Time Series Analysis

Oh Jian Hui

9/3/2021


```
library(itsmr); library(tidyverse); library(gridExtra)
df <- readRDS("../data/bike_clean_rep.rds")
tt <- 1:315
test_df <- df[316:365, ]
df <- df[tt, ]
xx <- df$rep_count
```

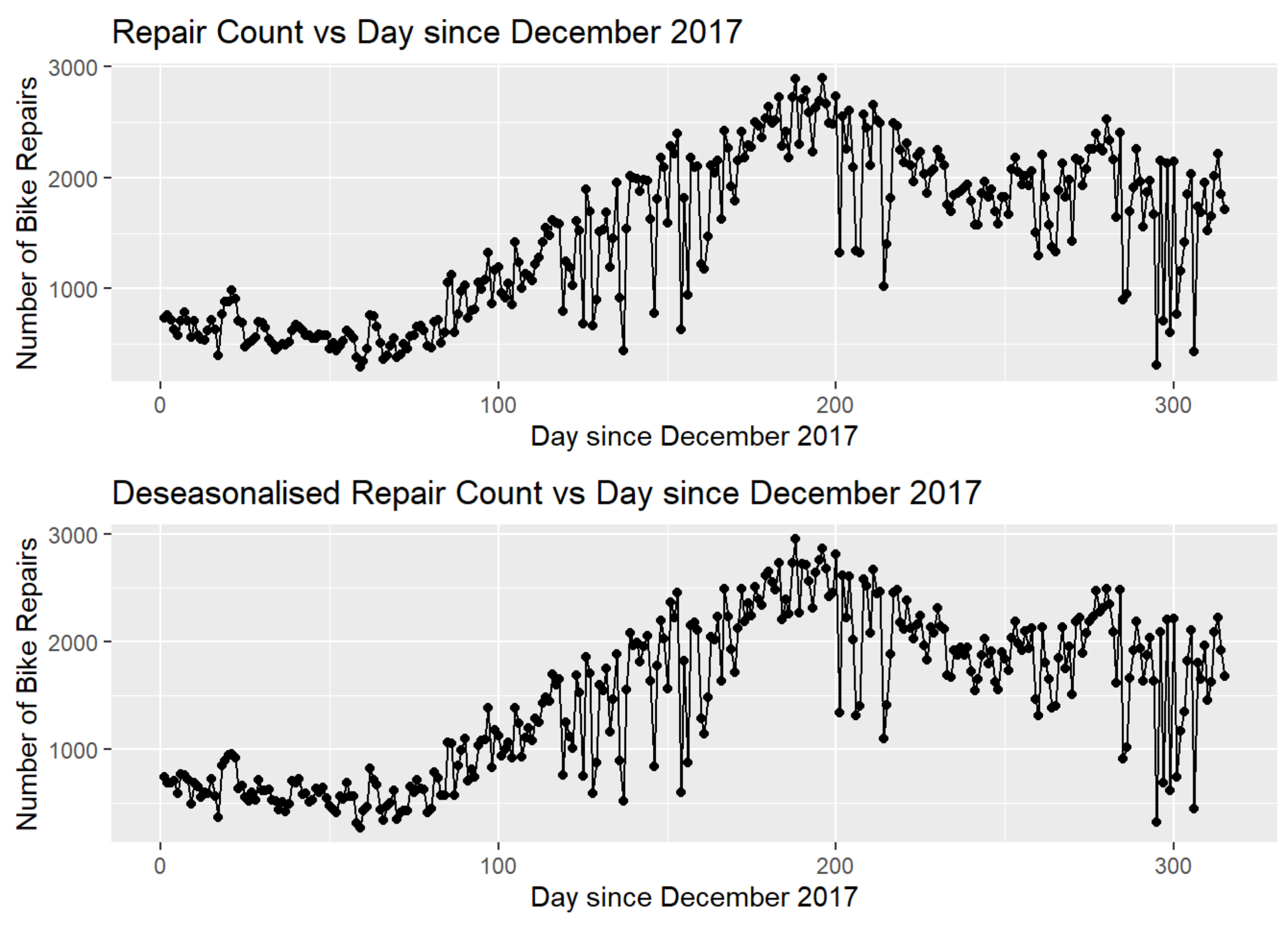
For this analysis, we will assume that the bike repair counts follow a Classical Time Series Decomposition model:

$$X_t = m_t + s_t + Y_t$$

where
X_t is the bike repair count random variable
m_t is the trend component
s_t is the seasonal component and
Y_t is the random noise with mean zero

