# 案例：门诊患者数量的时间序列分析

郑霏阳，吴翔

2019-10-17

```
# load pacakges
suppressMessages(library(magrittr))
suppressMessages(library(smooth))
set.seed(1234)
```

## 概述

我们通过案例来阐述如何使用时间序列分析方法研究**门诊患者数量**。所有分析过程均通过R语言实现。

本案例源自医疗卫生服务领域中的常用场景：

- 门诊患者数量可以视作时间序列，那么应该如何预测门诊患者数量，以便更好地调度医疗服务资源？

当我们分析**门诊患者数量**这一议题时，认为它可以分解为：

- 趋势因素
- 季节因素
- 随机因素

因此，本案例包括以下两个部分：

- 创设数据，从而构造一个门诊患者数量的时间序列数据。这一过程展现了计量经济学中**数据生成过程（data generating process, DGP）**这一重要概念。
- 分析时间序列数据。这一过程回顾了前几次课所讲授的主要分析工具。

## 创设门诊患者数量的时间序列数据

:

可以认为，门诊患者数量$Y_t$这一时间序列数据，由以下部分构成：

- 趋势因素$T_t$：使用Logistic模型刻画趋势因素
- 季节因素$S_t$：使用不同月份门诊患者数量不同这一特征，来刻画季节因素
- 随机因素$I_t$：使用正态分布刻画随机因素

