Regression analysis is one of the most demanding machine learning methods in 2019. One group of regression analysis for measuring effects and to evaluate a policy program is Interrupted Time Series. This method is well suited for benchmarking and finding improvements for optimization in organizations. It can, therefore, be used to design organizations so they generate more value for employees and customers. In this article, you learn how to do Interrupted Time Series in R.

**Program Evaluation Definition**

Program evaluation is the selection of outcomes of a program or policy, evaluated by a set of standards as a means to investigate if the program or policy contributes to improvement in a desired goal or result. When beginning an evaluation, program people therefore often want the answer questions like:

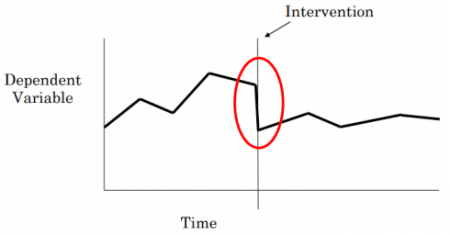
* Does the program work? And how can it be improved?
* Is the program worthwhile?

The main goal of program evaluation is, therefore, to investigate if the policy change contributes to the improvement of the program. Program evaluation can help improve program services, in particular:

* Planning the evaluation
* Determining the evaluation questions (Determining the goal of the evaluation: What is the evaluation question, what is the evaluation to find out?)
* Answering evaluation questions with evaluation methods (What methods will be used?)
* Making the results useful by determining how will the results be reported so that they can be used by the organization to make improvements

**Interrupted Time Series Methodology**

Interrupted Time Series (ITS) is a special kind of time series in which we know the specific point in the series at which an intervention occurred. The causal hypothesis in ITS is that observations after treatment will have a different level or slope from those before intervention – the interruption. ITS is a Strong quasi-experimental alternative to randomized design if this is not feasible.

The purpose of Interrupted Time Series is therefore to evaluate before- and after effect of a policy program as plottet in the below graph:  
[](https://i1.wp.com/datascienceplus.com/wp-content/uploads/2019/07/p5.png?ssl=1)

**Interrupted Time Series: Statistical model**

ITS analyses use regression-based techniques in it model but it also implements dichotomous variables for the ITS effect.  
Let is consider the standard linear regression:  
$$ y = α + βx + ε $$

Where: α = intercept, β = coefficient, x = independent variable and ε = residual (error).

If we add the following ITS based on segmented linear regression we get the following model:

$$ y\_t = β\_0 + β\_1 \* time\_t + β\_2 \* intervention\_t + β\_3 \* timepost\_t + ε\_t $$

Where: Y is the outcome , time indicates the number of quarters from the start of the series , intervention is a dichotomous variable taking the values 0 in the pre-intervention segment and 1 in the post-intervention segment, timepost is 0 in the pre-intervention segment and counts the quarters in the post-intervention segment at time. Furthermore: β0 estimates the base level of the outcome at the beginning of the series, β1 estimates the base trend, i.e. the change in outcome per quarter in the pre-intervention segment, β2 estimates the change in level in the post-intervention segment, β3 estimates the change in trend in the post-intervention segment and ε estimates the error.

**Interrupted Time Series in R**

Let us load the packages into the R library:

# Load the packages into the R library

library(Wats)

library(ggplot2)

library(nlme)

Let us load the [dataset](https://github.com/OuhscBbmc/Wats/blob/master/data/CountyMonthBirthRate2014Version.rda) into R:

## Monthly Growth Fertility Rates (GFR) for 12 urban Oklahoma counties

data(CountyMonthBirthRate2014Version)

## Extract canadian county as an example

canadian <- subset(CountyMonthBirthRate2014Version, CountyName == "canadian")

## Date of Oklahoma bombing

dateBombing <- as.Date("1995-04-19")

## Indicator for post-bombing dates

canadian$postBombing = dateBombing)

## Date as numeric

canadian$DateNum <- as.numeric(canadian$Date)

## Center the date at bombing

canadian$DateNumCtr <- canadian$DateNum - as.numeric(dateBombing)

Let us now try plotting the data:

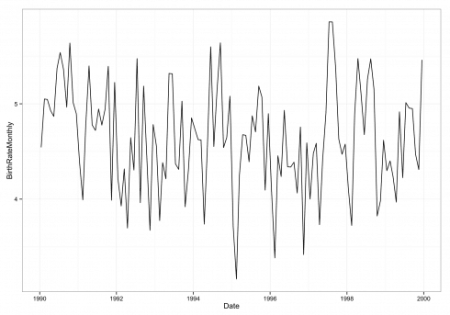
# Plotting data

ggplot(data = canadian, mapping = aes(x = Date, y = BirthRateMonthly)) +

layer(geom = "line") +

theme\_bw() + theme(legend.key = element\_blank())

The above coding gives us the following graph:

[](https://i2.wp.com/datascienceplus.com/wp-content/uploads/2019/07/p1-1.png?ssl=1)

