Time Series Analysis: Pharmaceutical, cosmetic and toiletry goods retailing in Australia

Asghar Mustafa

31 May 2019

This work aims to develop an optimal time series model for predicting future monthly turnover of retail of Pharmaceutical, cosmetic and toiletry goods in Australia. This would cover broadly 3 aspects of time series models i.e. Classical Methods, ETS Methods and ARIMA methods.

Data Reading

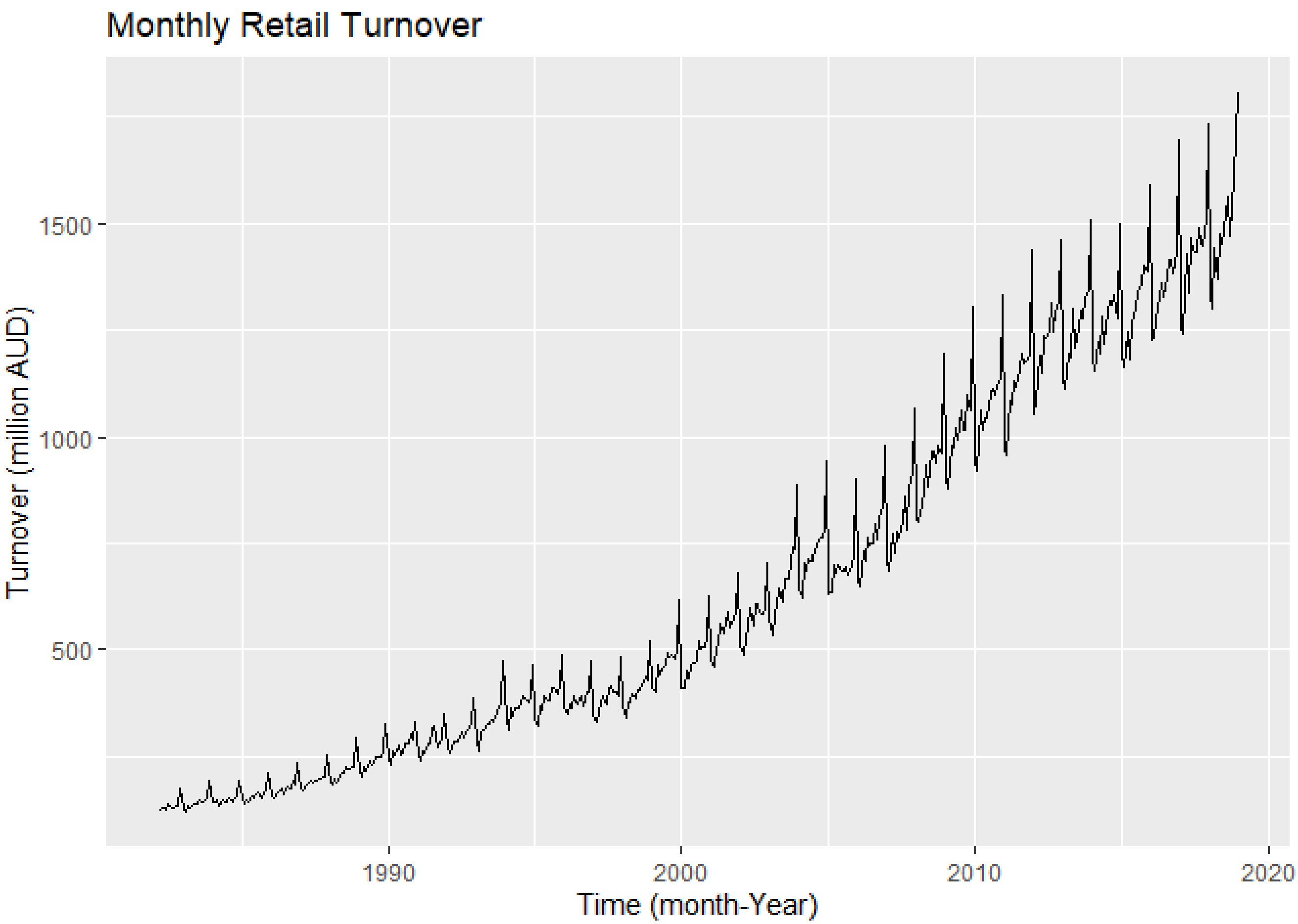
Setting up the libraries...

```
library(fpp2)
library(readxl)
library(dplyr)
library(kableExtra)
library(knitr)
```

```
x <- read_xlsx(here::here("/RetailDataIndividualFull.xlsx"), skip=3)
y<- pull(x,"28905644")
```

Plotting the total Monthly Turnover (in millions AUD) from retail of Pharmaceutical, cosmetic and toiletry goods in Australia between April 1982 and December 2016.

```
myts<- ts(y,start = c(1982,4), frequency = 12)
train <- window(myts,end=time(myts)[length(myts)-24])
autoplot(myts,xlab="Time (month-Year)",ylab="Turnover (million AUD)",main="Monthly Retail Turnover")
```

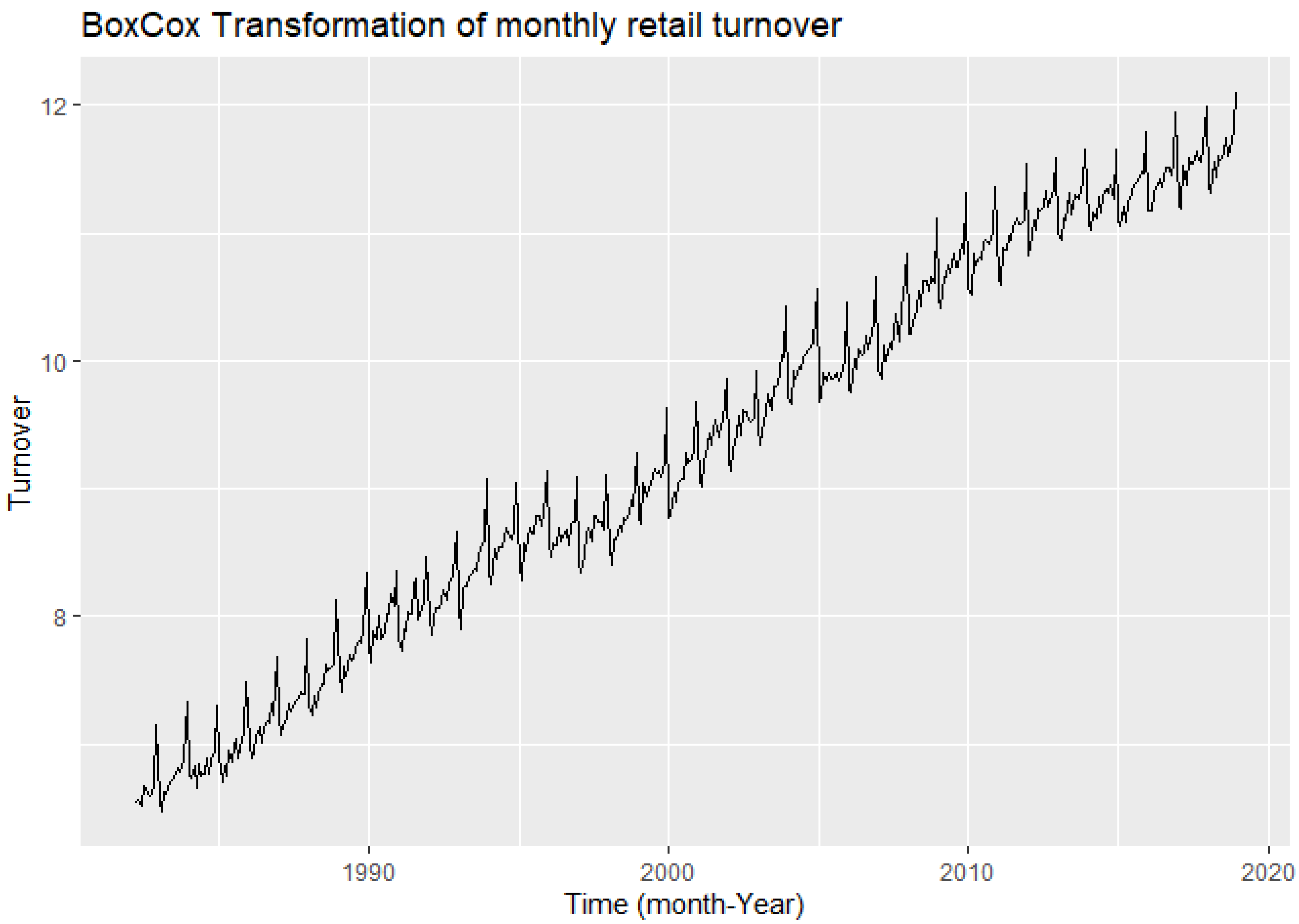


The plot shows an upward trend from 1982 to 2016 with strong seasonal (shown by regular spikes) and cyclic component (shown by the repetitive shape/behaviour) in the data.