周期设置为医院过去十年的数据，即$10 \times 12 = 120$个月的数据。

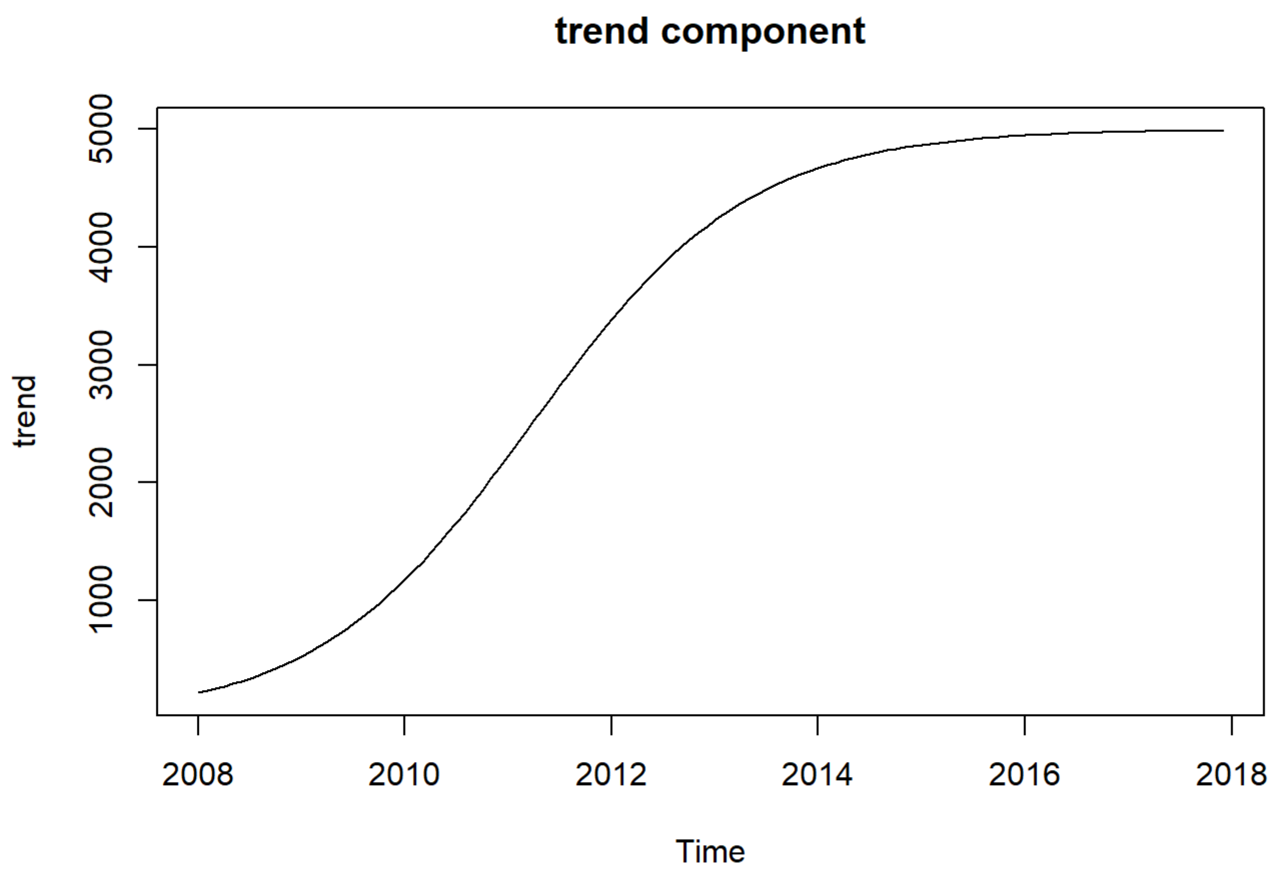
换言之，门诊患者数量表示为

$$Y_t = T_t + S_t + I_t, 1 \leq t \leq N.$$

## 构造趋势因素

使用Logistic增长模型构造趋势因素，

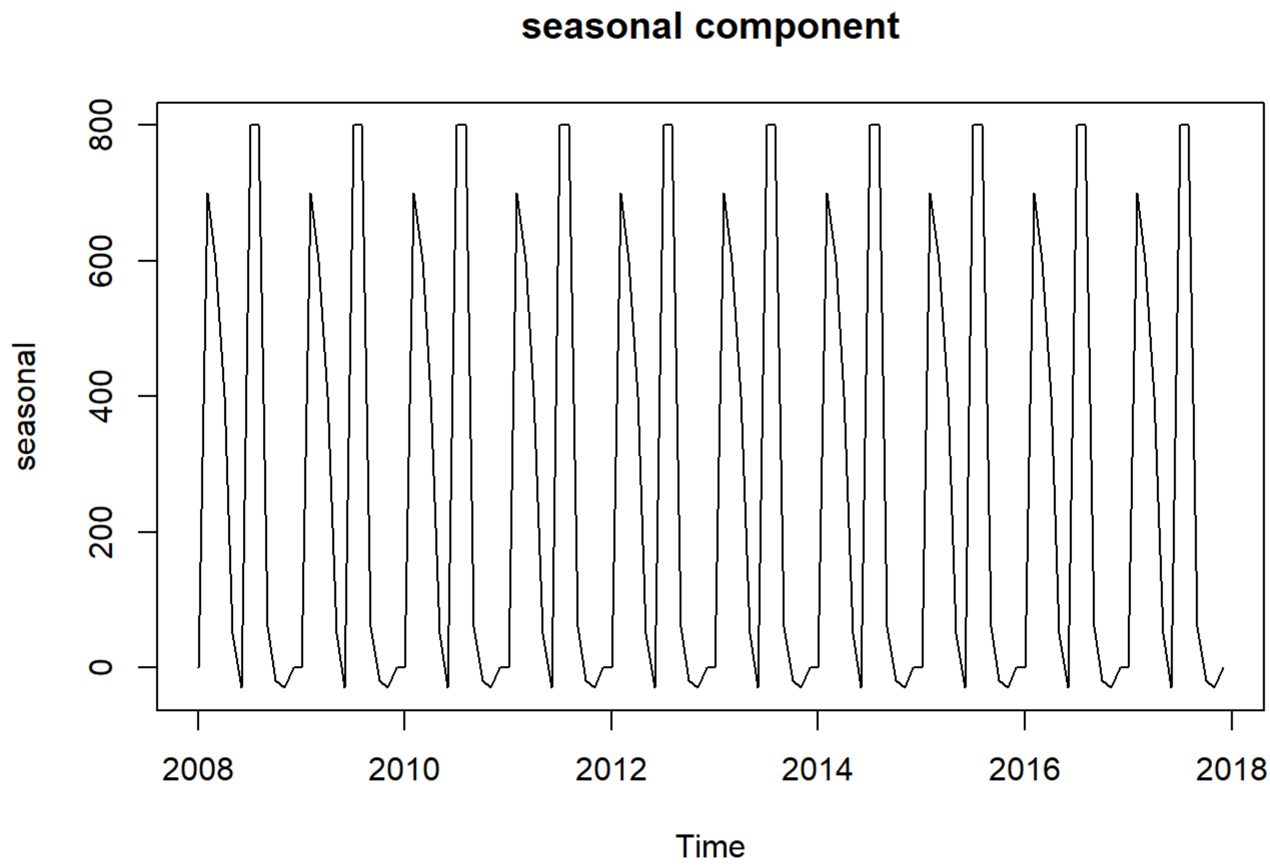
```
# create a time series
trend.start <- 200
r <- 0.08
N <- 5000
t <- 1:120
trend <- N / (1 + (N / trend.start - 1) * exp(- r * t))
trend <- trend %>% round() %>% ts(frequency = 12, start = c(2008, 1))
plot.ts(trend, main = "trend component")
```




