Examples of Variable Transformation in Data Exploration

Fanny Chow

Overview

Here are some examples of variable transformation in the process of data analysis. In general, variable transformation helps get us one step closer to fitting the appropriate model to the data.

Time-Series Analysis

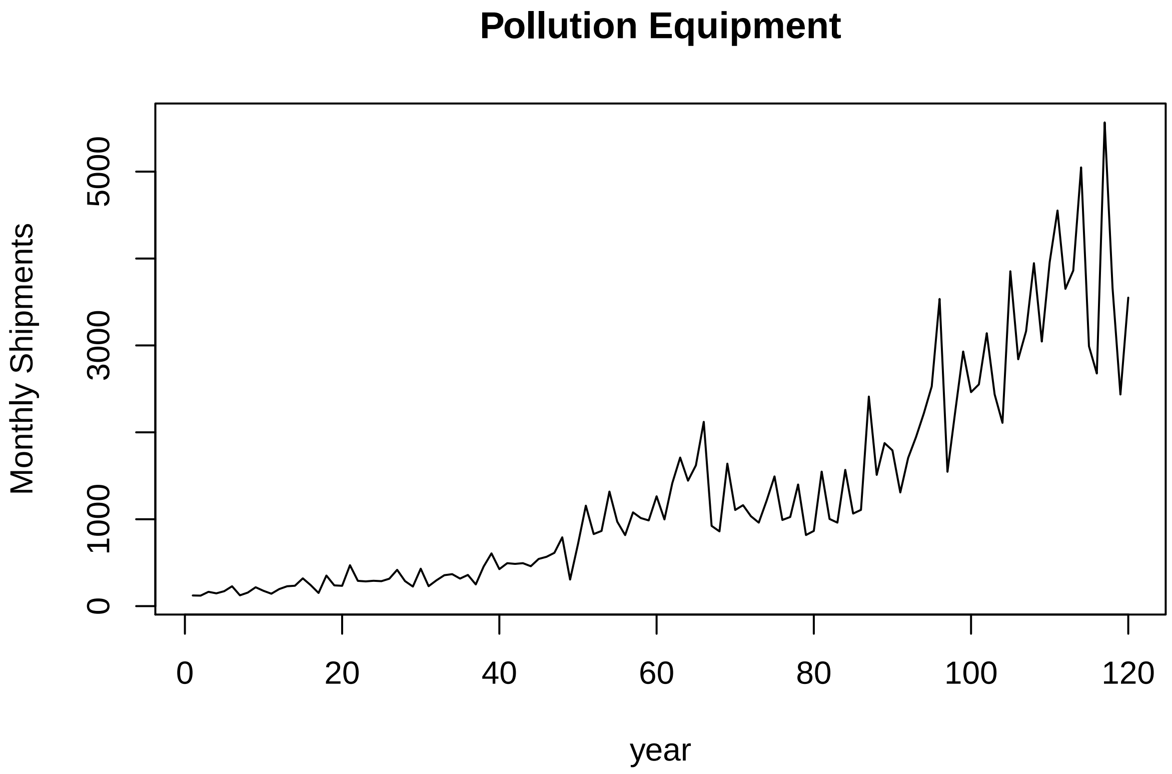
In time-series analysis, we are examining trends over time. Time-series techniques are frequently used by social scientists interested in predicting future events. For example: the future of the economy, future voting trends, or future stock markets. Before fitting a time-series model, we must make sure that there is constant variance over time.

This data set describes monthly shipments of pollution equipment (in thousands of French francs) from Jan 1986 – Oct 1996. Perhaps an environmental economist or climate scientist is interested in trends in the pollution industry.

```
# read in data
library(fma)
data("pollution")
require(fma)
data(pollution)
x = window(pollution, start=c(1986,1), end=c(1995,12))
t = 1:length(x)
x = as.vector(x)
```