Classical Methods

Since it can be observed that the time series is heteroskedastic (no constant error variance), therefore it can be useful to transform the data. To do this box cox function in R was used with different lambda values including lambda=0 (log transformation), lambda=1/3 (cube root transformation), lambda=-1 (Inverse transformation) and lambda=0.10788 (box cox calculated transformation).

```
lam=BoxCox.lambda(myts)
autoplot(BoxCox(myts,lambda=lam),xlab="Time (month-Year)",ylab="Turnover", main="BoxCox Transformation of monthly ret
```

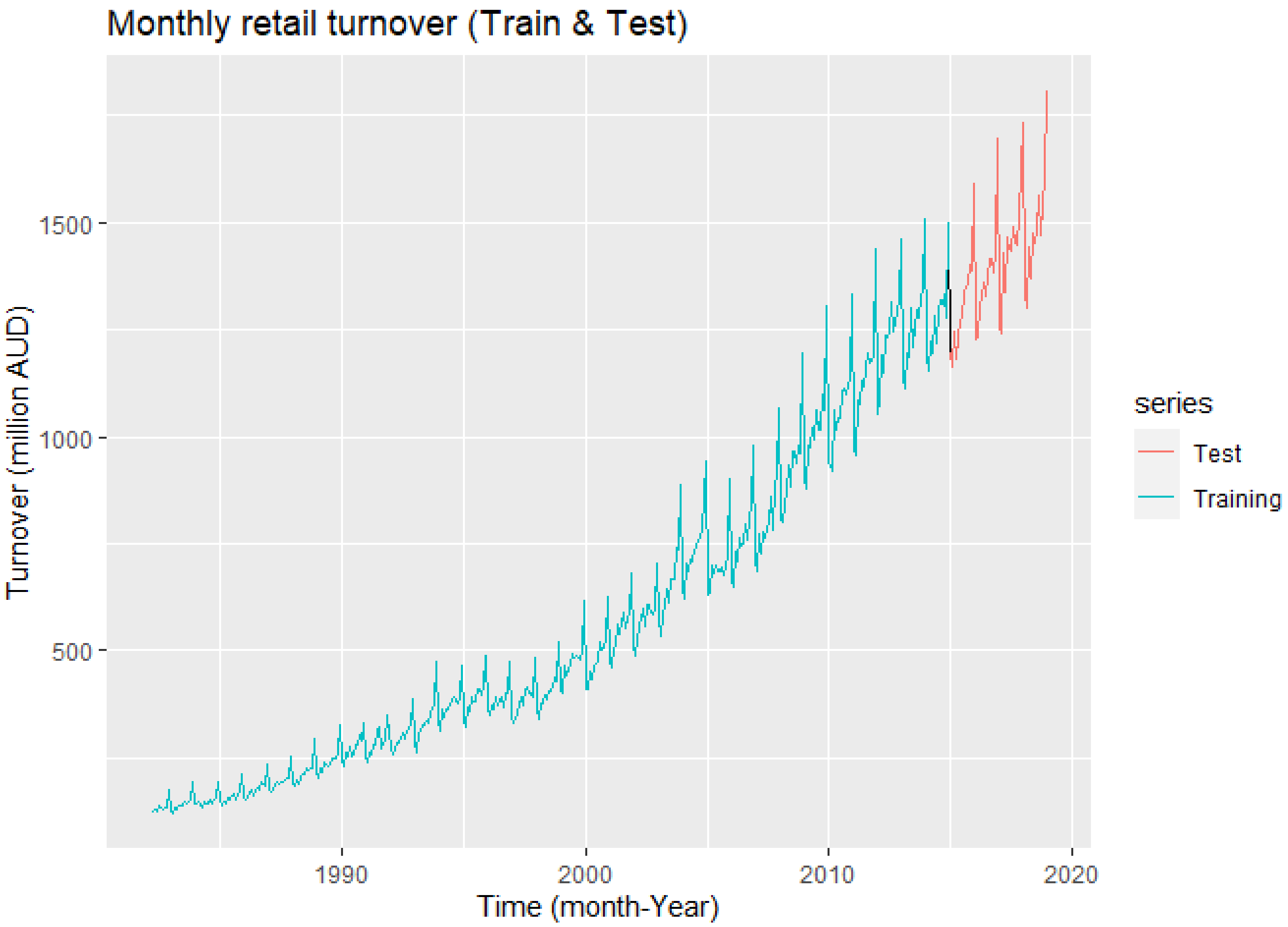


As can be seen from the attached figure, the lambda=0.107 produces the most homoscedastic data and somewhat balances the seasonal and random component across the time series. Other transformations still show signs of hetroskedasticity across the time series.

```
training<-window(myts, start=c(1982,4), end=c(2014,12))
test<-window(myts, start=c(2015,1))

autoplot(myts,xlab="Time (month-Year)",ylab="Turnover (million AUD)", main="Monthly retail turnover (Train & Test)")
```

```
autolayer(training, series="Training") +
autolayer(test, series="Test")
```



Moreover, the plot shows segregation of data in train and test.

```
f1 <- snaive(training, h=length(test))
```

```
accuracy(f1,test)%>%kable()
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	35.92388	53.37026	40.29029	6.682525	7.478686	1.000000	0.7783112	NA
Test set	128.47500	151.99458	129.60417	8.798384	8.890764	3.216759	0.6935665	1.208381

```
f2 <- rwf(training, drift=TRUE, h=length(test))
accuracy(f2,test)%>%kable()
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	0.0000	75.07381	42.19993	-1.030052	7.818995	1.047397	-0.3826597	NA
Test set	-182.8479	214.34195	199.65353	-13.849901	14.822950	4.955376	0.1861301	1.775751

```
f3 <- naive(training, h=length(test))
accuracy(f3,test)%>%kable()
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	3.506378	75.15565	42.65485	-0.0382175	7.893271	1.058688	-0.3826597	NA
Test set	-96.941667	168.08258	140.69583	-7.8930073	10.471420	3.492053	0.4099014	1.388561

```
f4 <- meanf(training, h=length(test))
accuracy(f4,test)%>%kable()
```

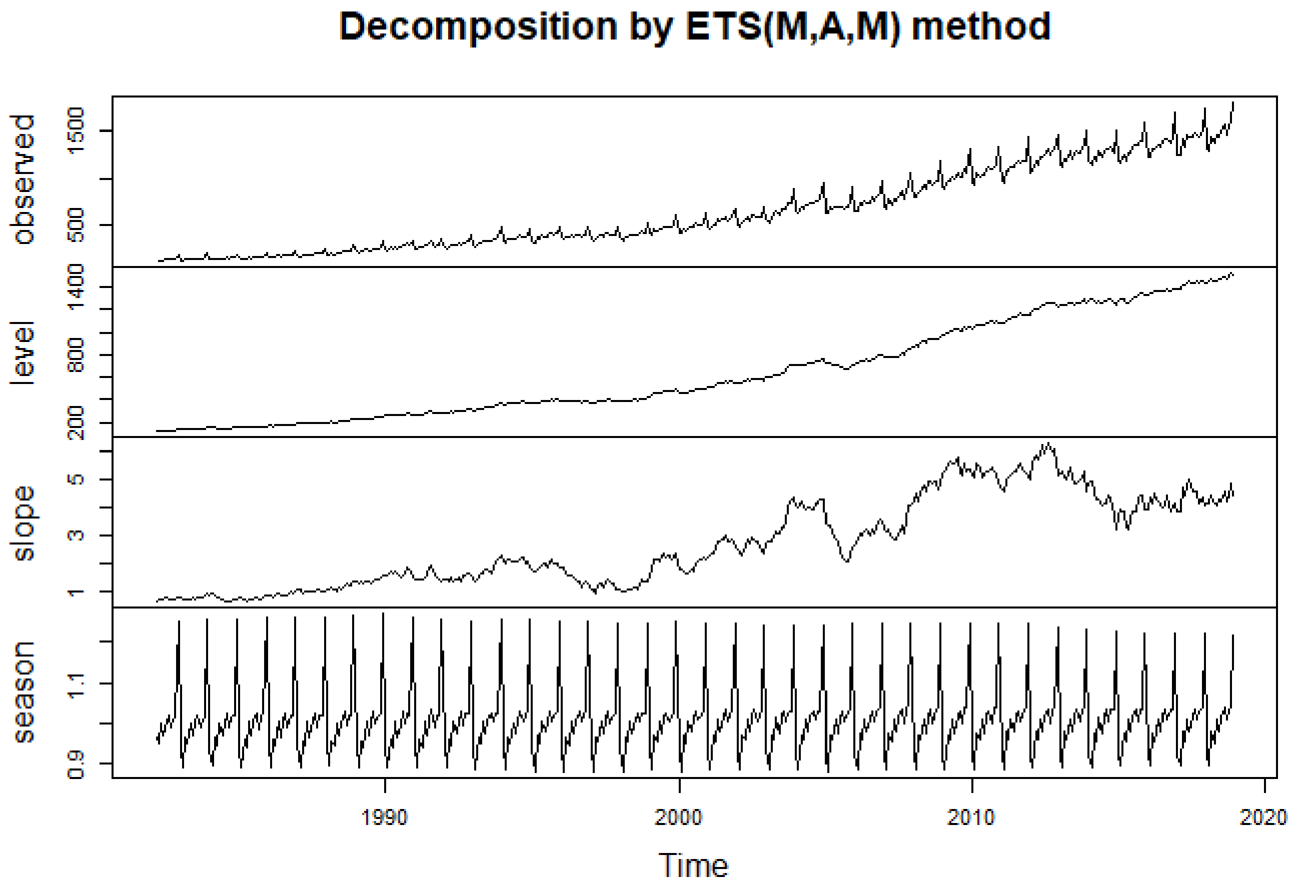
	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	0.0000	362.0075	305.4514	-58.14205	87.11058	7.581266	0.9680577	NA
Test set	846.6571	857.7192	846.6571	59.97427	59.97427	21.013924	0.4099014	6.980858

As can be seen from the above table, RMSE of seasonal naive is better than RMSE of naive. Moreover, snaive also has a better MAE, MAPE and MASE which suggests that Seasonal naive is a better forecasting method for future predictions of Monthly turnover for retail of Pharmaceutical, cosmetic and toiletry goods in Australia.

ETS Models

As can be seen from the retail data has a linearly upward trending curve therefore this is considered an additive trend. The size of the variation (variance) in the turnover for the given time series also significantly increases with time which suggests the multiplicative seasonality of the time series. This multiplicative increase in seasonality along with additive increase in trend suggests that the model with a multiplicative error would give the best model. Therefore, the selected model is MAM i.e. multiplicative errors, Additive trend and Multiplicative Seasonality.