Examine Seasonal Component

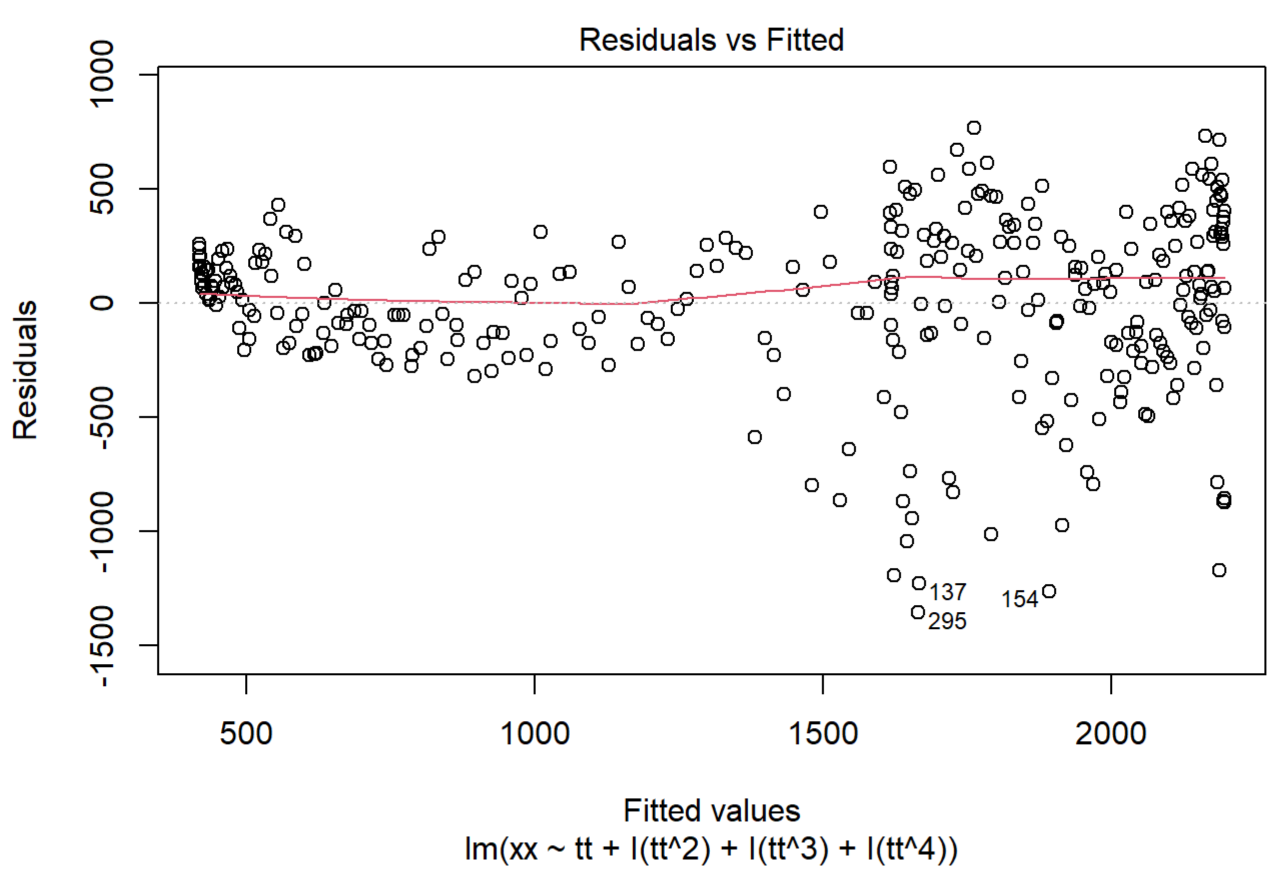
```
p1 <- ggplot(mapping = aes(x = tt, y = xx)) +
  geom_point() +
  geom_line() +
  labs(title = "Repair Count vs Day since December 2017",
       x = "Day since December 2017", y = "Number of Bike Repairs")
st <- season(xx, 7)
dt <- xx - st
p2 <- ggplot(mapping = aes(x = tt, y = dt)) +
  geom_point() +
  geom_line() +
  labs(title = "Deseasonalised Repair Count vs Day since December 2017",
       x = "Day since December 2017", y = "Number of Bike Repairs")
grid.arrange(p1, p2)
```



