UK Retail Sales Example

Jiantong Wang

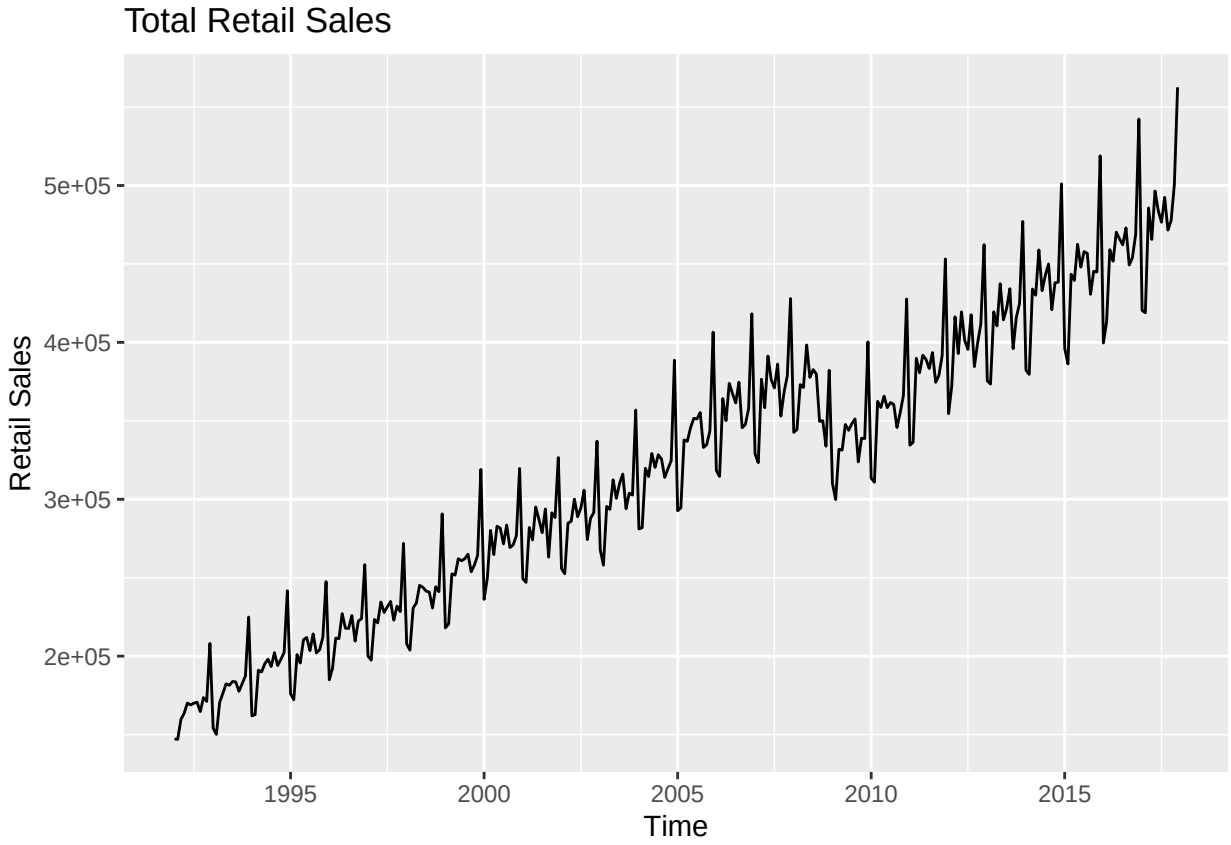
Data Set and Research Question

The data set we used is the monthly UK Retail Sales data from January 1992 to December 2017. There are in total 312 observations. There are 5 variables, including Year, Month, Sales (in millions), S_Factor(Season Adjust Factor) and CPI. In our project, we are focusing on the sales amount, and aiming at fitting a model to predict the retail sales in the future.