```
fit_Assign3 <- ets(myts,model = 'MAM')
plot(fit_Assign3)
```

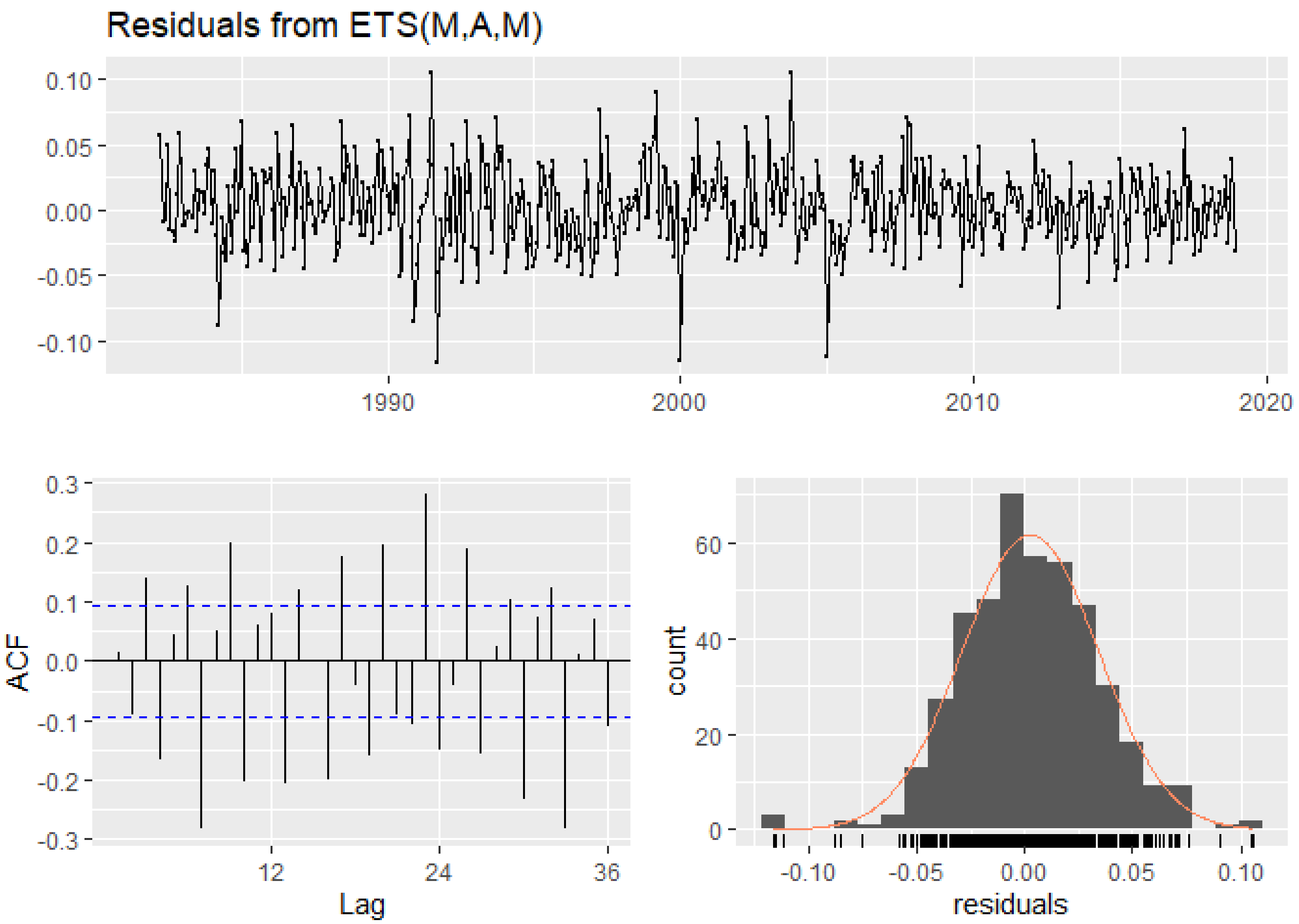


```
summary(fit_Assign3)
```

```
## ETS(M,A,M)
##
## Call:
##  ets(y = myts, model = "MAM")
##
##   Smoothing parameters:
```

```
##     alpha = 0.4705
##     beta  = 0.0095
##     gamma = 0.0831
##
##   Initial states:
##     l = 124.9944
##     b = 0.6418
##     s = 0.9657 0.8928 0.9394 1.2458 1.0121 1.0134
##            0.9918 1.0215 1.0007 0.9697 1.0016 0.9456
##
##   sigma:  0.0322
##
##      AIC     AICc      BIC
## 5158.732 5160.179 5228.246
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.6659529 21.38441 14.65863 0.1568738 2.454953 0.3494417
##                     ACF1
## Training set -0.07827521
```

```
checkresiduals(fit_Assign3)
```



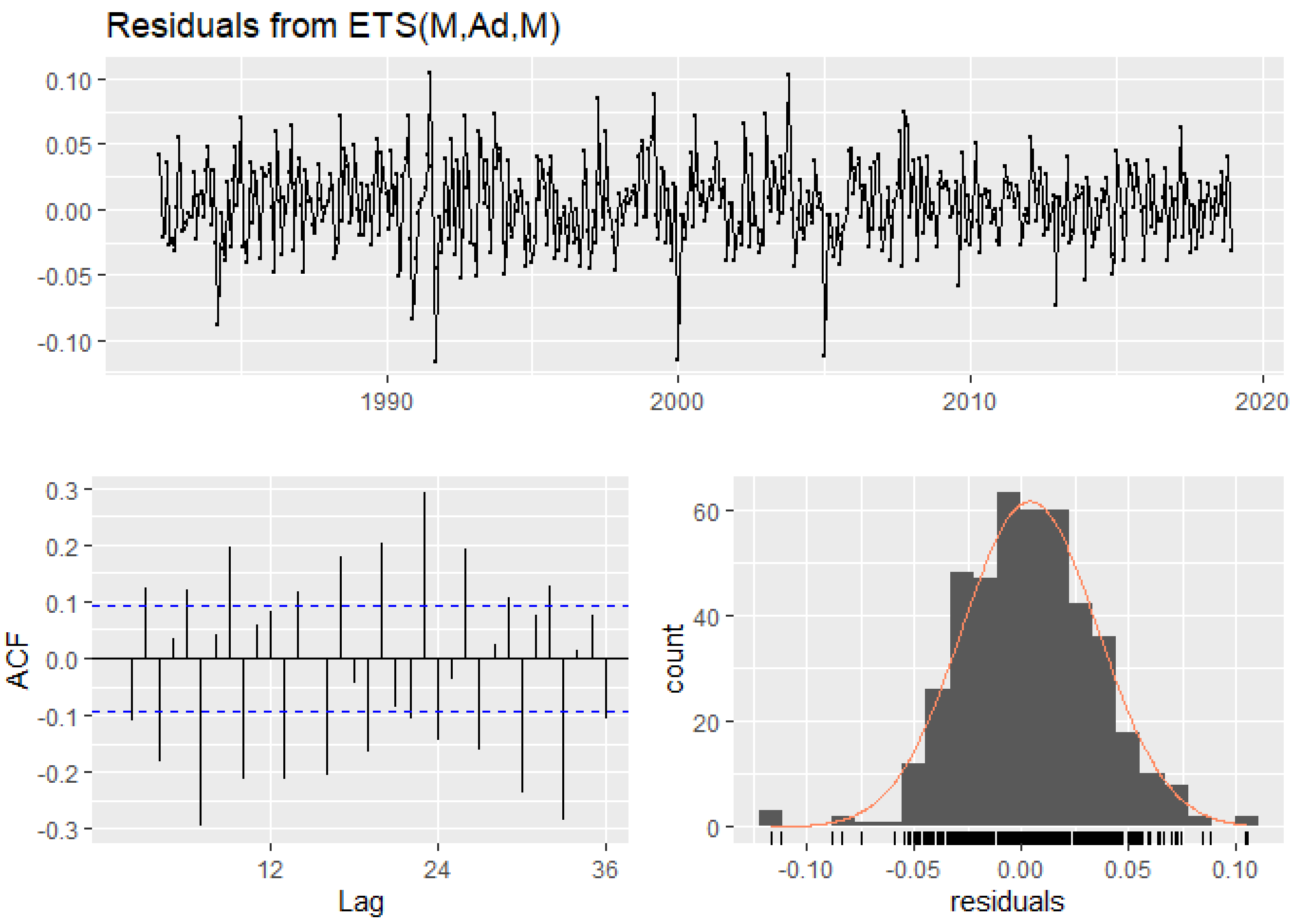
```
##
##  Ljung-Box test
##
## data:  Residuals from ETS(M,A,M)
## Q* = 256.59, df = 8, p-value < 2.2e-16
##
## Model df: 16.   Total lags used: 24
```

As indicated by the ACF plot there are multiple lag plots which are significant. Moreover, the Ljung Box test also shows that the residuals are significant at any level of significance and thus rejects the NULL (p value approx 0). As can be seen from residuals graph there is no major skewness in the residual distribution,although the auto correlation test is significant which means prediction intervals are not reliable, thus I will go with this model for point forecasts.