A 7 days seasonal period is very reasonable considering the bike usage is affected by the customers' weekly schedule.

Model Trend Component

```
md <- lm(xx ~ tt + I(tt ^ 2) + I(tt ^ 3) + I(tt ^ 4))
plot(md, which = 1); summary(md)
```

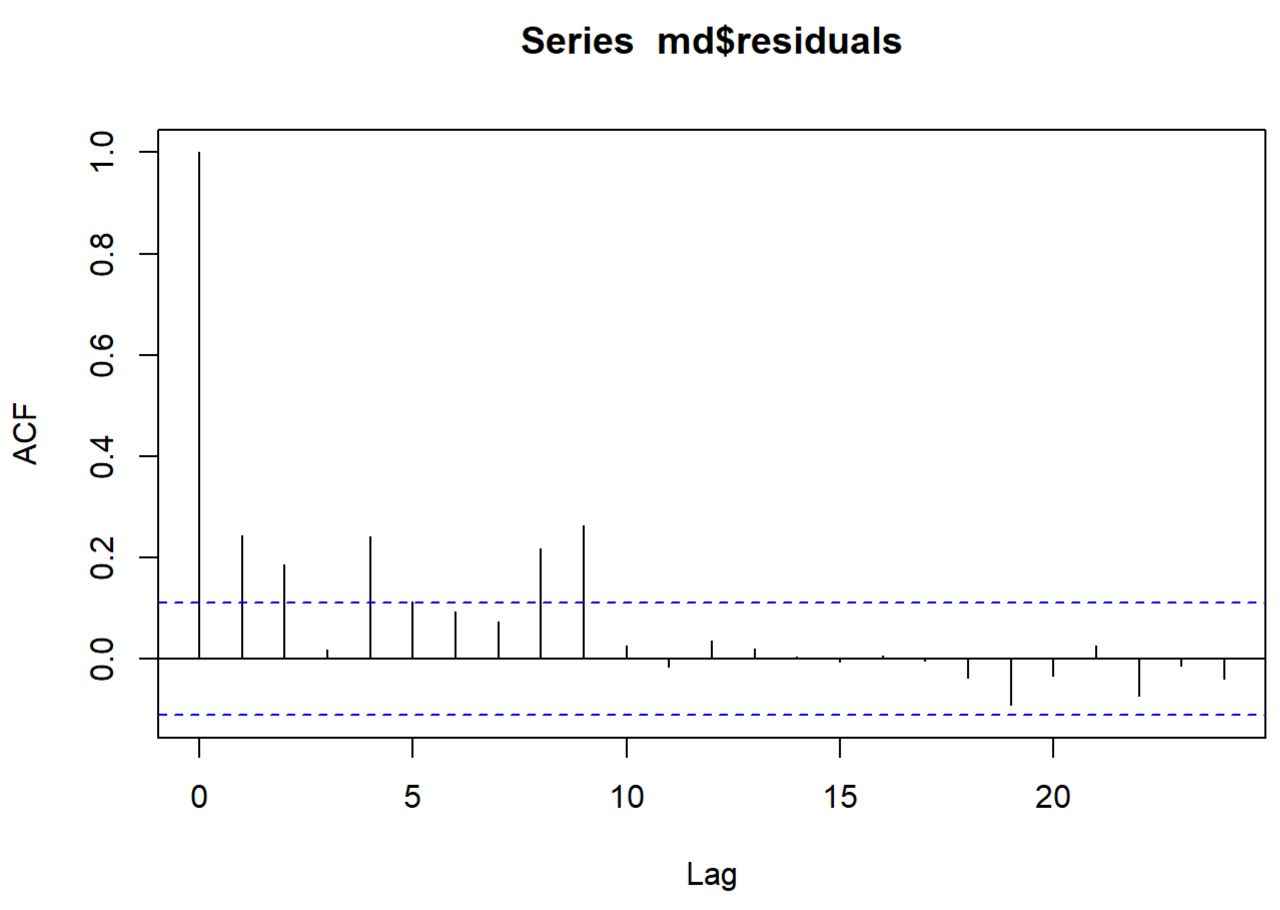



```
##
## Call:
## lm(formula = xx ~ tt + I(tt^2) + I(tt^3) + I(tt^4))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1355.45  -171.96    56.85   240.21   767.11
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.053e+03  1.062e+02   9.917  < 2e-16 ***
## tt          -3.429e+01  4.641e+00  -7.389 1.39e-12 ***
## I(tt^2)      5.520e-01  5.959e-02   9.262  < 2e-16 ***
## I(tt^3)     -2.406e-03  2.831e-04  -8.499 8.19e-16 ***
## I(tt^4)      3.231e-06  4.445e-07   7.268 2.99e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 369.8 on 310 degrees of freedom
## Multiple R-squared:  0.7421, Adjusted R-squared:  0.7388
## F-statistic:   223 on 4 and 310 DF,  p-value: < 2.2e-16
```