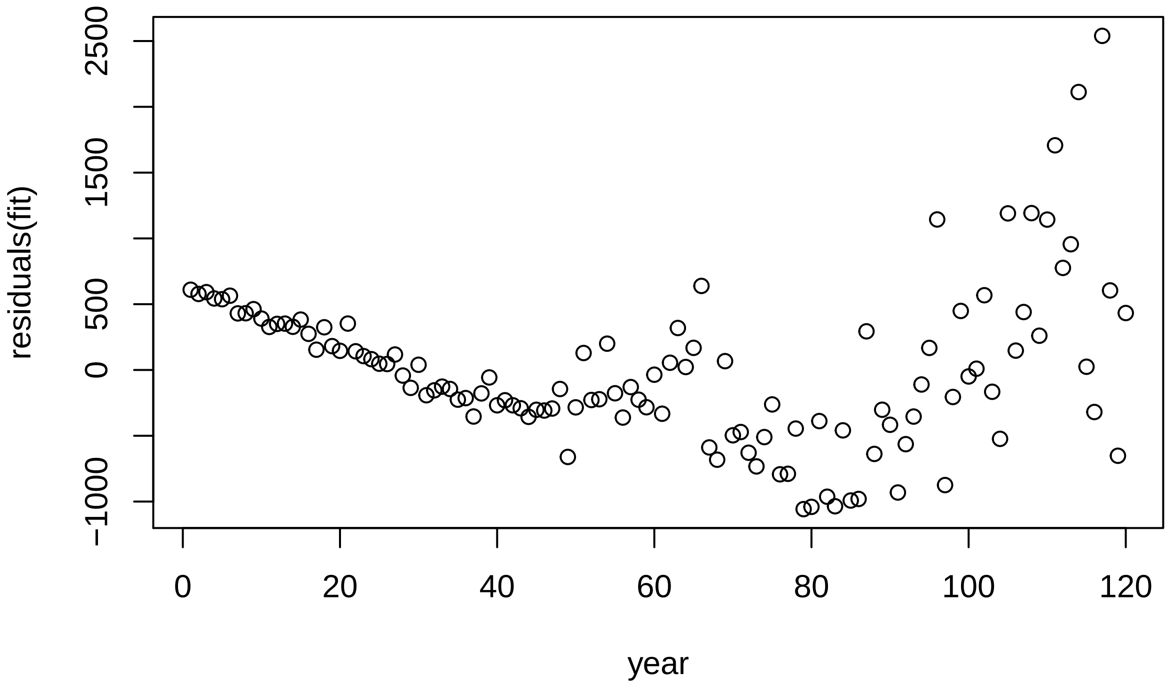
Let's plot the data and see what happens through the years.

```
plot(t,x, type="l", main="Pollution Equipment", ylab="Monthly Shipments", xlab="year")
```