# 构造季节因素

## 进而，构造季节因素。

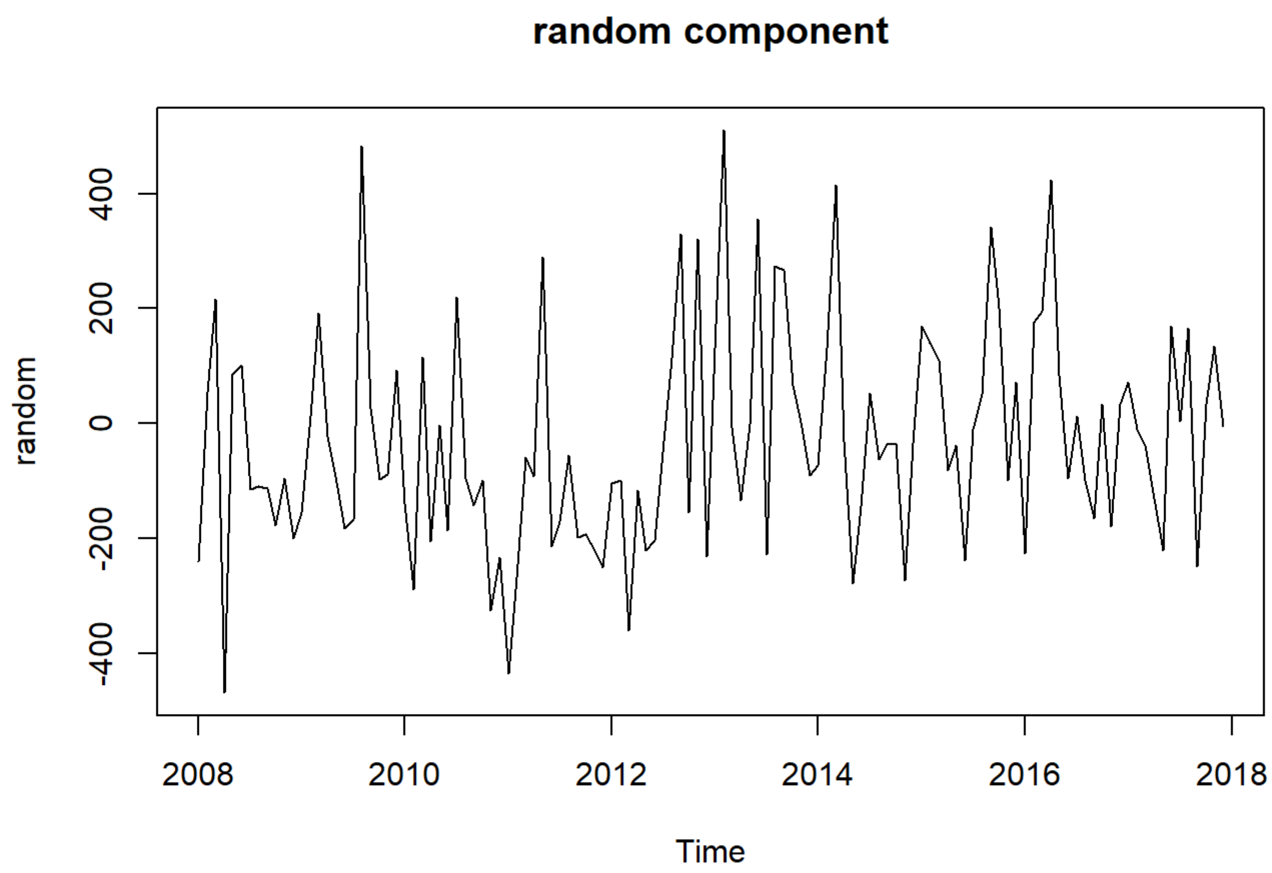
```
# create a time series
month.adjust <- c(0, 700, 600, 400, 50, -30, 800, 800,
60, -20, -30, 0)
seasonal <- rep(month.adjust, 10) %>% round() %>% ts(fr
equency = 12, start = c(2008, 1))
plot.ts(seasonal, main = "seasonal component")
```




# 构造随机因素

## 最后，构造随机因素。

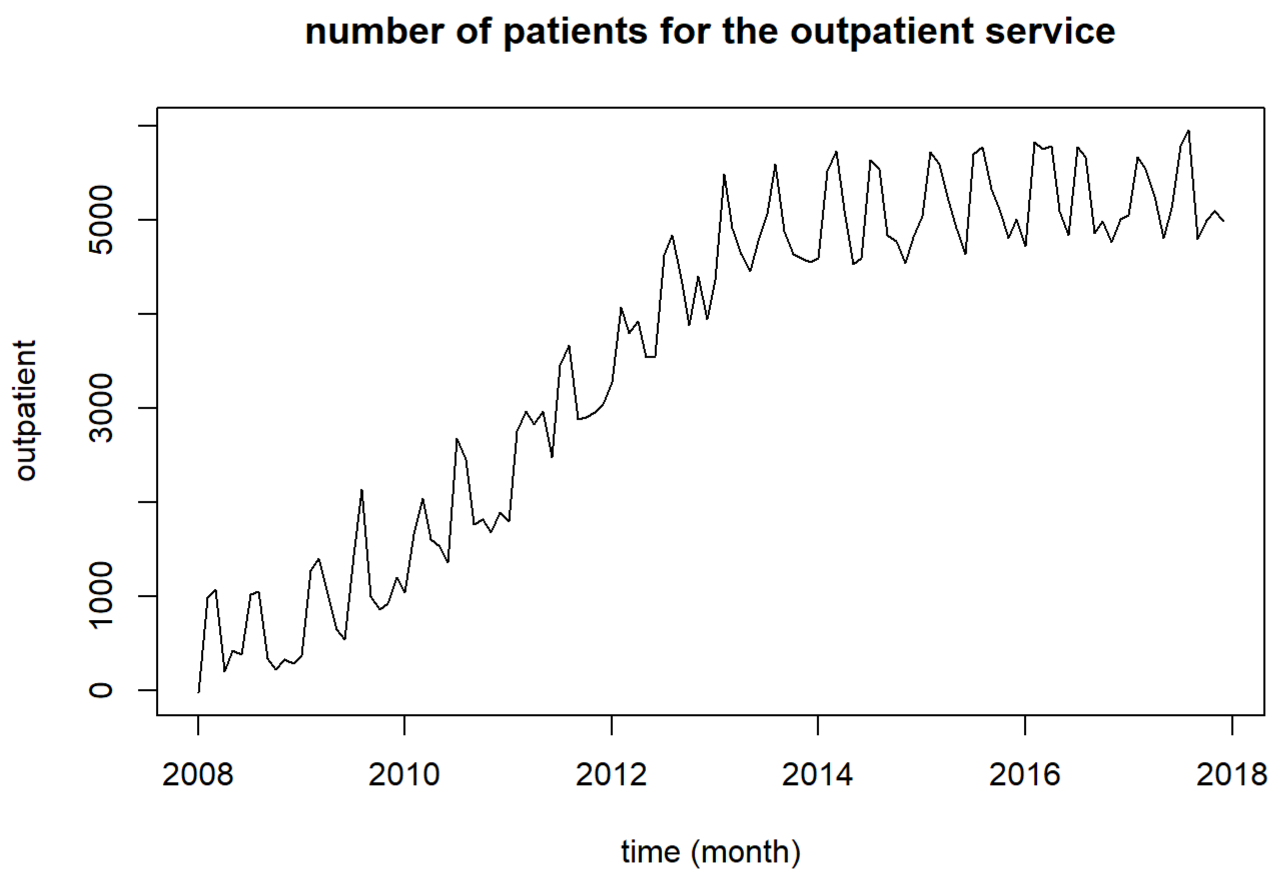
```
# create a time series
random <- rnorm(n = 120, mean = 0, sd = 200) %>% round
() %>% ts(frequency = 12, start = c(2008, 1))
plot.ts(random, main = "random component")
```