The adjusted R squared is 0.7388, indicating that the linear model is able to explain close to three quarters variation in data.

Model Noise

```
ACF <- acf(md$residuals)
```

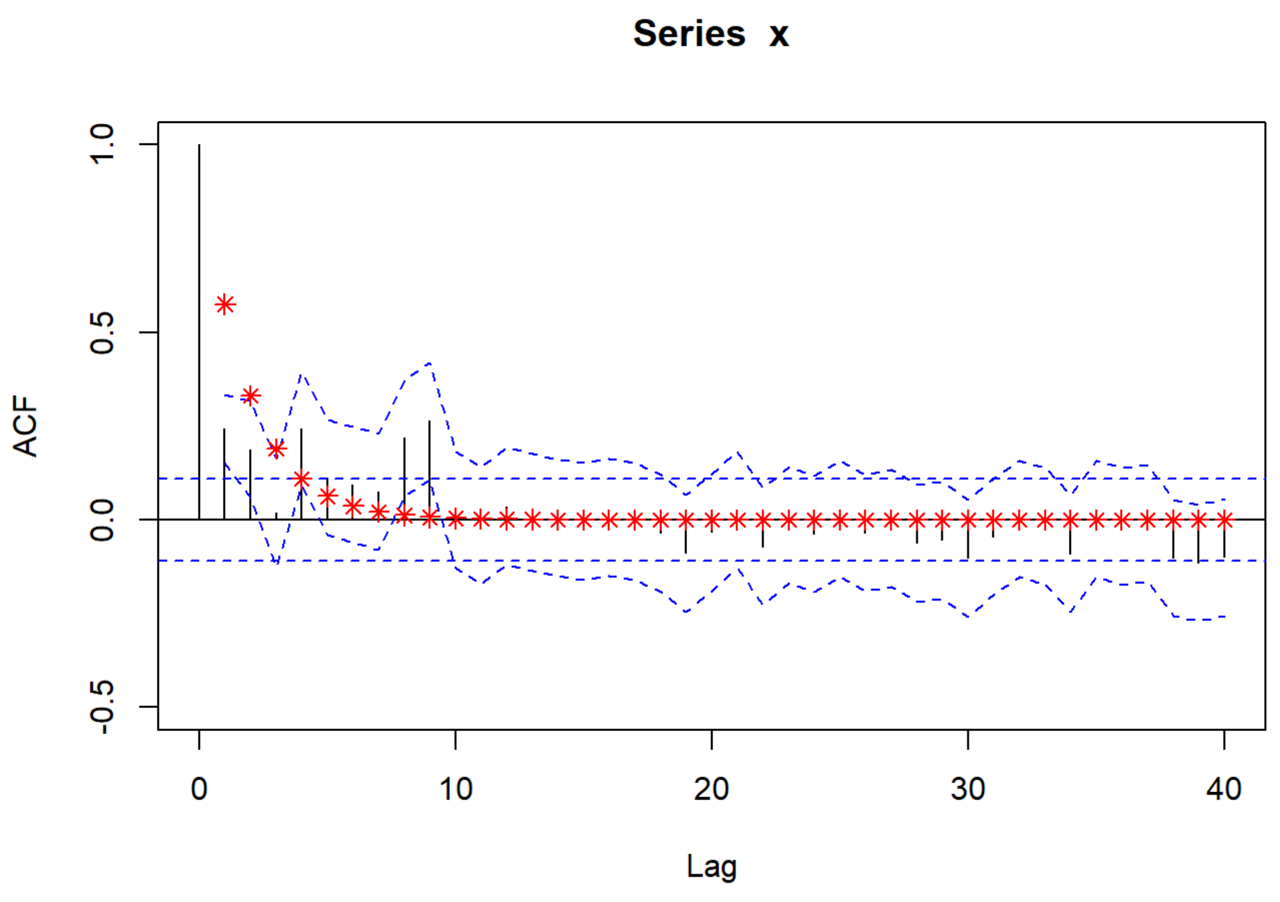



```
cat(ACF$acf[1] / ACF$acf[2], ACF$acf[2] / ACF$acf[3], ACF$acf[3] / ACF$acf[4])
```

```
## 4.111379 1.302208 10.1875
```

The sample autocorrelations suggest a possible autoregressive model. However, the first few sample autocorrelations suggest it might not be the first order AR.

```
source("sup_functions.R")
do_not_use_ <- acf_ci_ar(md$residuals, 0.575)
```



The above is a 95% confidence interval plot assuming that the data follows first order autoregression model, where red points are the theoretical values.
The confidence interval plot provides strong evidence that the first order autoregressive model is not optimal, since there are 4 values with confidence interval not containing the expected value.

```
minAICC(md$residuals, m = 20, method = "yw")
```

```