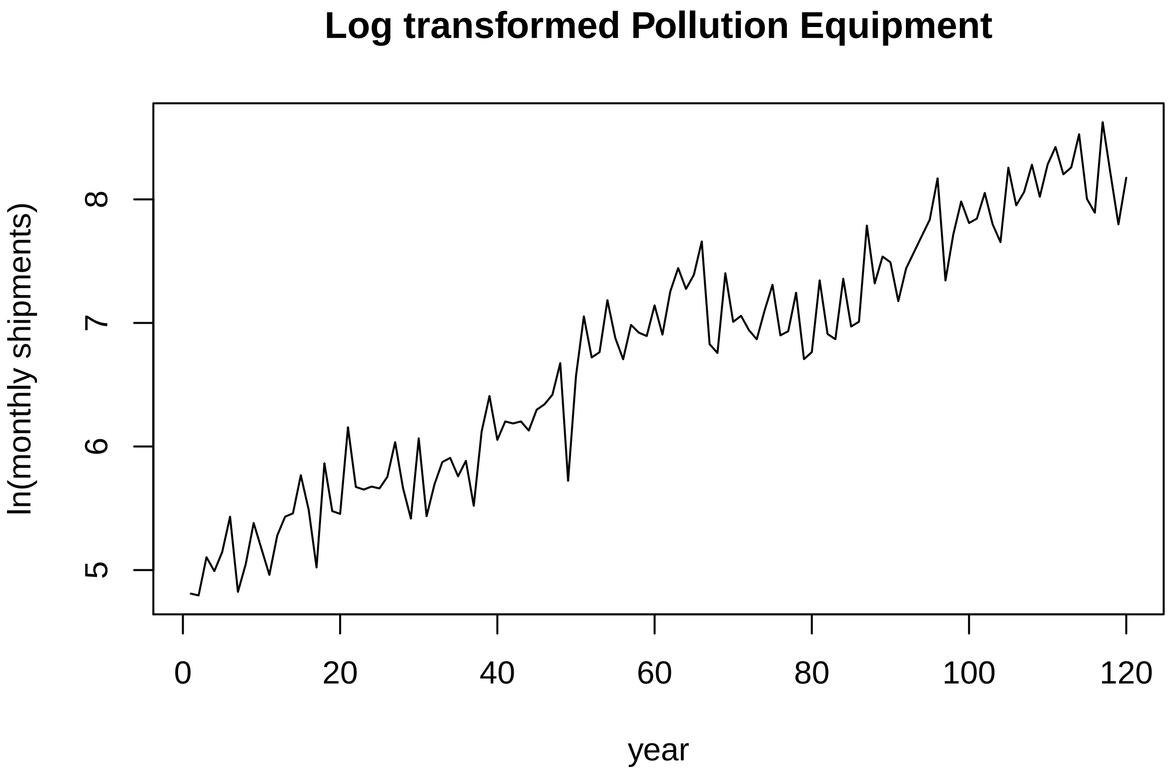
We are interested in fitting the data, let's try fitting it to a line using linear regression. After the fitting the data, let's plot the residuals (fitted - observed value) below. Notice that variance is not constant, which is a prerequisite to later steps in fitting a good time-series model.

```r
fit=lm(x~t)
plot(residuals(fit), xlab="year")
```



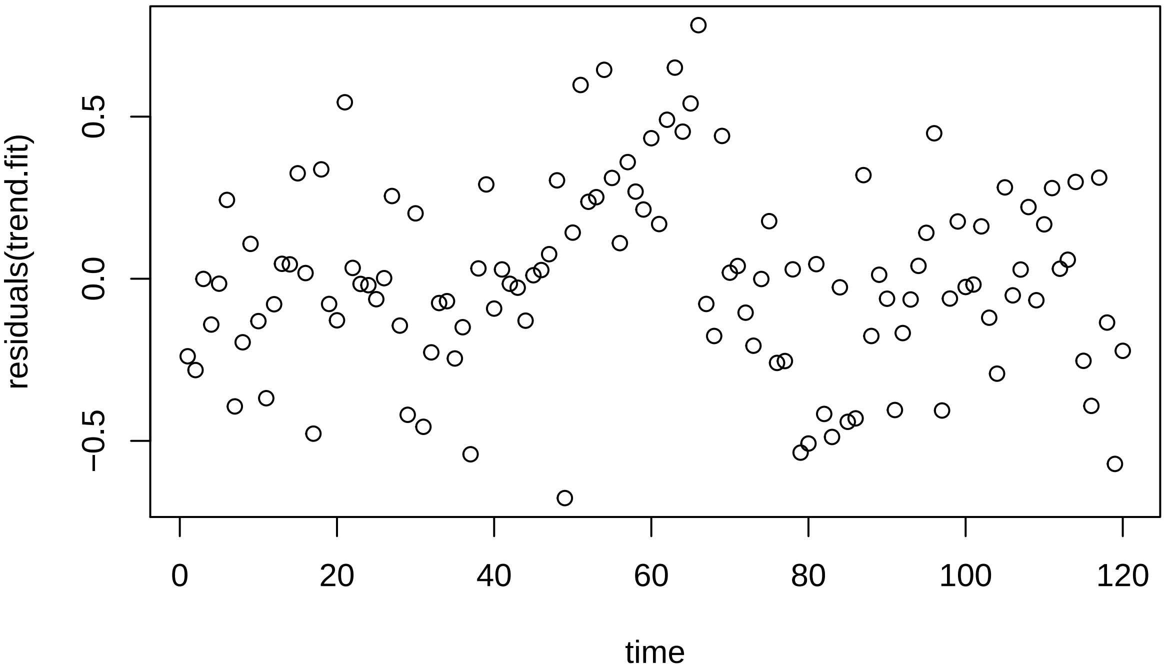
There is an increase in variance over time, so a log transformation is recommended. Let's transform the data with a ln transformation to stabilize the variance.

```
transx = log(x)
plot(t, transx, type="l", ylab="ln(monthly shipments)", xlab="year", main="Log transformed Pollution Equ
```