```
fit2_new<-ets(myts,model = 'MAM',damped=T)
summary(fit2_new)
```

```
## ETS(M,Ad,M)
##
## Call:
##  ets(y = myts, model = "MAM", damped = T)
##
##   Smoothing parameters:
##     alpha = 0.4874
##     beta  = 0.0229
##     gamma = 0.0769
##     phi   = 0.98
##
##   Initial states:
##     l = 126.0723
##     b = 1.38
##     s = 0.967 0.8903 0.9384 1.2446 1.0159 1.0142
##            0.9909 1.0234 1.0024 0.9677 0.9996 0.9456
##
##   sigma:  0.0326
##
##      AIC     AICc      BIC
## 5168.153 5169.774 5241.756
##
## Training set error measures:
##                    ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 1.829648 21.58025 14.81845 0.2934406 2.479641 0.3532515
##                      ACF1
## Training set -0.09585978
```

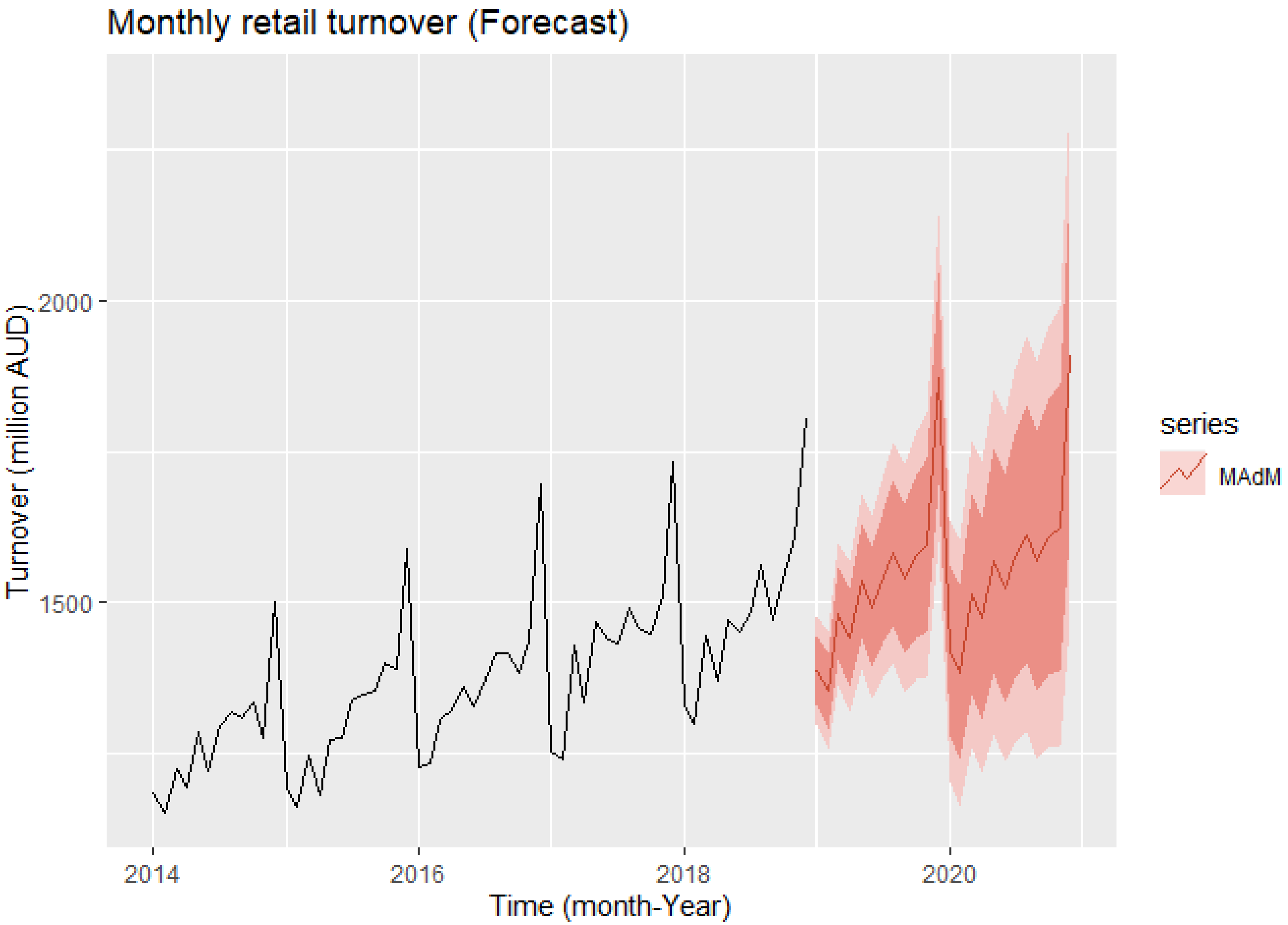
```
checkresiduals(fit2_new)
```



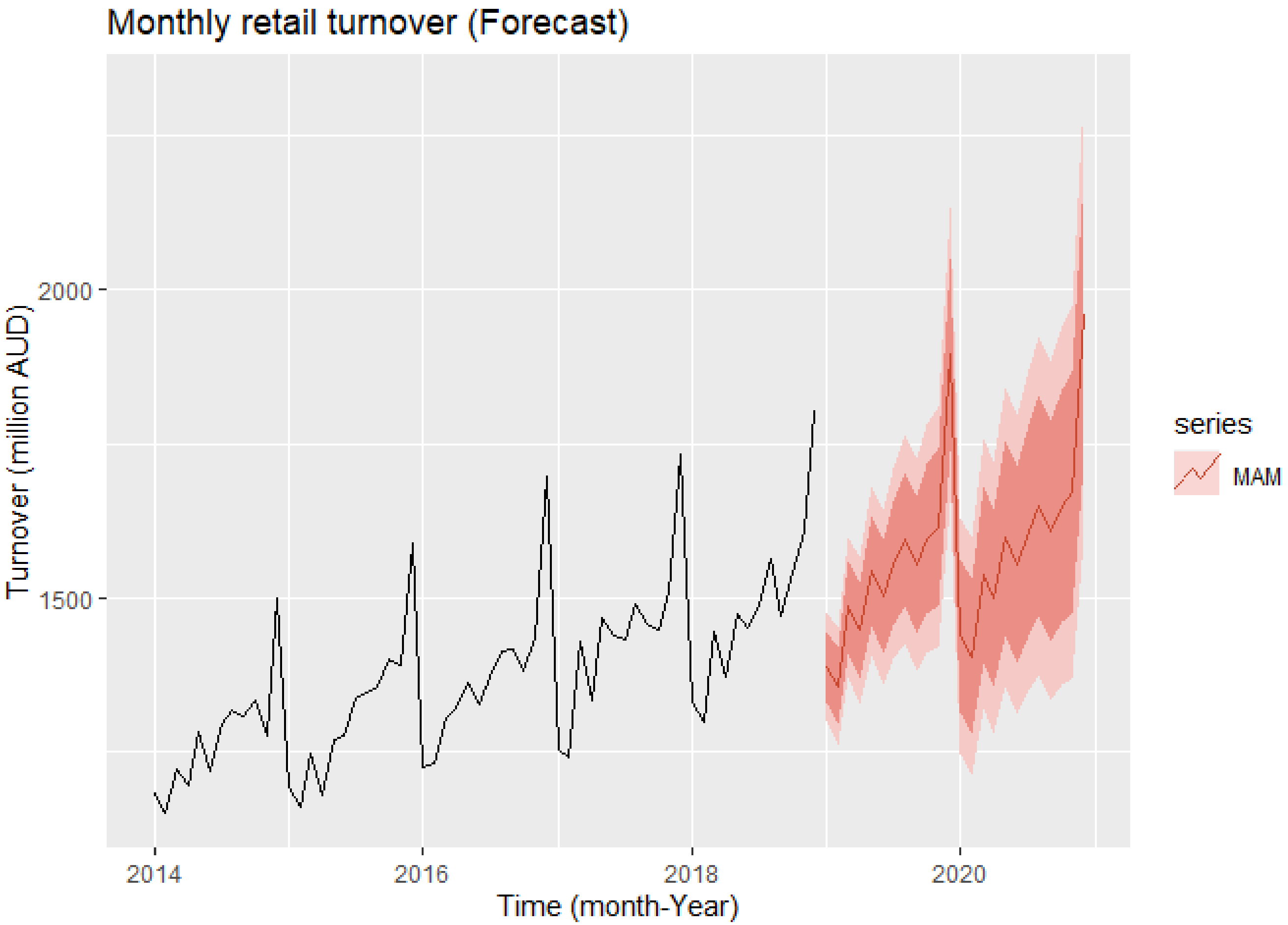
```
##
##  Ljung-Box test
##
```

```
## data:  Residuals from ETS(M,Ad,M)
## Q* = 268.73, df = 7, p-value < 2.2e-16
##
## Model df: 17.   Total lags used: 24
```

```
f2_1<-forecast(fit2_new,h=24)
f2_11<-forecast(fit_Assign3,h=24)
autoplot(window(myts,start=c(2014,1)),xlab="Time (month-Year)",ylab="Turnover (million AUD)", main="Monthly retail tu
  autolayer(f2_1, series="MAdM")
```



