y = trend2fitted), col = "steelblue")


ggplot(aes(x=trend), data = data_recession)+
  geom_line(aes(y = logecomnsa))+
  geom_line(aes(y = trendfitted), col = "red") +
  geom_line(aes(y = trend2fitted), col = "steelblue")


ggplot(aes(x=trend), data = data_post)+
  geom_line(aes(y = logecomnsa))+
  geom_line(aes(y = trendfitted), col = "red") +
  geom_line(aes(y = trend2fitted), col = "steelblue")


summ_models_pre = rbind(glance(pre_trend),(glance(pre_trend2)))
summ_models_pre = cbind(Model = c("Season + Trend", "Season + Trend + Trend^2"),summ_models_pr
e)

summ_models_rec = rbind(glance(recession_trend),(glance(recession_trend2)))
summ_models_rec = cbind(Model = c("Season + Trend", "Season + Trend + Trend^2"),summ_models_re
c)

summ_models_post = rbind(glance(post_trend),(glance(post_trend2)))
summ_models_post = cbind(Model = c("Season + Trend", "Season + Trend + Trend^2"),summ_models_p
ost)

kable(summ_models_pre)
kable(summ_models_rec)
kable(summ_models_post)


ggplot(data = data_pre, aes(x = trend)) +
  geom_line(aes(y = trendresid)) +
  geom_hline(yintercept = 0, col = "red")+
  ggtitle("Season + Trend Model Residuals Pre-Recession")+
  ylab("Residuals")

ggplot(data = data_recession, aes(x = trend)) +
  geom_line(aes(y = trendresid)) +
  geom_hline(yintercept = 0, col = "red")+
  ggtitle("Season + Trend Model Residuals During Recession")+
  ylab("Residuals")


ggplot(data = data_post, aes(x = trend)) +
  geom_line(aes(y = trendresid)) +
  geom_hline(yintercept = 0, col = "red")+
  ggtitle("Season + Trend Model Residuals Post-Recession")+
  ylab("Residuals")


kable(tidy(dwtest(pre_trend)))
kable(tidy(dwtest(post_trend)))
kable(tidy(dwtest(recession_trend)))


Acf(data_pre$trendresid, main = "Pre-Recession Model with Trend + Seasonality")
Pacf(data_pre$trendresid,main = "Pre-Recession Model with Trend + Seasonality")

Acf(data_recession$trendresid, main = "Recession Model with Trend + Seasonality")
Pacf(data_recession$trendresid,main = "Recession Model with Trend + Seasonality")

Acf(data_post$trendresid, main = "Post-Recession Model with Trend + Seasonality")
Pacf(data_post$trendresid,main = "Post-Recession Model with Trend + Seasonality")



ts_data_pre = data_pre[,"logecomnsa"]
ts_data_post = data_post[,"logecomnsa"]

ar1pre = arima(ts_data_pre, order = c(1,0,0), xreg = data_pre %>% select(trend, Q2, Q3, Q4))
ar1post = arima(ts_data_post, order = c(1,0,0), xreg = data_post %>% select(trend, Q2, Q3, Q4)
)

ar1pre2 = arima(ts_data_pre, order = c(1,0,0), xreg = data_pre %>% select(trend, trend2, Q2, Q
3, Q4))
ar1post2 = arima(ts_data_post, order = c(1,0,0), xreg = data_post %>% select(trend, trend2, Q2
, Q3, Q4))

data_pre$ar1fittedtrend  = fitted(ar1pre)
data_pre$ar1residtrend   = ar1pre$residuals

data_post$ar1fittedtrend = fitted(ar1post)
data_post$ar1residtrend   = ar1post$residuals

data_pre$ar1fittedtrend2 = fitted(ar1pre2)
data_pre$ar1residtrend2   = ar1pre2$residuals

data_post$ar1fittedtrend2= fitted(ar1post2)
data_post$ar1residtrend2   = ar1post2$residuals



kable(tidy(ar1pre))
kable(tidy(ar1pre2))

summary(ar1pre)
summary(ar1pre2)
```

```r
kable(data.frame(Pre_model = c("AR(1) + Trend + Seasonal", "AR(1) + Trend + Trend^2+ Seasonal"
),
            BIC = c(AIC(ar1pre, k = log(length(ts_data_post))),AIC(ar1pre2, k = log(length(ts_
data_pre)))),
           sigma = c(ar1pre$sigma2, ar1pre2$sigma2),
           loglik = c(ar1pre$loglik,ar1pre2$loglik)))

kable(tidy(ar1post))
kable(tidy(ar1post2))

summary(ar1post)
summary(ar1post2)

kable(data.frame(Post_model = c("AR(1) + Trend + Seasonal", "AR(1) + Trend + Trend^2+ Seasonal
"),
           BIC = c(AIC(ar1post, k = log(length(ts_data_post))),AIC(ar1post2, k = log(length(ts
_data_post)))),
           sigma = c(ar1post$sigma2, ar1post2$sigma2),
           loglik = c(ar1post$loglik,ar1post2$loglik)))