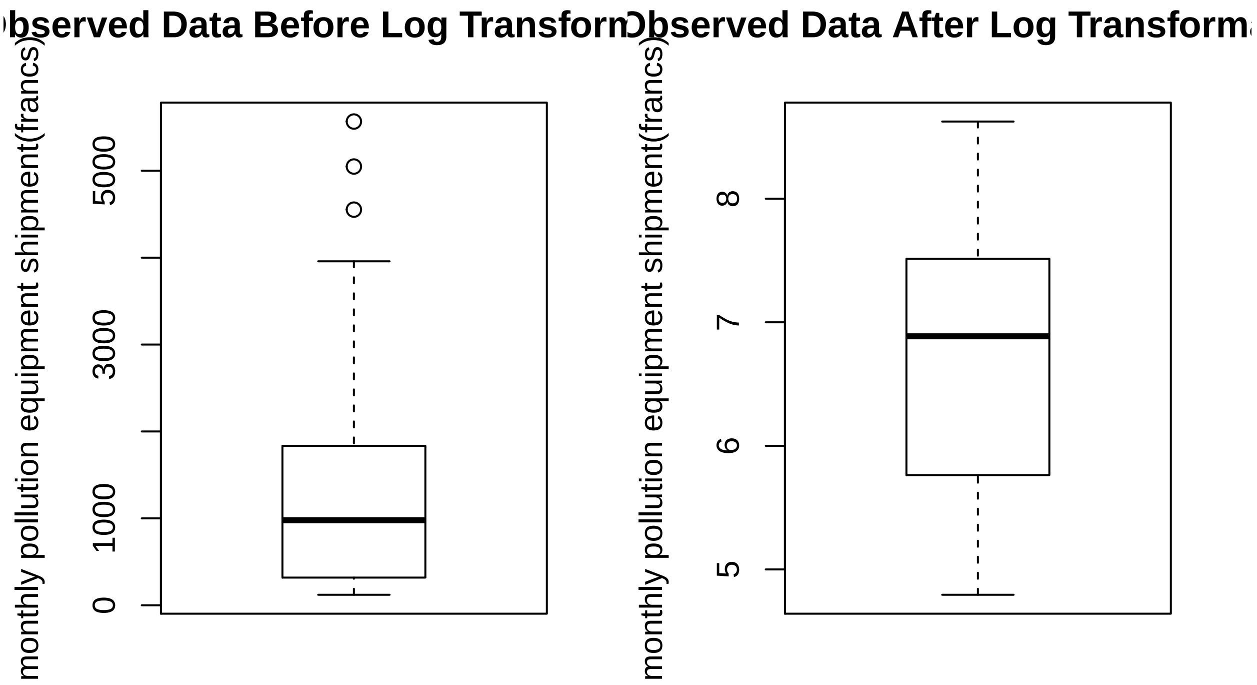
Let's take this log transformed data and fit a regression line. Notice how the residuals appear closer to constant variance – a definite improvement!

```
trend.fit = lm(transx~t)
plot(residuals(trend.fit), xlab="time")
```



As a side note, also notice how the three outliers are minimized after performing the log transformation.

```
#check for outliers
par(mfrow=c(1,2))
boxplot(x, main = "Observed Data Before Log Transformation", ylab = "monthly pollution equipment shipmen
boxplot(log(x), main = "Observed Data After Log Transformation", ylab = "monthly pollution equipment shi
```



Now that variance is stablized, we can go on to next steps in time-series analysis, which we won't cover here.

Variable Transformation in Linear Regression

See attached document ‘sta108_project’ in folder.