```
autoplot(window(myts,start=c(2014,1)),xlab="Time (month-Year)",ylab="Turnover (million AUD)", main="Monthly retail tu
  autolayer(f2_11, series="MAM")
```

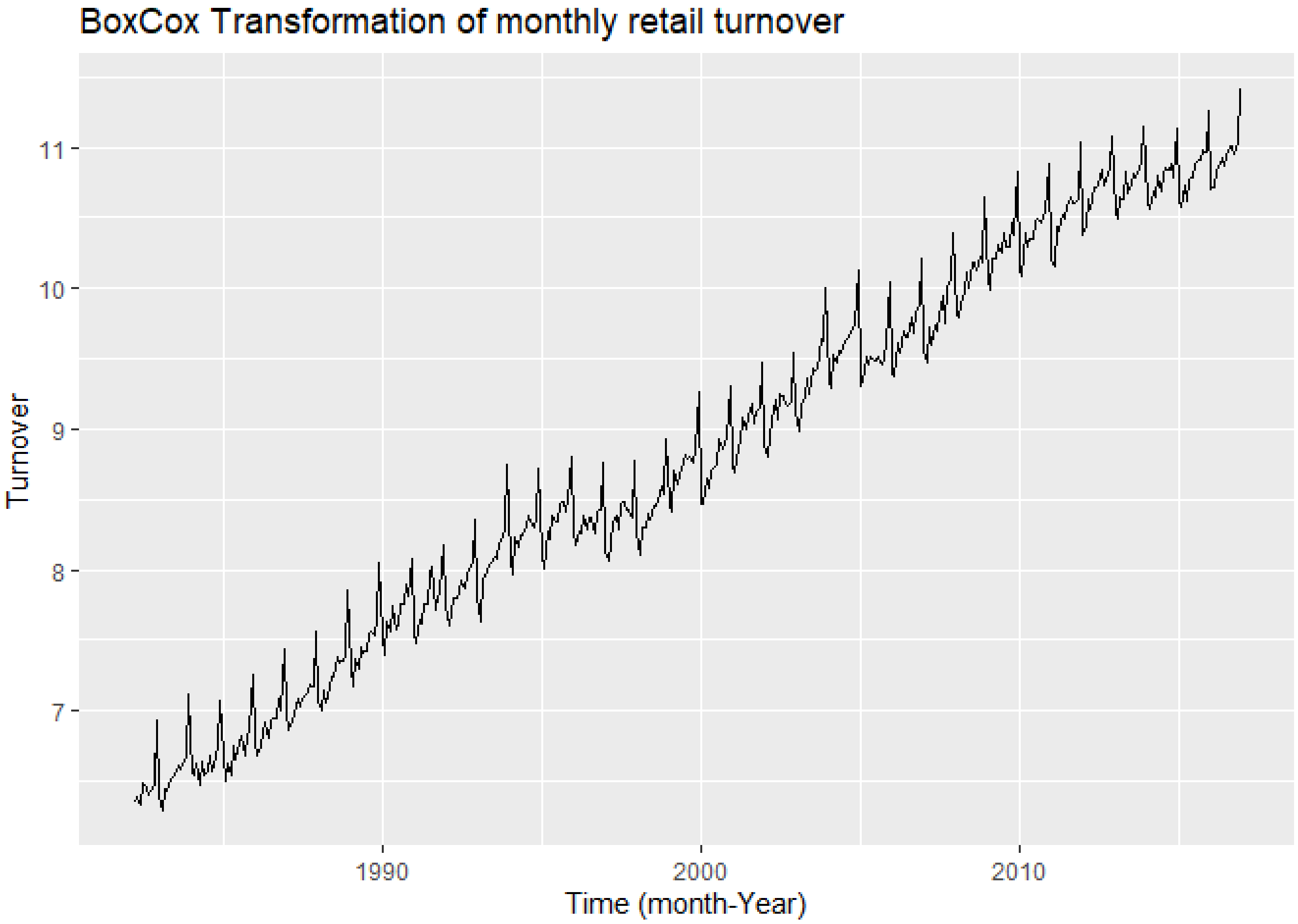


As shown by the graph the damped model has slightly higher prediction interval than the non damped model. On the other hand the forecasts from MAM are slightly higher compared to MAdM model because of the damping factor introduced by MAdM however the difference is very small due to the damping factor (phi) of 0.98 in the damped model thus both models look plausible.

ARIMA Models

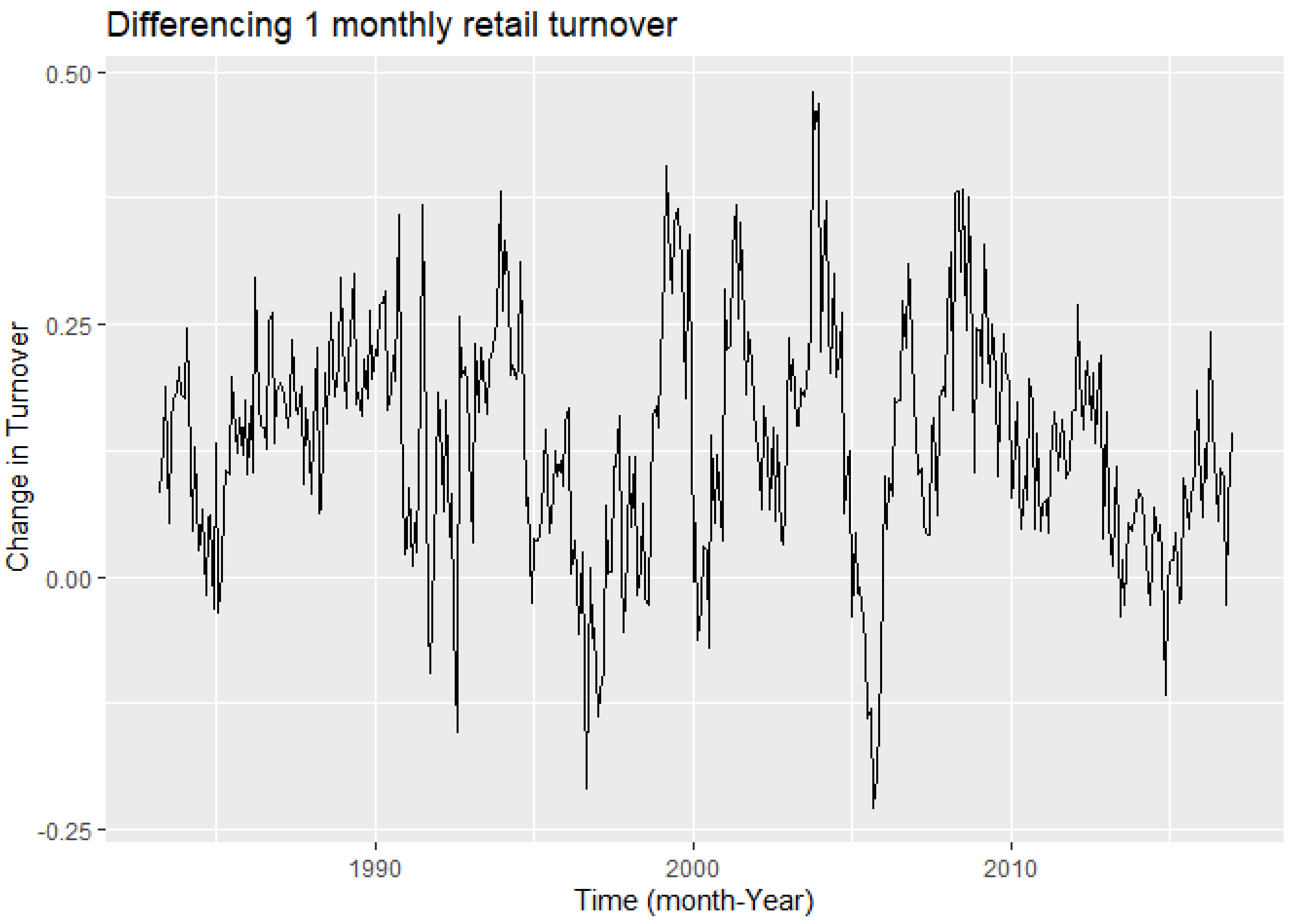
Now testing with ARIMA Models

```
lam=BoxCox.lambda(train)
autoplot(BoxCox(train,lambda=lam),xlab="Time (month-Year)",ylab="Turnover", main="BoxCox Transformation of monthly re
```



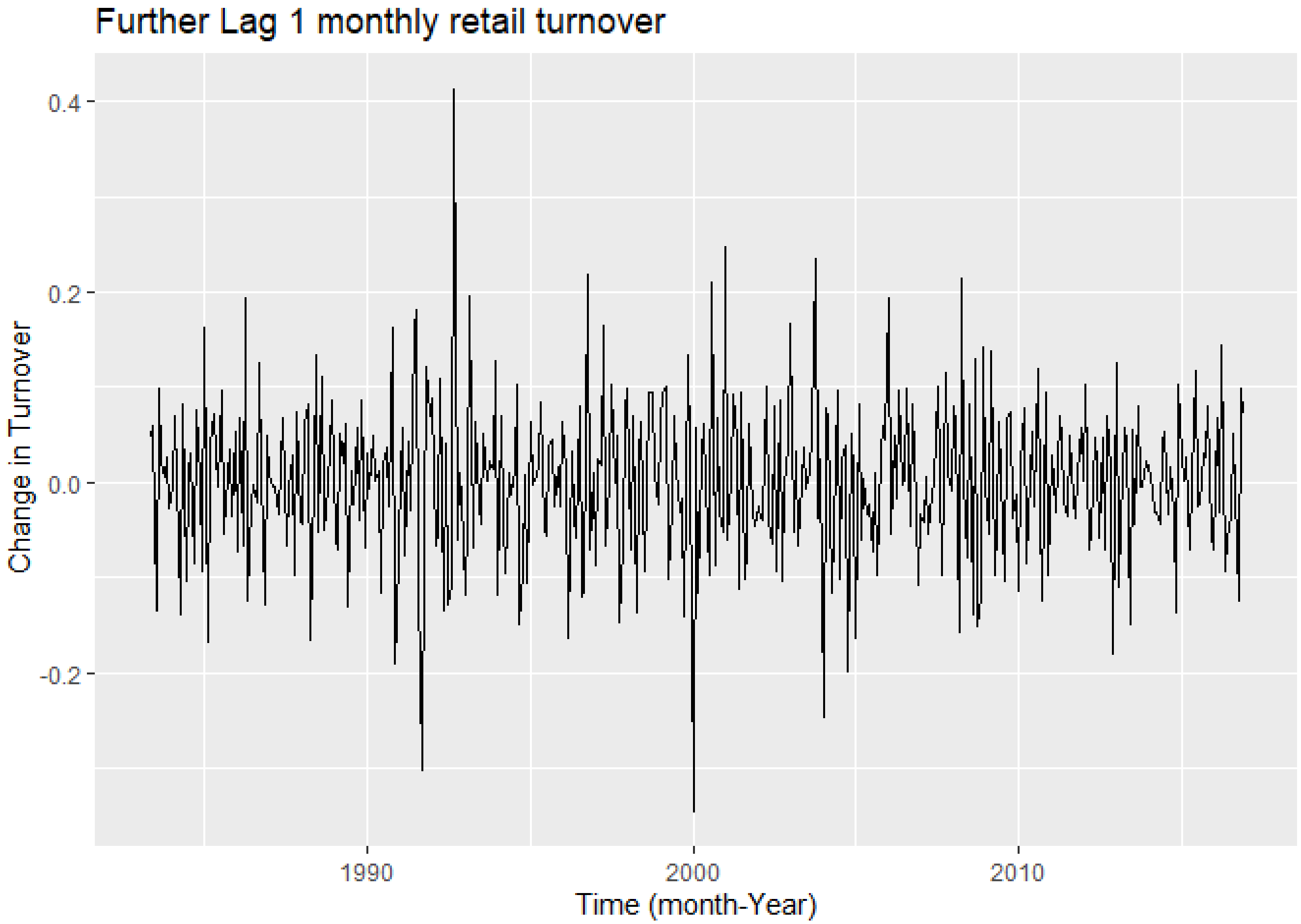
Variance Stabilization: In the first step box cox transformation was chosen to remove heteroskedasticity from the data. This makes the turnover data homoscedastic as shown above.