ggplot(data = data_pre, aes(x=trend))+
  geom_line(aes(y=logecomnsa))+
  geom_line(aes(y= ar1fittedtrend, col = "red"))+
  geom_line(aes(y= ar1fittedtrend2, col = "steelblue"))+
  ggtitle("AR(1) Model for Pre Recession Data")+
  scale_color_hue(labels = c("AR 1 with Trend","AR 1 with Trend^2" )) +
  labs(col = "Models")

ggplot(data = data_post, aes(x=trend))+
  geom_line(aes(y=logecomnsa))+
  geom_line(aes(y= ar1fittedtrend, col = "red"))+
  geom_line(aes(y= ar1fittedtrend2, col = "steelblue"))+
  ggtitle("AR(1) Model for Post Recession Data")+
  scale_color_hue(labels = c("AR 1 with Trend","AR 1 with Trend^2" )) +
  labs(col = "Models")


ggplot(data_pre, aes(x = trend)) +
  geom_line(aes(y = ar1residtrend), col = "red") +
  geom_line(aes(y = ar1residtrend2), col = "steelblue") +
  geom_hline(yintercept = 0, col = "black")+
  ggtitle("Pre-recession Data Model Residuals")

ggplot(data_post, aes(x = trend)) +
  geom_line(aes(y = ar1residtrend), col = "red") +
  geom_line(aes(y = ar1residtrend2), col = "steelblue") +
  geom_hline(yintercept = 0, col = "black")+
  ggtitle("Post-recession Data Model Residuals")



Pacf(data_pre$ar1residtrend, main = "Pre Recession Trend Model")
Pacf(data_pre$ar1residtrend2, main = "Pre Recession Trend^2 Model")

Pacf(data_post$ar1residtrend, main = "Post Recession Trend Model")
Pacf(data_post$ar1residtrend2, main = "Post Recession Trend^2 Model")
```

```
ggplot(data = data_pre)+
  geom_histogram(aes(ar1residtrend2), binwidth = .005)+
  ggtitle("Pre-recession Data Residuals")
ggplot(data = data_recession)+
  geom_histogram(aes(trendresid), binwidth = .005)+
  ggtitle("Recession Data Residuals")
ggplot(data = data_post)+
  geom_histogram(aes(ar1residtrend), binwidth = .005)+
  ggtitle("Post-recession Data Residuals")



x1 = data.frame(trend = 32:35, trend2 = (32:35)^2, Q2 = c(0,0,0,1), Q3 = c(1,0,0,0), Q4 = c(0,
1,0,0))

predic = predict(ar1post2, n.ahead = 4, newxreg = x1,
        se.fit = TRUE)


final_predictions = data.frame(pred = (append(data_post$ar1fittedtrend2, predic$pred)), actual
= append(data_post$logecomnsa, c(NA, NA, NA, NA)))
final_predictions = data.frame(pred = (append(data_post$ar1fittedtrend2, predic$pred)), actual
= append(data_post$logecomnsa, c(NA, NA, NA, NA)), trend = 1:nrow(final_predictions))

ggplot(data= final_predictions, aes(x=trend))+
  geom_line(aes(y=pred), col = "steelblue")+
  geom_line(aes(y=actual)) +
  ggtitle("Future Predictions")

ggplot(data= final_predictions, aes(x=trend))+
  geom_line(aes(y=exp(pred)), col = "steelblue")+
  geom_line(aes(y=exp(actual))) +
  ggtitle("Future Predictions") +
  ylab("ECOMNSA")

#Getting Interval and Density Forecasts

whitelm<-arima(ar1post2$residuals^2, order = c(1,0,0), xreg = data_post %>% select(trend, tren
d2, Q2, Q3, Q4))
sd_resid = sqrt(fitted(whitelm))
std_resid = ar1post2$residuals / sd_resid


x2 = data.frame(trend = 32, trend2 = (32)^2, Q2 = c(0), Q3 = c(1), Q4 = c(0))

sample_std_resid = sample(std_resid, 10000, replace = TRUE)
new_arrival_sd = sqrt(predict(whitelm, n.ahead=1, newxreg = x2 )$pred)[[1]]
new_arrival_mean = (predict(ar1post2, n.ahead=1, newxreg = x2 )$pred)[[1]]

dist_new_arrival = sample_std_resid*new_arrival_sd + new_arrival_mean

new_arrival_distribution_hist = ggplot()+
  geom_histogram(aes(dist_new_arrival),
                 col = "white",
                 fill = "darkgreen",
                 binwidth = .005,
                 alpha=.5) +
                 labs(title="Density Prediction ") +
                 labs(x="Log Wage", y="Count")
```

```r
new_person_predictions = cbind(Point_estimate = new_arrival_mean,
                               data.frame(as.list(quantile(dist_new_arrival, c(0.025,0.975))))
)


new_arrival_distribution_hist
kable(new_person_predictions)
```