## $min.order
## [1] 10
##
## $min.aicc
## [1] 4565.496
##
## $aicc
##  [1] 4616.124 4598.959 4595.125 4596.146 4578.693 4580.624 4582.696 4583.190
##  [9] 4577.577 4570.102 4565.496 4566.440 4568.265 4566.046 4568.250 4570.510
## [17] 4571.887 4572.823 4573.999 4576.161 4578.419
```

The suggested autoregressive model (using Yule-Walker Algorithm) is the 10th order AR.

Estimation

```
armd <- arima(md$residuals, order = c(10, 0, 0))
test_yt <- predict(armd, n.ahead = 50)$pred
test_mt <- predict(md, newdata = data.frame(tt = 316:365))
test_st <- rep_len(st[(which(st == st[315])[1] + 1):(which(st == st[315])[1] + 7)], length.out = 50)

pred_x <- test_yt + test_mt + test_st
mse <- mean((test_df$rep_count - pred_x) ^ 2)
mpae <- 100 * mean(abs((test_df$rep_count - pred_x) / (test_df$rep_count)))
cat("Mean Squared Error on Test Set: ", mse); cat("Mean Percentage Absolute Error on Test Set: ", mpae)
```

```
## Mean Squared Error on Test Set:  490719.8
```

```
## Mean Percentage Absolute Error on Test Set:  51.52899
```