```
myts_new<-BoxCox(train,lambda=lam)
myts_sta1<-diff(myts_new,lag=12)
autoplot(myts_sta1,xlab="Time (month-Year)",ylab=" Change in Turnover", main="Differencing 1 monthly retail turnover"
```



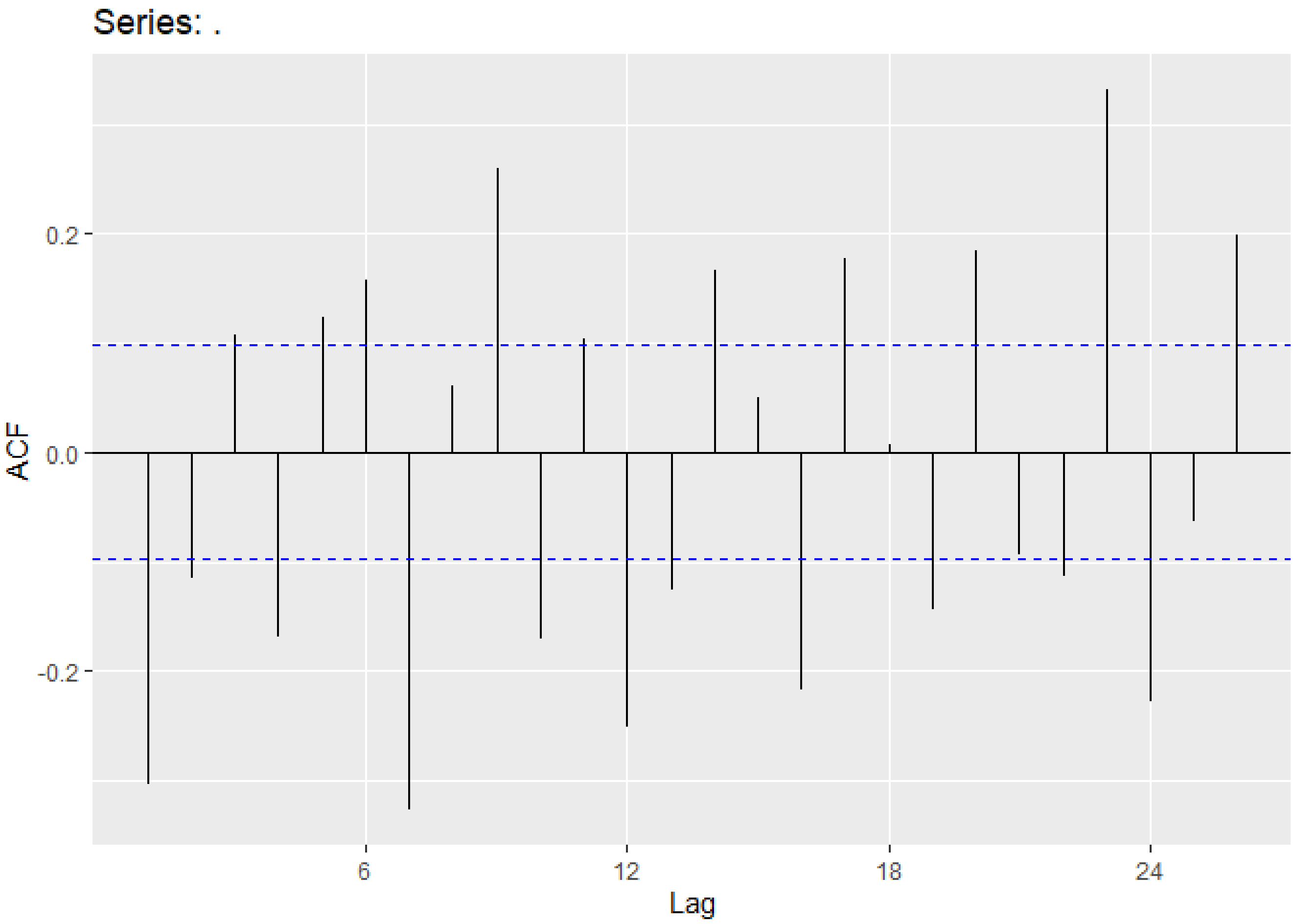
First order seasonal Difference: The attached figures shows first order difference of the transformed data at Lag 12. It shows slight evidence of the sinusoidal behaviour.

```
myts_sta2<-diff(myts_sta1,lag =1)
autoplot(myts_sta2,xlab="Time (month-Year)",ylab=" Change in Turnover", main="Further Lag 1 monthly retail turnover")
```

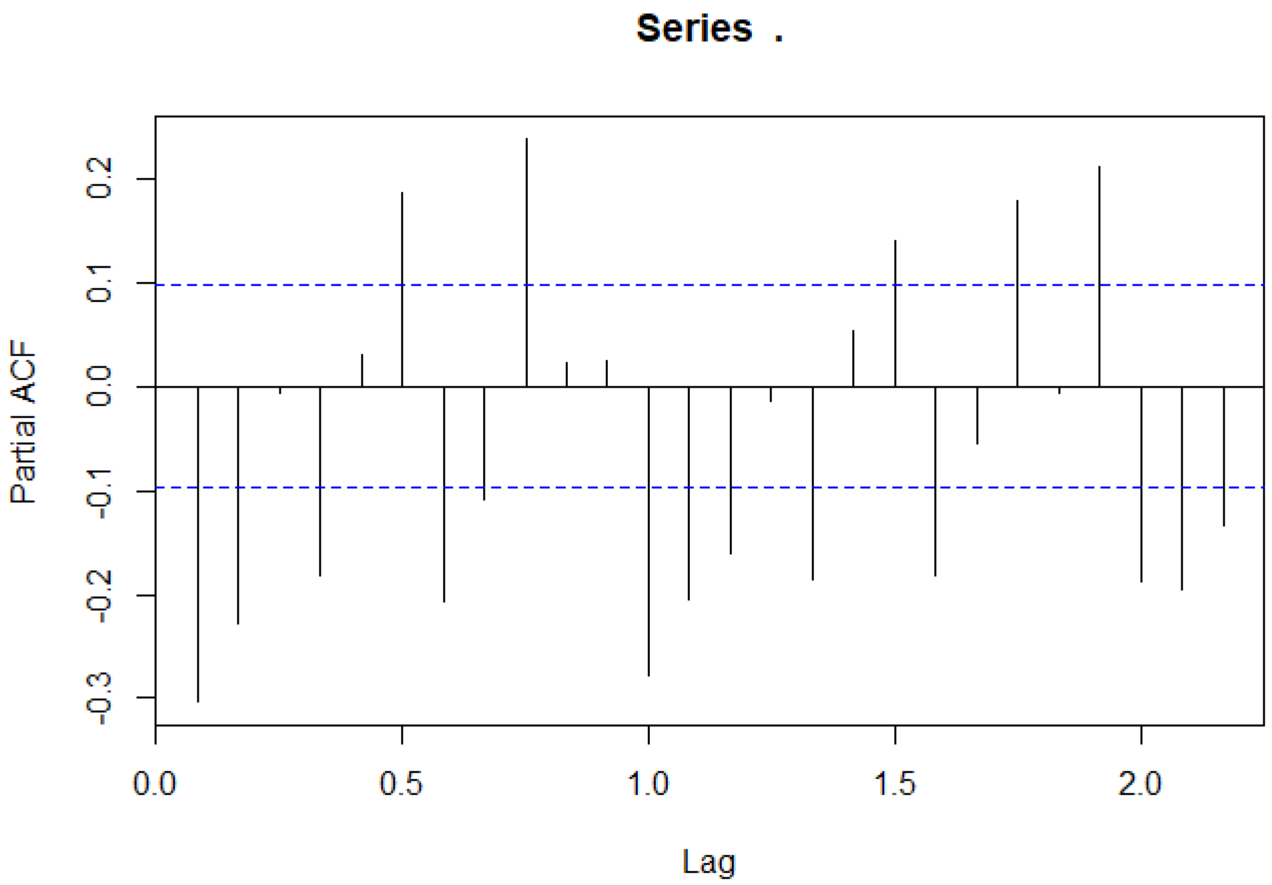


Further lag 1 difference: A further lag 1 difference has been to remove the slight evidence of sinusoidal behaviour therefore the figure 3 now shows data to be highly stationary with no pattern of change.