## 构造时间序列数据

此时，使用加法模型将三个因素组合起来，得到最终的时间序列数据。亦即我们所创设的医院在过去十年间的每月门诊患者数量。

```
# create the number of patients for the outpatient service
outpatient <- trend + seasonal + random
plot.ts(outpatient, main = "number of patients for the outpatient service", xlab = "time (month)")
```




# 分析：非参数方法

在构造了医院门诊患者数量的时间序列数据$Y_t$之后，我们可以使用不同的分析方法来分析这一时间序列。

需要注意的是，此时我们知晓**数据生成过程**，因而有能力事先判断某一具体的分析方法是否符合实际问题的假设。

在此之前，我们先撰写函数来计算预测误差MSE和MAD。

```r
# write a function to assess accuracy
accuracy.ts <- function(y, yhat){
    # calculate MSE and MAD
    mse <- sum((y - yhat)^2) / length(y)
    mad <- sum(abs(y - yhat)) / length(y)
    # return MSE and MAD
    res <- data.frame(mse = mse, mad = mad)
    return(res)
}
```

## 移动平均法

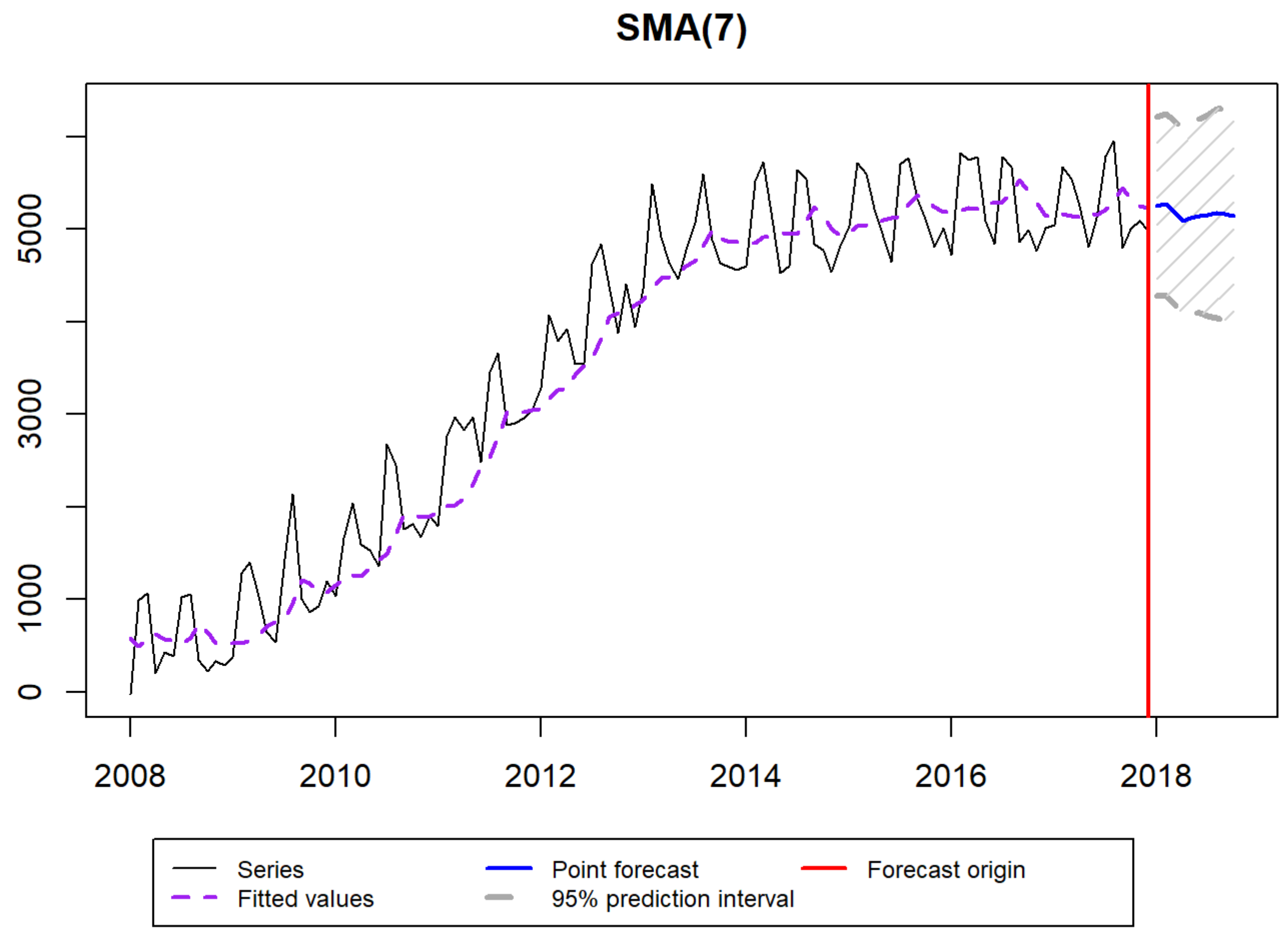
### 首先，采用移动平均法来分析时间序列$Y_t$。

```r
# moving average
ma.ts <- sma(outpatient, h = 30)
summary(ma.ts)
```

```
## Time elapsed: 0.19 seconds
## Model estimated: SMA(7)
## Initial values were produced using backcasting.
##
## Loss function type: MSE; Loss function value: 23908
8.6134