Data Visualization

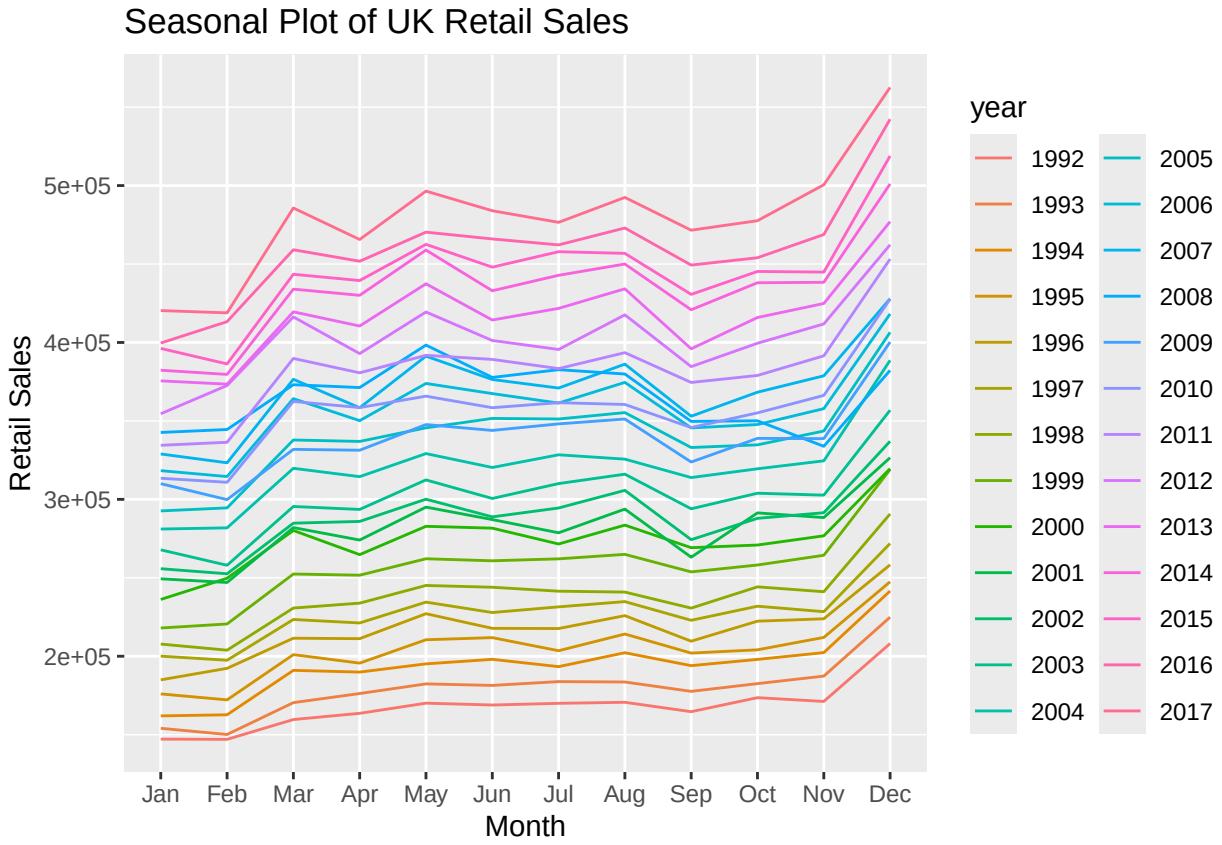
Time series plot

We can first make a time series plot to take an overview of the data.



From the plot, we can notice that over the years, there is an increasing pattern of the retails sales in UK. There is also an annual seasonal pattern. For each year, there is a high peak and low valley of retail sales amount. We can make a seasonal plot of the data for the detailed seasonal pattern.

Seasonal Plot of the UK Retail Sales



From the seasonal plot above, we can find that for each year, the sales amount achieve its highest value in December, and it ahieves the minimum value in February.

Model Development

When developing the models, we use the data from Jan 1992 to December 2016 as training data and left the data in the year of 2017 as testing data for model performance.

```
Retail.training.ts <- window(Retail.ts,start = c(1992,1), end = c(2016,12))
Retail.testing.ts <- window(Retail.ts,start = c(2017,1), end = c(2017,12))
```

State Space Model

From previous part, we know that there are seasonal effects of our data. To include seasonal effects, Holt-Winters model would be the proper model to fit. (We can use ets() function to fit a stat space model, if you don't specify the model structure, ets() function will automatically fit the best model for it.)

```
## ETS(M,A,M)
##
## Call:
##  ets(y = Retail.training.ts, damped = F)
##
##   Smoothing parameters:
##     alpha = 0.3744
##     beta  = 0.1035
##     gamma = 0.1799
```

```
##
##   Initial states:
##     l = 161254.0357
##     b = 1130.6076
##     s = 1.2004 1.01 0.9983 0.9681 1.0208 1.0004
##           1.0062 1.0334 0.9847 0.9916 0.8928 0.8934
##
##   sigma:  0.0205
##
##      AIC     AICc      BIC
## 6962.999 6965.169 7025.964
```

The state space model we fit is "MAM" structure. We have the parameters: alpha = 0.43, beta = 0, gamma = 0.21. The model has a fixed trend term.

ARIMA model

Stationarity Test

We can use ADF test to test whether our data is stationary. Our null hypothesis is that our data is not stationary and the alternative hypothesis is our data is stationary.

```
adf.test(Retail.training.ts)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  Retail.training.ts
## Dickey-Fuller = -2.4829, Lag order = 6, p-value = 0.3728
## alternative hypothesis: stationary
```

We get a test results with P-value greater than 0.05 therefore, we fail to reject the null hypothesis. Therefore, our data is non-stationary, so we should consider fitting an ARIMA model instead of ARMA model.