```
myts_sta2 %>% ggAcf()
```

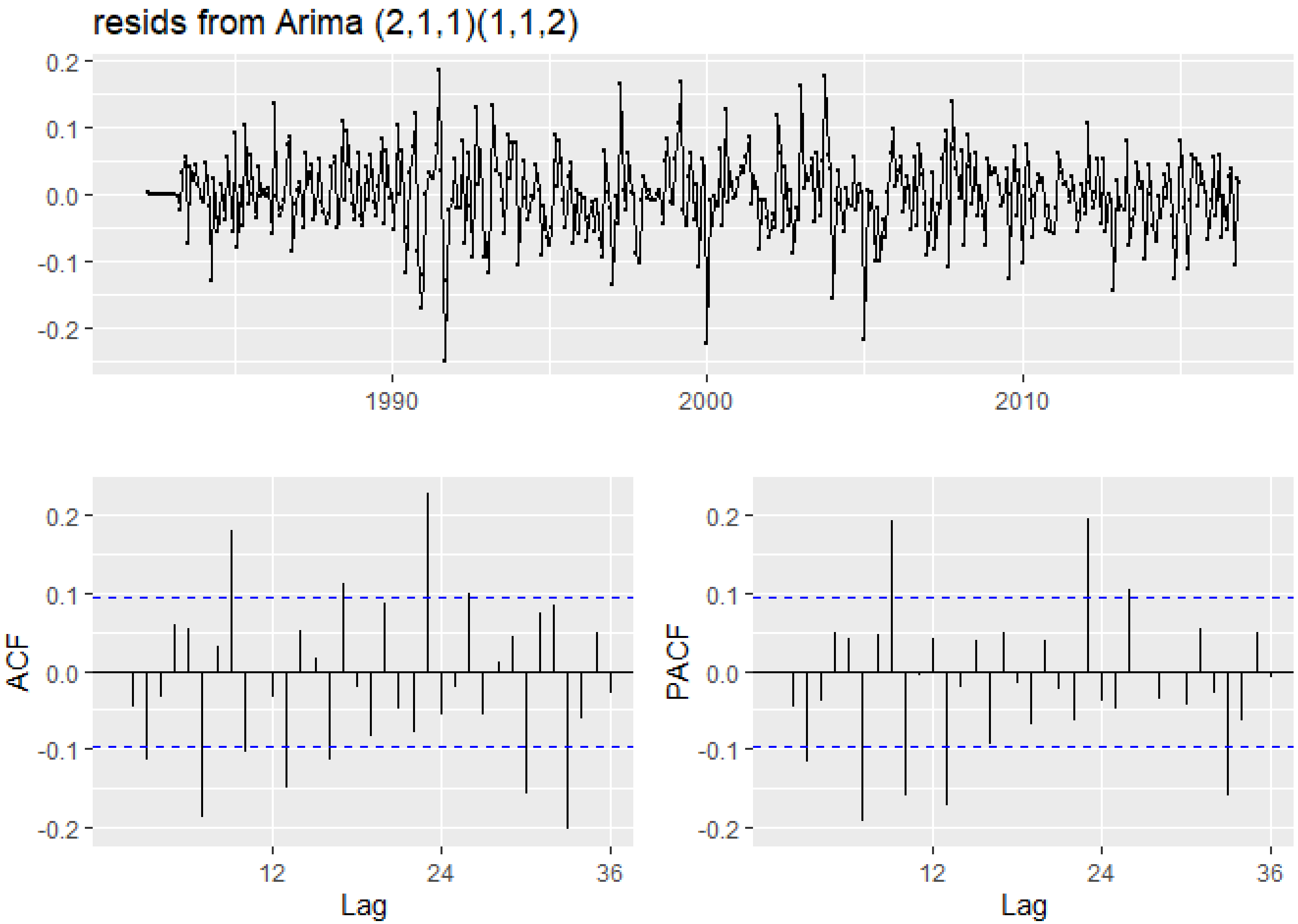


```
myts_sta2 %>% pacf()
```



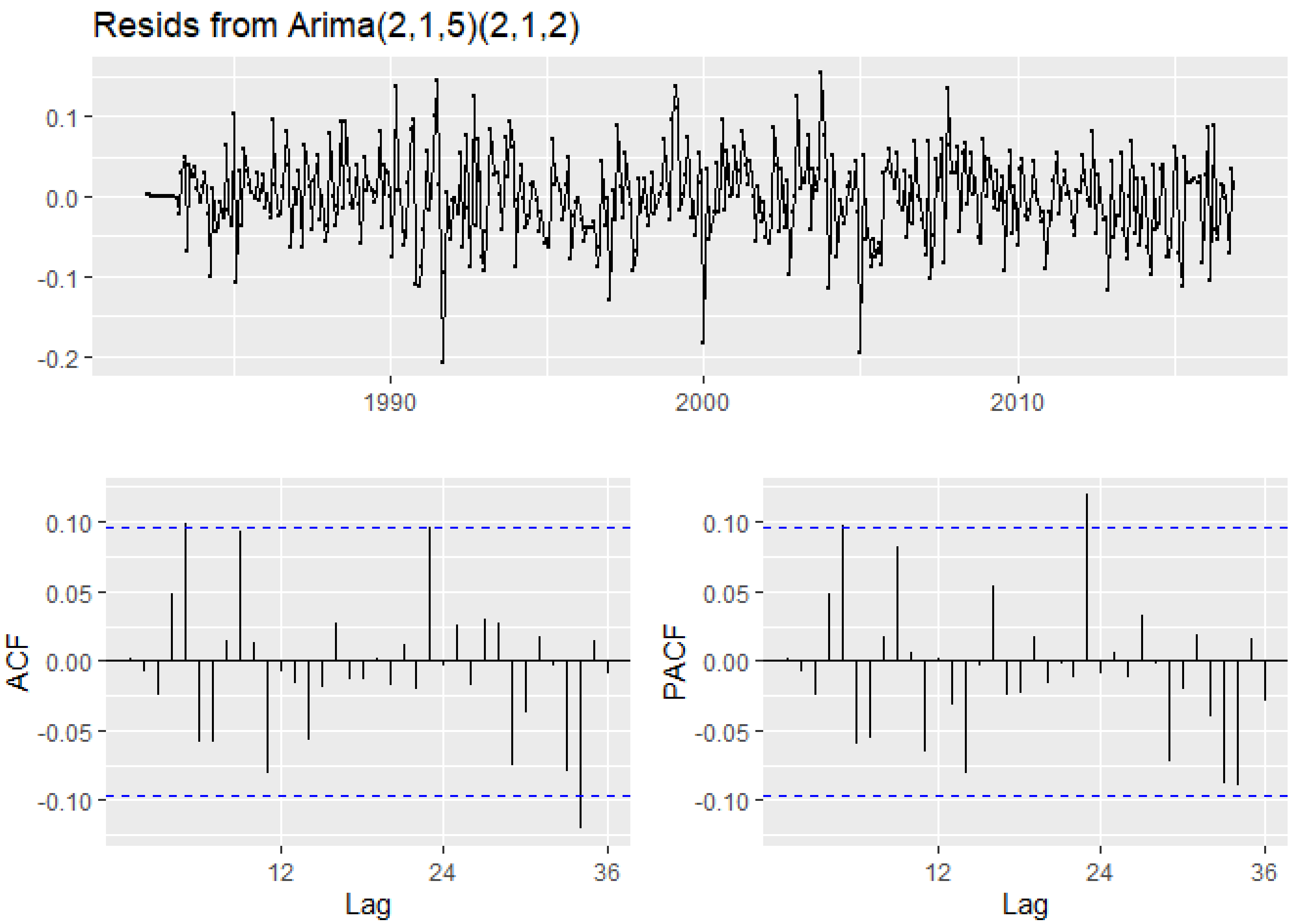
Since transformation involved 1 seasonal difference and 1 lagged difference therefor d=1 and ds=1. From the PACF it can be seen there are 2 significant successive lags and 1 significant seasonal lag so p=2 and ps=1. For ACF, there is 1 significant successive lag (second being just significant) and 2 consecutive seasonal lags therefore Q=1 and Qs=2. Thus suggested model is Arima (2,1,1)(1,1,2).

```
fit1<-Arima(train ,order=c(2,1,1),seasonal=c(1,1,2),lambda=lam)
fit1 %>% residuals() %>% ggtsdisplay(main="resids from Arima (2,1,1)(1,1,2)")
```

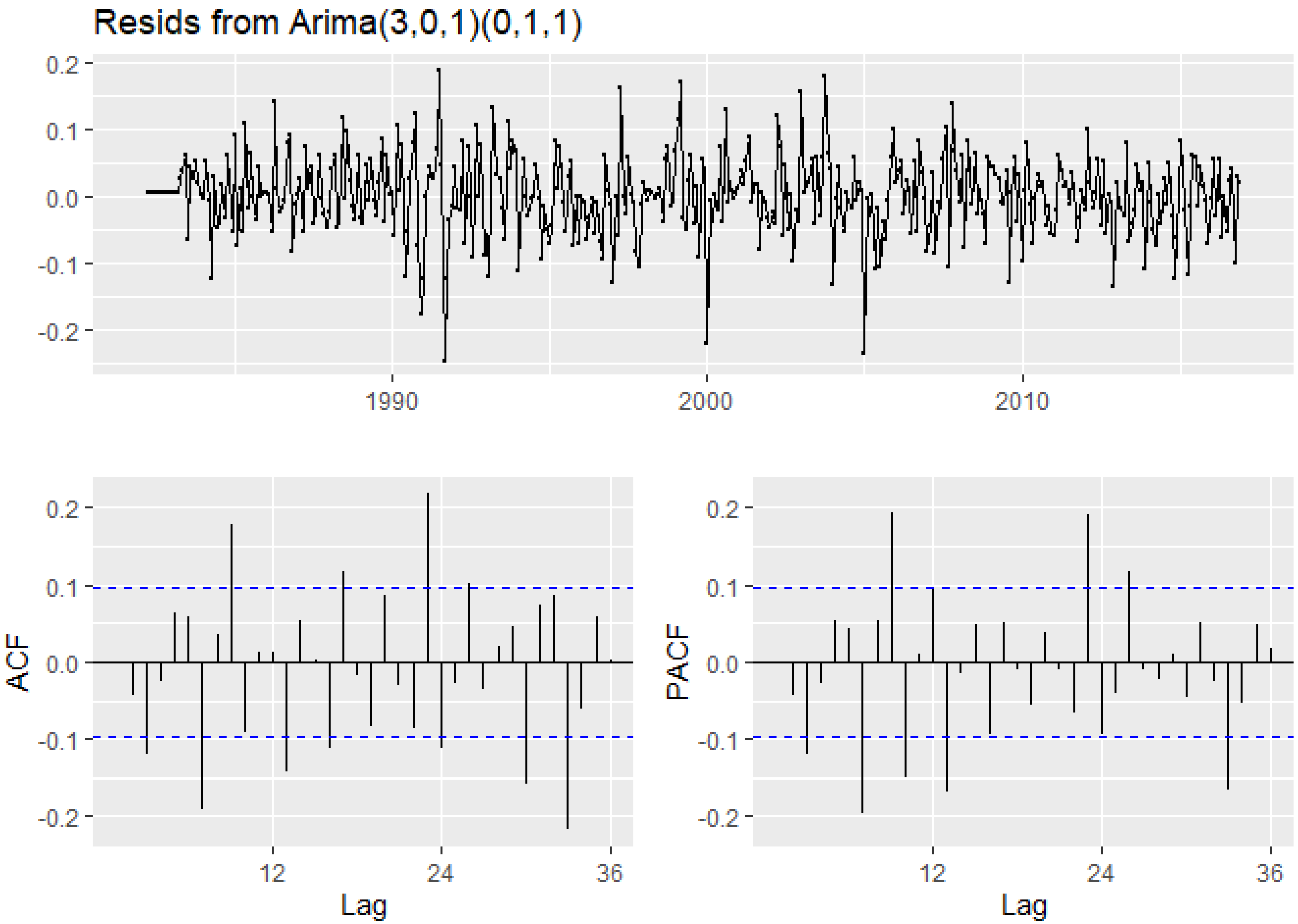


Trying out a few more ARIMA Models