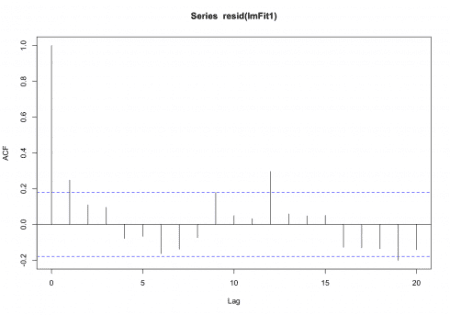
Let us now try to do a linear model fit for a correlogram:

# Linear model fit

lmFit1 <- lm(BirthRateMonthly ~ DateNum, data = canadian)

## Auto- and Cross- Covariance and -Correlation Function Estimation

acf(resid(lmFit1))

The above coding gives us the following graph:  
[](https://i1.wp.com/datascienceplus.com/wp-content/uploads/2019/07/p2-1.png?ssl=1)

The segmented interrupted time series model we use into model dat dataset is the following:

$$ E[Y]=β0+β1Time+β2Period2+β3Time×Period2 $$

Where: β0 is the pre-period intercept, β1 is the pre-period slope (baseline time trend), β2 is the immediate effect of the event, β3 is the slope change after the event, β0+β2 is the post-period intercept, β1+β3 is the post-period slop. Time is the date centered at the event (necessary for interpretability of immediate effect) and Period2 is the indicator for the time period after the event.

Let us fit this model by the dataset in R:

## Fit interrupted time series model

glsFit1 <- gls(model = BirthRateMonthly ~ DateNumCtr + postBombing + DateNumCtr:postBombing,

data = canadian,

correlation = corAR1(0.25))

# Summary of interrupted time series model

summary(glsFit1)

*Generalized least squares fit by REML*

*Model: BirthRateMonthly ~ DateNumCtr + postBombing + DateNumCtr:postBombing*

*Data: canadian*

*AIC BIC logLik*

*234.8338 251.3553 -111.4169*

*Correlation Structure: AR(1)*

*Formula: ~1*

*Parameter estimate(s):*

*Phi*

*0.260391*

*Coefficients:*

*Value Std.Error t-value p-value*

*(Intercept) 4.399954 0.17586388 25.019089 0.0000*

*DateNumCtr -0.000278 0.00015751 -1.765280 0.0801*

*postBombingTRUE 0.059139 0.25593277 0.231074 0.8177*

*DateNumCtr:postBombingTRUE 0.000439 0.00025098 1.747655 0.0832*

*Correlation:*

*(Intr) DtNmCt pBTRUE*

*DateNumCtr 0.860*

*postBombingTRUE -0.665 -0.572*

*DateNumCtr:postBombingTRUE -0.559 -0.644 -0.122*

*Standardized residuals:*

*Min Q1 Med Q3 Max*

*-2.27069072 -0.70619943 0.03102496 0.60955441 2.29540604*

*Residual standard error: 0.5530618*

*Degrees of freedom: 120 total; 116 residual*

Let us now try to make a overplot prediction in the before- and after period of the dataset in R:

## Create prediction dataset

newdata <- data.frame(DateNumCtr = seq(min(canadian$DateNumCtr), max(canadian$DateNumCtr), by = 1))

newdata$postBombing = 0)

## Predict before- and after period of the dataset

newdata$BirthRateMonthly <- predict(glsFit1, newdata = newdata)

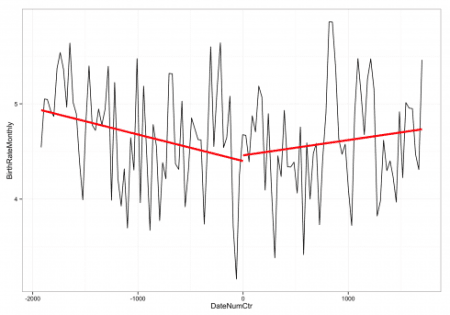
ggplot(data = canadian, mapping = aes(x = DateNumCtr, y = BirthRateMonthly)) +

layer(geom = "line") +

layer(data = subset(newdata, DateNumCtr = 0), geom = "line", color = "red", size = 1.5) +

theme\_bw() + theme(legend.key = element\_blank())

The above coding gives us the following graph:

[](https://i2.wp.com/datascienceplus.com/wp-content/uploads/2019/07/p3.png?ssl=1)

It is also possible to do explicit modeling of monthly fluctuations:

## Fit monthly fluctuations

glsFit2 <- gls(model = BirthRateMonthly ~ DateNumCtr + postBombing + DateNumCtr:postBombing + factor(Month),

data = canadian,

correlation = corAR1(0.25))

summary(glsFit2)

*Generalized least squares fit by REML*

*Model: BirthRateMonthly ~ DateNumCtr + postBombing + DateNumCtr:postBombing + factor(Month)*