## Error standard deviation: 489
## Sample size: 120
## Number of estimated parameters: 2
## Number of degrees of freedom: 118
## Information criteria:
##  AIC AICc  BIC BICc
## 1831 1831 1836 1837
```

可以看到，**调节参数（tuning parameter）**取值$m=7$时，移动平均法达到最佳预测效果。

```
# plot the result
ma.ts %>% forecast() %>% plot()
```




评估预测误差如下.

```
# accuracy
accuracy.ts(outpatient, fitted(ma.ts))
```

```
##      mse mad
## 1 239089 401
```

## 指数平滑法

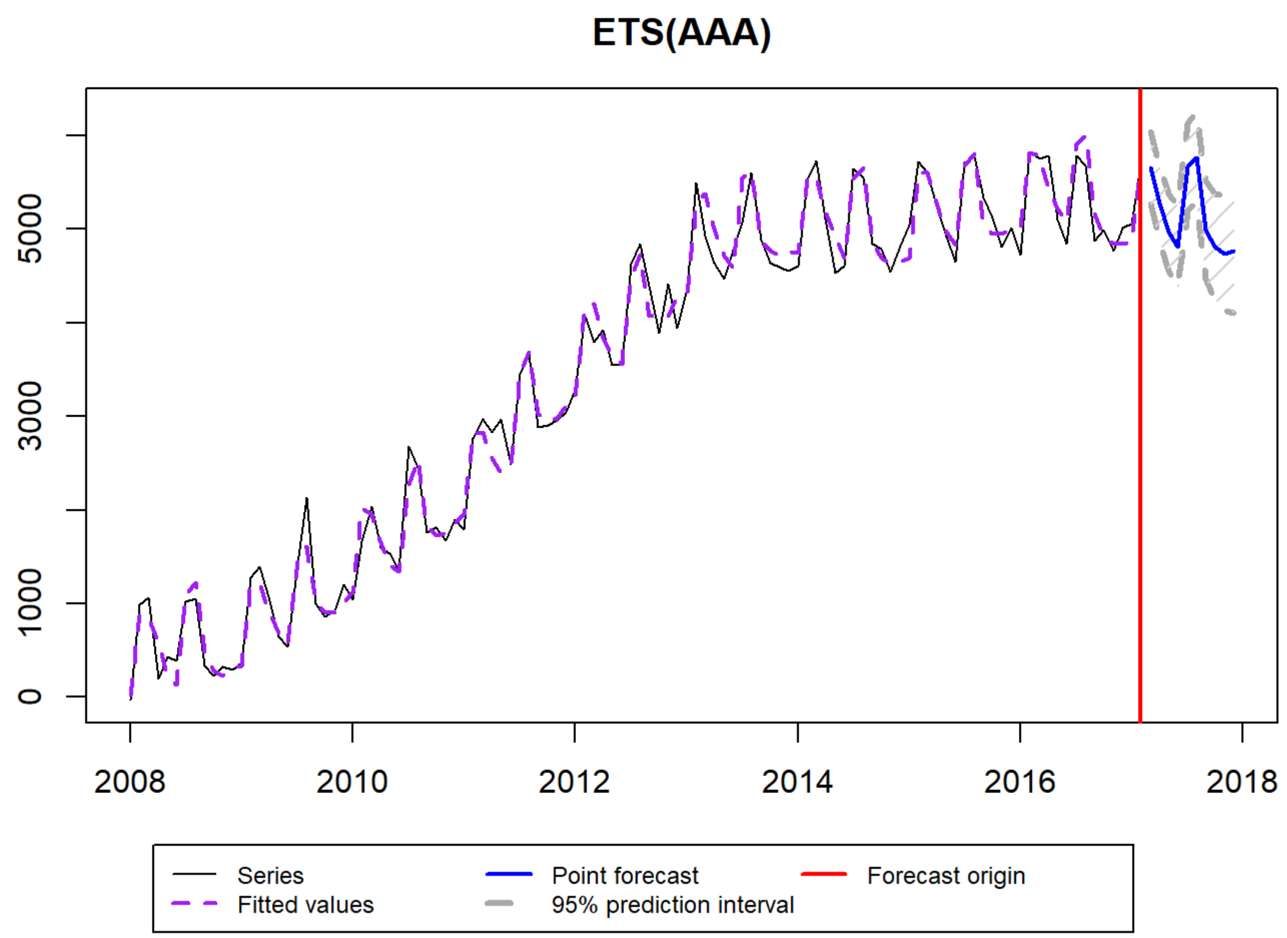
此外，还可以采用指数平滑法来分析时间序列$Y_t$。

```
# exponential smoothing
es.ts <- es(outpatient, holdout = T)
summary(es.ts)
```

```
## Time elapsed: 0.51 seconds
## Model estimated: ETS(AAA)
## Persistence vector g:
## alpha  beta gamma
## 0.185 0.051 0.000
## Initial values were optimised.
##
## Loss function type: MSE; Loss function value: 37916.7618
## Error standard deviation: 195
## Sample size: 110
## Number of estimated parameters: 17
## Number of provided parameters: 1
## Number of degrees of freedom: 93
## Information criteria:
##  AIC AICc  BIC BICc
## 1506 1513 1552 1567
##
## Forecast errors:
## MPE: 1.8%; sCE: -27.3%; Bias: 46.3%; MAPE: 3.6%
## MASE: 0.485; sMAE: 5.5%; sMSE: 0.4%; rMAE: 0.365; rRMSE: 0.364
```