Model fitting

```
model2 <- auto.arima(Retail.training.ts)
model2
```

```
## Series: Retail.training.ts
## ARIMA(1,0,1)(2,1,2)[12] with drift
##
## Coefficients:
##          ar1      ma1    sar1     sar2     sma1    sma2      drift
##       0.9636  -0.3917  0.8179  -0.6678  -1.2075  0.6015  1063.8602
## s.e.  0.0177   0.0554  0.0732   0.0757   0.0892  0.0773   217.0545
##
## sigma^2 = 37155108:  log likelihood = -2922.14
## AIC=5860.29   AICc=5860.8   BIC=5889.59
```

The model we fit is a seasonal ARIMA model. The seasonal components follows an ARIMA(2,1,2) model and the regular components follows an ARIMA(1,0,1) model.

Model Comparison

Out-of-sample RMSE

We can based on the performance of prediction on the testing data to determine which model to use. Here we use the data from Jan 2017 to Dec 2017 as the testing data, and use model1 and model2 to make predictions for the year of 2017.

Based on model1, we can have the point estimations for each month in the year of 2017 as:

```
prediction_1 <- forecast(model1,h=12)
prediction_1$mean
```

```
##           Jan      Feb      Mar      Apr      May      Jun      Jul      Aug
## 2017 427506.1 431607.7 490385.9 483339.1 510199.1 496485.4 500613.1 511255.3
##           Sep      Oct      Nov      Dec
## 2017 479758.0 494158.7 502893.1 581012.2
```

Based on model2, we can have the point estimations for each month in the year of 2017 as:

```
prediction_2 <- forecast(model2,h=12)
prediction_2$mean
```

```
##           Jan      Feb      Mar      Apr      May      Jun      Jul      Aug
## 2017 429323.9 442183.4 488526.3 476745.0 496486.7 484719.0 481804.1 496804.2
##           Sep      Oct      Nov      Dec
## 2017 468878.0 478407.3 493876.3 557489.6
```

Therefore, the out-of-sample RMSE of model1 and model2 for the year of 2017 are: 14458.92 and 8564.449. Model2 has a lower out-of-sample RMSE, which indicates it performs better than model1 in prediction. So we choose model2, the ARIMA model with structure as ARIAM(1,0,1)(2,1,2)[12].

Information Criterion

We can also rely the Information Criterion to select the model. A model with lower information criterion is better. For the sample size is relatively larger, so we choose to use BIC. The BIC of model1 is 7025.964 and the BIC of model2 is 5889.589. So model 2 is better.

Summary

Our project aimed to predict future UK retail sales using data from January 1992 to December 2017. We identified for annual sales, there is a seasonal pattern with peaks in December and valleys in February.

We trained two models, a State Space Model (Holt-Winters) and an ARIMA model on data up to 2016 and tested them on data of 2017. Both out-of-sample RMSE and BIC criteria demonstrated the superior performance of the ARIMA model, making it our chosen model for forecasting UK retail sales. Future work could include additional model tuning and validation for broader applications.

Appendix

R Codes

```
# Read in the data
library(forecast)
library(tseries)
Retail <- read.csv("RetailSales2018_Clean.csv")
# Create time series object
Retail.ts <- ts(Retail$Sales,start = c(1992,1),frequency = 12)
# Draw time series plot
```

```r
autoplot(Retail.ts,main = "Total Retail Sales",
         xlab = "Time", ylab = "Retail Sales")
# Draw seasonal plot
ggseasonplot(Retail.ts, main = "Seasonal Plot of UK Retail Sales",
             ylab = "Retail Sales", xlab = "Month")

# Split the dataset into training data and testing data
Retail.training.ts <- window(Retail.ts,start = c(1992,1), end = c(2016,12))
Retail.testing.ts <- window(Retail.ts,start = c(2017,1), end = c(2017,12))

# Fit state space model
model1 <- ets(Retail.training.ts,damped = F)
model1

# Test stationarity
adf.test(Retail.training.ts)

# Fit ARIMA model
model2 <- auto.arima(Retail.training.ts)
model2

# Get point estimations from model 1
prediction_1 <- forecast(model1,h=12)
prediction_1$mean

# Get point estimations from model2
prediction_2 <- forecast(model2,h=12)
prediction_2$mean

# Calculate the RMSE
sqrt(mean( (Retail.testing.ts - prediction_1$mean)^2 ))
sqrt(mean( (Retail.testing.ts - prediction_2$mean)^2 ))

# BIC
BIC(model1)
BIC(model2)
```