*Data: canadian*

*AIC BIC logLik*

*227.1337 272.251 -96.56685*

*Correlation Structure: AR(1)*

*Formula: ~1*

*Parameter estimate(s):*

*Phi*

*0.2130256*

*Coefficients:*

*Value Std.Error t-value p-value*

*(Intercept) 4.056658 0.19580489 20.717860 0.0000*

*DateNumCtr -0.000271 0.00012568 -2.152381 0.0337*

*postBombingTRUE -0.010639 0.20603440 -0.051637 0.9589*

*factor(Month)2 -0.215932 0.18251667 -1.183083 0.2394*

*factor(Month)3 0.289306 0.20102227 1.439174 0.1531*

*factor(Month)4 0.141948 0.20482160 0.693033 0.4898*

*factor(Month)5 0.559483 0.20585345 2.717869 0.0077*

*factor(Month)6 0.524528 0.20591945 2.547250 0.0123*

*factor(Month)7 0.763552 0.20589578 3.708439 0.0003*

*factor(Month)8 0.637137 0.20586202 3.094971 0.0025*

*factor(Month)9 0.793359 0.20574894 3.855955 0.0002*

*factor(Month)10 0.415827 0.20511644 2.027271 0.0452*

*factor(Month)11 -0.062976 0.20186108 -0.311976 0.7557*

*factor(Month)12 0.645521 0.18516481 3.486198 0.0007*

*DateNumCtr:postBombingTRUE 0.000445 0.00019924 2.231004 0.0278*

*Correlation:*

*(Intr) DtNmCt pBTRUE fc(M)2 fc(M)3 fc(M)4 fc(M)5 fc(M)6 fc(M)7 fc(M)8 fc(M)9 f(M)10*

*DateNumCtr 0.634*

*postBombingTRUE -0.476 -0.578*

*factor(Month)2 -0.474 -0.013 0.010*

*factor(Month)3 -0.527 -0.023 0.020 0.550*

*factor(Month)4 -0.544 -0.033 0.031 0.465 0.595*

*factor(Month)5 -0.503 0.015 -0.056 0.446 0.508 0.601*

*factor(Month)6 -0.510 0.006 -0.044 0.443 0.491 0.517 0.606*

*factor(Month)7 -0.516 -0.003 -0.031 0.442 0.488 0.500 0.522 0.606*

*factor(Month)8 -0.522 -0.013 -0.018 0.442 0.487 0.497 0.504 0.522 0.606*

*factor(Month)9 -0.527 -0.022 -0.006 0.442 0.487 0.496 0.499 0.504 0.522 0.606*

*factor(Month)10 -0.530 -0.031 0.007 0.440 0.485 0.494 0.496 0.498 0.502 0.520 0.605*

*factor(Month)11 -0.527 -0.039 0.019 0.433 0.477 0.486 0.486 0.488 0.490 0.494 0.513 0.597*

*factor(Month)12 -0.486 -0.041 0.030 0.397 0.438 0.447 0.446 0.447 0.449 0.450 0.455 0.473 0.558*

*DateNumCtr:postBombingTRUE -0.404 -0.637 -0.125 0.009 0.010 0.010 0.020 0.017 0.014 0.011 0.008 0.005 0.001 -0.009*

*f(M)11 f(M)12*

*Standardized residuals:*

*Min Q1 Med Q3 Max*

*-2.16271810 -0.60854485 -0.01295044 0.61383062 2.38590679*

*Residual standard error: 0.4603772*

*Degrees of freedom: 120 total; 105 residual*

The above table shows that the immediate change postBombingTRUE is non-significant. The trend changes DateNumCtr:postBombingTRUE is positive and significant.

Let us also try plotting the explicit modeling of monthly fluctuations:

## Provide dataset for prediction of monthly fluctuations

newdata2 <- data.frame(Date = seq(min(canadian$Date), max(canadian$Date), by = 1))

newdata2$DateNum <- as.numeric(newdata2$Date)

newdata2$postBombing = dateBombing)

newdata2$DateNum <- as.numeric(newdata2$Date)

newdata2$DateNumCtr <- newdata2$DateNum - as.numeric(dateBombing)

newdata2$Month <- as.numeric(format(newdata2$Date, "%m"))

## Predict monthly fluctuations

newdata2$BirthRateMonthly <- predict(glsFit2, newdata = newdata2)

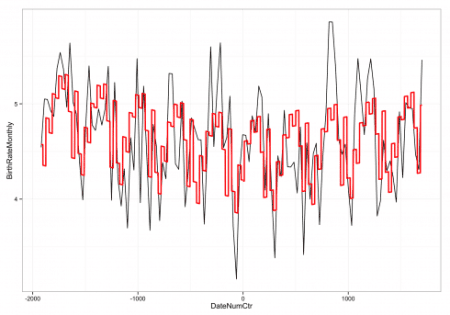
## Plot monthly fluctuations

ggplot(data = canadian, mapping = aes(x = DateNumCtr, y = BirthRateMonthly)) +

layer(geom = "line") +

layer(data = subset(newdata2, DateNumCtr = 0), geom = "line", color = "red", size = 1) +

theme\_bw() + theme(legend.key = element\_blank())

The above coding gives us the following graph:  
[](https://i1.wp.com/datascienceplus.com/wp-content/uploads/2019/07/p4.png?ssl=1)