可以看到，**调节参数（tuning parameter）**取值$\alpha = 0.043$时，指数平滑法达到最佳预测效果。

```
# plot the result
es.ts %>% forecast() %>% plot()
```




评估预测误差如下.

```
# accuracy
accuracy.ts(outpatient, fitted(es.ts))
```

```
##      mse mad
## 1 34755 134
```

依然可以看到，指数平滑法预测效果优于移动平均法。

# 门诊患者数量分析：参数方法

我们可以使用参数方法来理解时间序列$Y_t$。这通常包括以下几个部分：

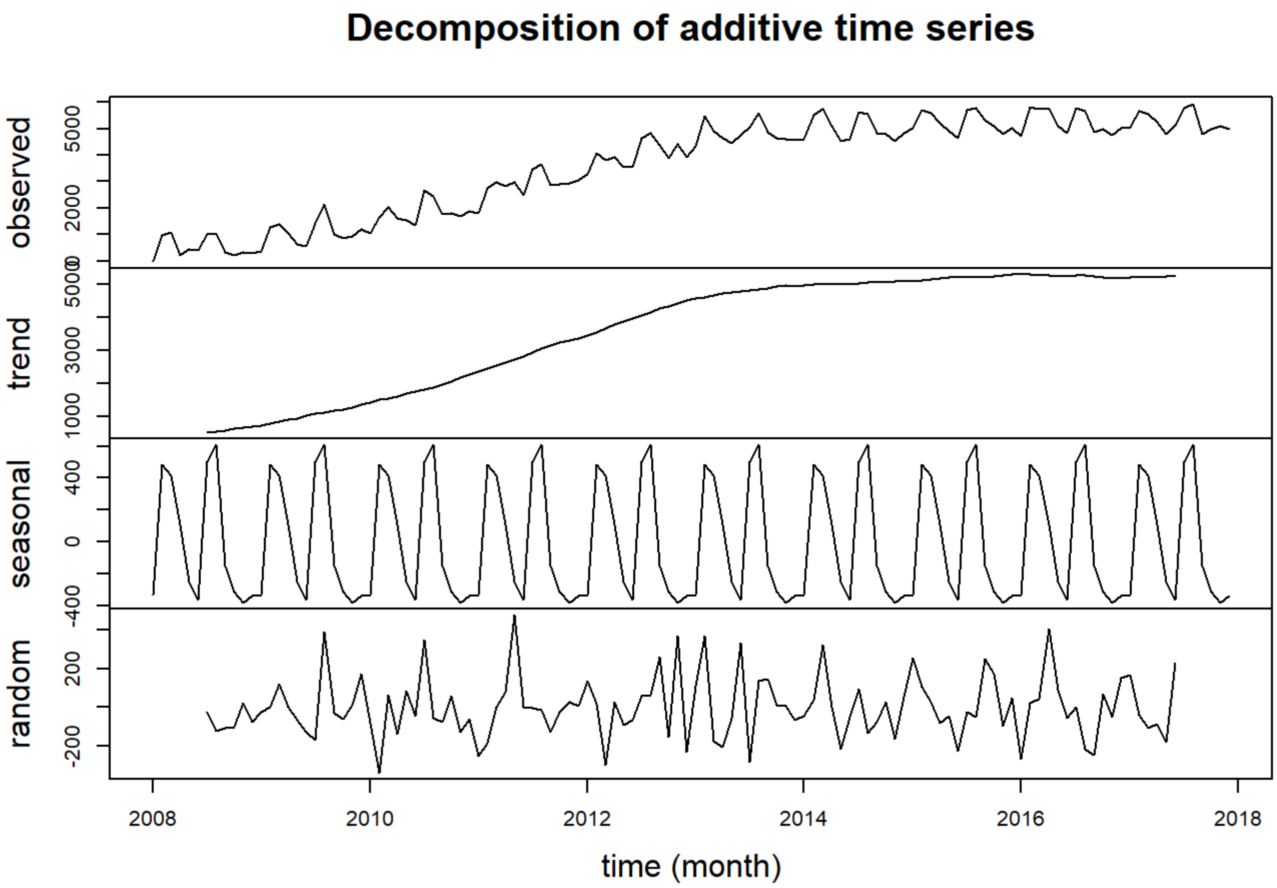
- 分解时间序列的各个要素

- 逐个分析其中每个要素