```
Arima(train ,order=c(2,1,5),seasonal=c(2,1,2),lambda=lam)%>% residuals() %>% ggtsdisplay(main="Resids from Arima(2,1,
```

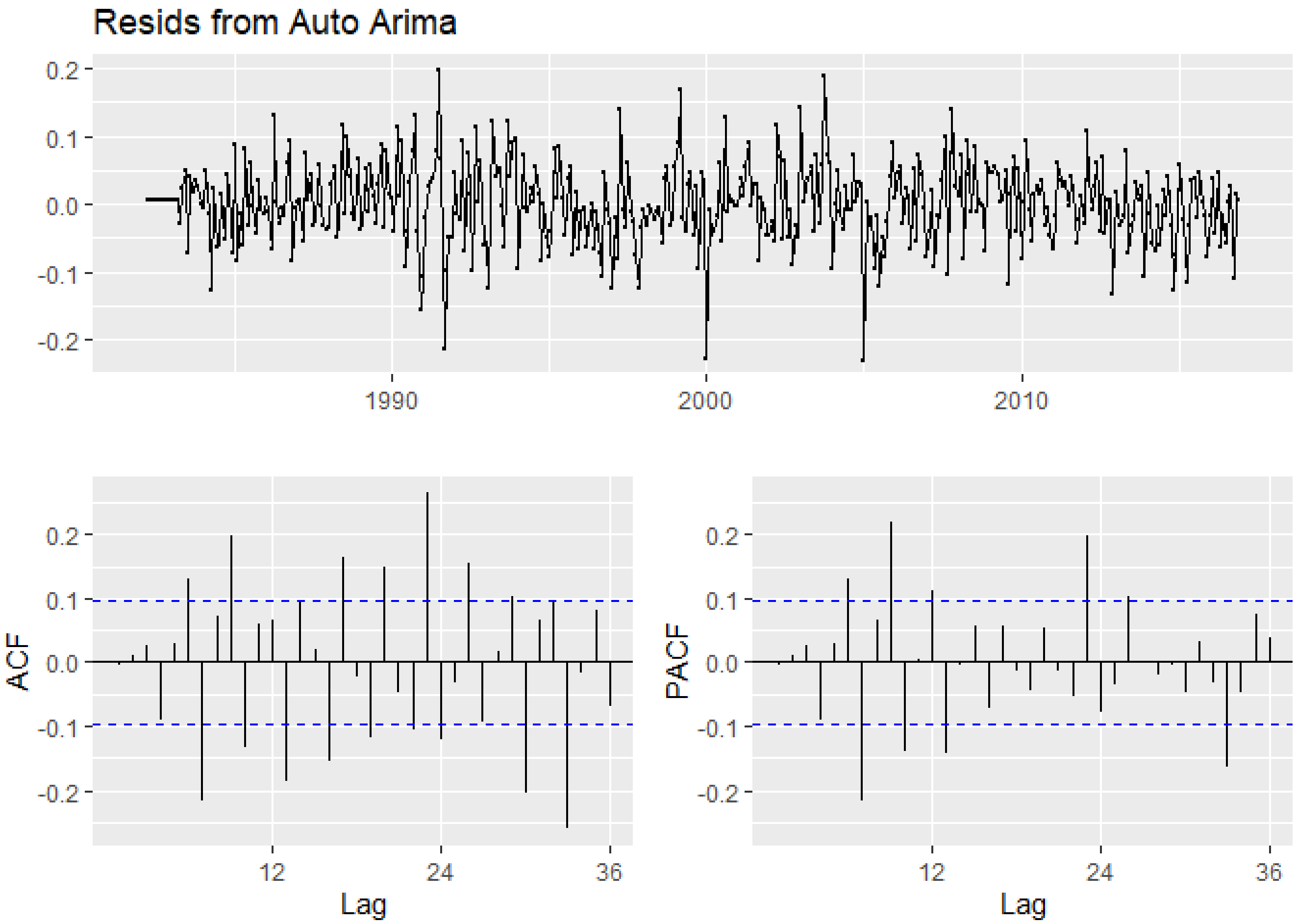


```
Arima(train ,order=c(3,0,1),seasonal=c(0,1,1),lambda=lam)%>% residuals() %>% ggtsdisplay(main="Resids from Arima(3,0,
```



Checking with Auto ARIMA

```
auto.arima(train,lambda=lam)%>% residuals() %>% ggtsdisplay(main="Resids from Auto Arima")
```



```
getrmse <- function(x,h,...)
{
  train.end <- time(x)[length(x)-h]
  test.start <- time(x)[length(x)-h+1]
  train <- window(x,end=train.end)
  test <- window(x,start=test.start)
  fit <- Arima(train,...)
  fc<-forecast(fit,h=h)
  return(round(accuracy(fc,test)[2,"RMSE"],4))
}
getrmse(train,h=24,order=c(2,1,5),seasonal=c(2,1,2),lambda=lam)
```

```
## [1] 27.7317
```

```
getrmse(train,h=24,order=c(3,0,1),seasonal=c(0,1,1),lambda=lam)
```

```
## [1] 42.0358
```

```
getrmse(train,h=24,order=c(2,1,1),seasonal=c(1,1,2),lambda=lam)
```

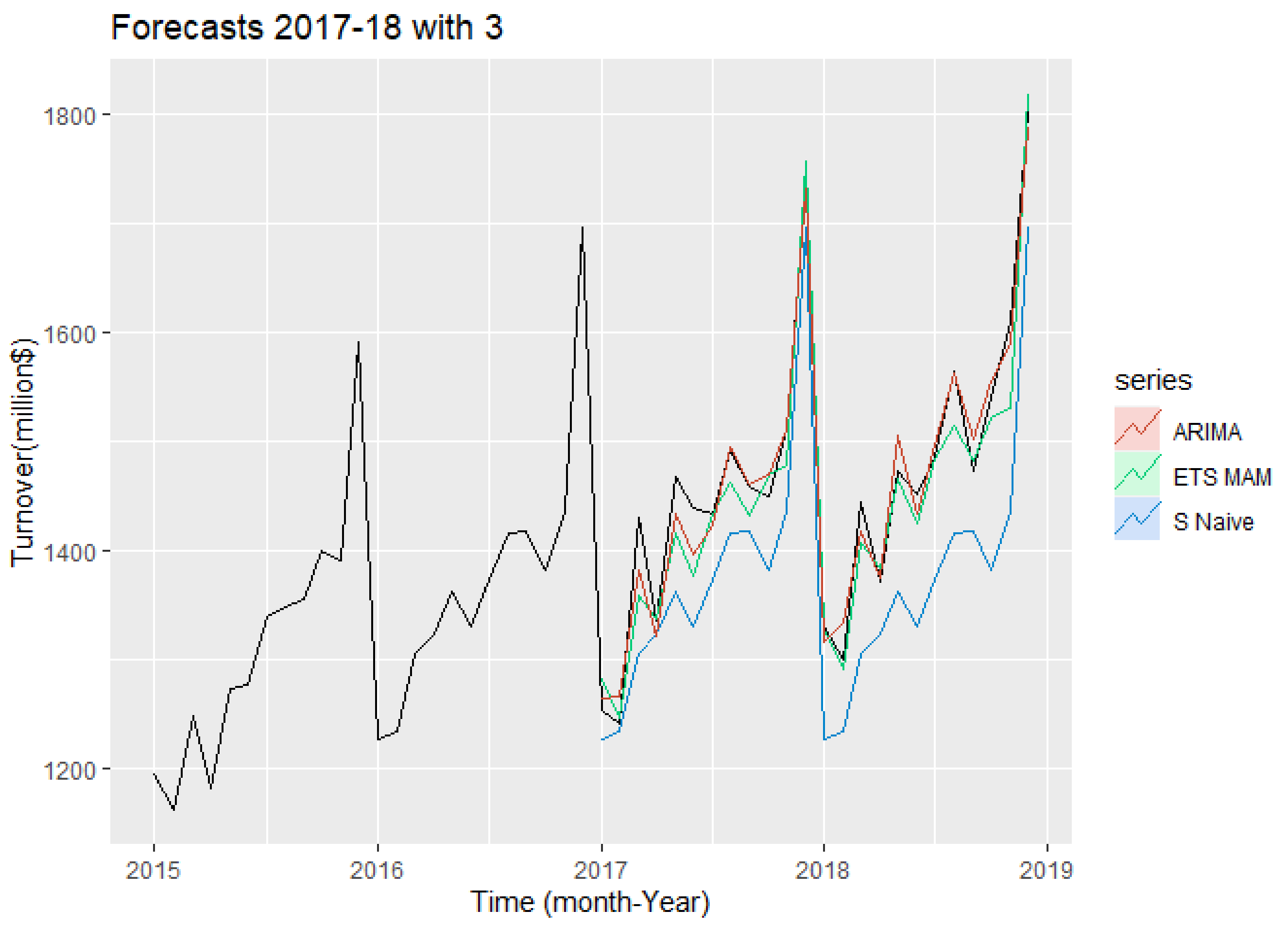
```
## [1] 38.7381
```

Using last 2 years data as the test set and the RMSE method. The RMSE is compared with few other arima methods and my chosen midel is ARIMA(2,1,5)(2,1,2).

Final Model

```
fit_choose1<-Arima(train ,order=c(2,1,5),seasonal=c(2,1,2),lambda=lam)
fit_choose2<-ets(train,model = 'MAM')
fit_benchmark<- snaive(train, h=24,lambda=lam)

autoplot(window(myts,start=c(2015,1)),xlab="Time (month-Year)",ylab="Turnover(million$)", main="Forecasts 2017-18 wit
  autolayer(forecast(fit_choose2,h=24),PI=FALSE,series="ETS MAM")+
  autolayer(fit_benchmark,PI=FALSE,series="S Naive")+
  autolayer(forecast(fit_choose1,h=24),PI=FALSE,series="ARIMA")
```



As shown the ARIMA Model most closely replicates the original time series data. Therefore my chosen model for retail data is the ARIMA model.