## 时间序列要素分解

### 首先，使用加法模型将医院十年来的月度门诊患者数量数据，分解成三个部分。

```
# decomposition
outpatient.comp <- decompose(outpatient, type = "additive")
# plot the decomposition
plot(outpatient.comp, xlab = "time (month)")
```

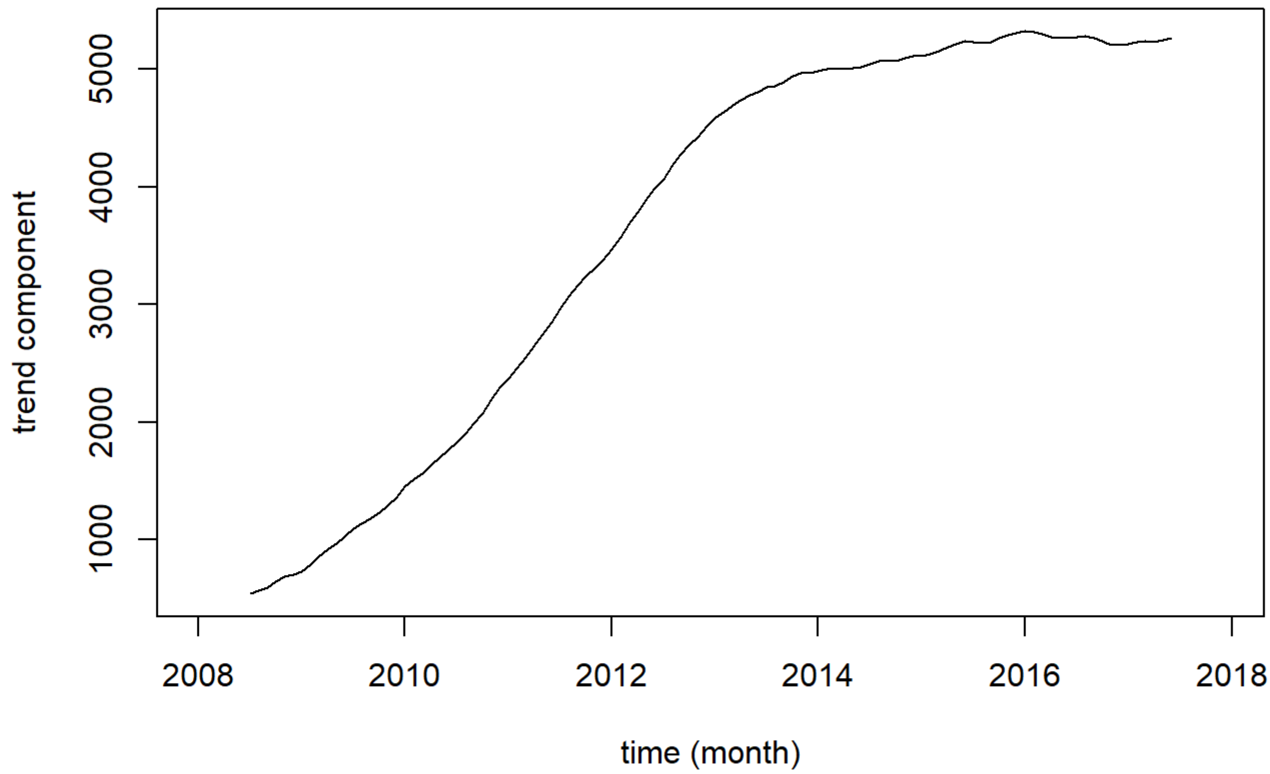



### 进而，可以分别分析三个部分。

## 趋势因素建模

### 先绘制趋势因素，观察其形状。

```
# plot the trend
plot(outpatient.comp$trend, xlab = "time (month)", ylab
= "trend component")
```




可以看到，趋势因素呈现S型，因而我们进一步分析这一情境，决定采用Logistic增长模型来刻画趋势因素。

由于Logistic增长模型的数学表达式为

$$Y_t = \frac{L}{1 + c \cdot e^{-rt}},$$

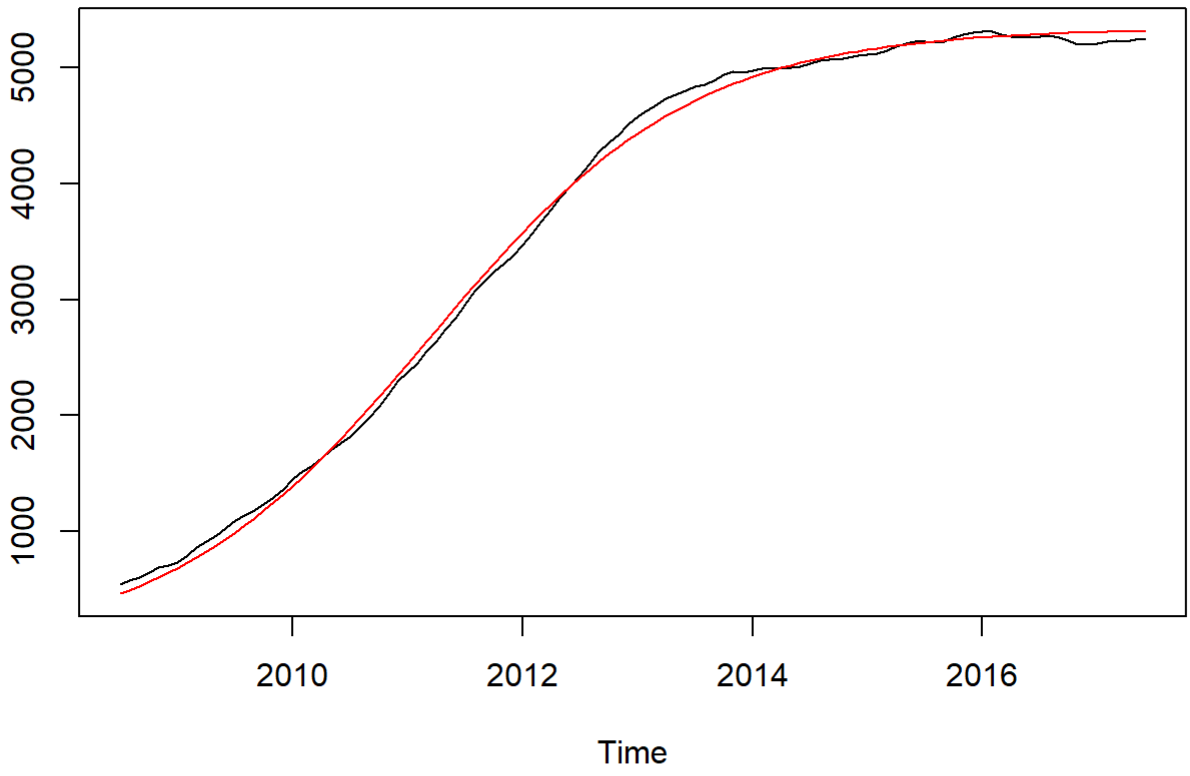
可以采用非线性最小二乘法估计模型。

```
# create a dataframe
y <- na.omit(as.numeric(outpatient.comp$trend))
t <- which(!is.na(outpatient.comp$trend))
dat <- data.frame(y = y, t = t)
# specific the model
suppressMessages(library(nls2))
starts <- list(l = 10000, c = 20, r = 0.1)
fit <- nls(y ~ l / (1 + c * exp(-r * t)), data = dat, s
tart = starts)
summary(fit)
```

```
##
## Formula: y ~ l/(1 + c * exp(-r * t))
##
## Parameters:
##   Estimate Std. Error t value Pr(>|t|)
## l 5.34e+03   1.47e+01   363.0   <2e-16 ***
## c 1.76e+01   5.24e-01    33.6   <2e-16 ***
## r 7.30e-02   8.12e-04    89.9   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
0.1 ' ' 1
##
## Residual standard error: 74.9 on 105 degrees of free
dom
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 1.98e-06
```

比较两个曲线：趋势因素，以及Logistic模型刻画的趋势因素。

```
y <- y  %>% ts(frequency = 12, start = c(2008, 7))
y.nls <- fitted(fit) %>% ts(frequency = 12, start = c(2008, 7))
ts.plot(y, y.nls, gpars = list(col = c("black", "red")))
```




类似地，可以评估预测误差。

```
# accuracy
accuracy.ts(y, predict(fit))
```

```
##     mse  mad
## 1 5460 63.7
```

可以看到，误差非常小。

# 总结

- 可以依据情形选择参数方法或者非参数方法，来分析时间序列数据
- 参数方法通常能够增进我们对问题的理解，同时需要针对情境做具体的统计建模