Post01 Exploratory Data Analysis

Jing Wang

10/31/2017

Introduction

Exploratory data analysis, or EDA, is the examination of data and relationships among variables, through both numerical and graphical mothods. In real life, it can be very useful in terms of leading to insights, new questions, or the process of building predictive models. It is an opportunity to check some of your assumptions and intuitions about the data set, and it takes place before more formal, more rigorous statistical analyses.

Goals of EDA

So what exactly is exploratory data analysis? The Wikipedia definition is: “exploratory data analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics, often with visual methods. A statistical model can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modeling or hypothesis testing task.”(https://en.wikipedia.org/wiki/Exploratory_data_analysis)

First Step

Exploratory data analysis is an opportunity to let the data surprise you. Think back to the goals of your investigation. What question are you trying to answer? That question might be relatively simple or one dimensional like the comparison of two groups on a single outcome variable that you care about. Even in such a case, exploratory data analysis is an opportunity to learn about surprises in the data. Features of the data that might lead to unexpected results. It can also be an opportunity to learn about other interesting things that are going on in your data. What should you do first? Well certainly you want to understand the variables that are most central to your analysis, often, this is going to take the form of producing summaries and visualizations of those individual variables.Check out https://www.udacity.com/course/data-analysis-with-r--ud651 for more information.

Some Approaches

I really like the way http://www.itl.nist.gov/div898/handbook/eda/section1/eda11.htm introduces to the audience the powerfulness of EDA. Here I’ll quote directly from the website. “Most EDA techniques are graphical in nature with a few quantitative techniques. The reason for the heavy reliance on graphics is that by its very nature the main role of EDA is to open-mindedly explore, and graphics gives the analysts unparalleled power to do so, enticing the data to reveal its structural secrets, and being always ready to gain some new, often unsuspected, insight into the data. In combination with the natural pattern-recognition capabilities that we all possess, graphics provides, of course, unparalleled power to carry this out.”

The website also provides some graphical approaches to start off our EDA. And graphical representation of the data can be very helpful to analysts.In the graphs are the stories of the data narrated! So here we go:

- Plotting the raw data (such as data traces, histograms, bihistograms, probability plots, lag plots, block plots, and Youden plots.
- Plotting simple statistics such as mean plots, standard deviation plots, box plots, and main effects plots of the raw data.
- Positioning such plots so as to maximize our natural pattern-recognition abilities, such as using multiple plots per page.

A lot of the methods are already covered in Statistics 133. I’ll briefly explain some of the graphical techniques that we have not seen in class, using our NBA players data https://github.com/ucb-stat133/stat133-fall-2017/blob/master/data/nba2017-player-statistics.csv.

Run Charts with R

A run chart is a simple line graph of a measure over time with the median shown as a horizontal line dividing the data points so that half of the points are above the median and half are below.The main purpose of the run chart is to detect process improvement or process degradation, which will turn up as non-random patterns in the distribution of data points around the median.

Here is an example using the nba2017-players-statistics.csv to make a simple run chart of the salaries of all players:

Firstly, let’s download and import the data:

```
github <- "https://github.com/ucb-stat133/stat133-fall-2017/raw/master/"
file <- "data/nba2017-player-statistics.csv"
csv <- paste0(github, file)
download.file(url = csv, destfile = 'nba2017-player-statistics.csv')
dat <- read.csv(file="nba2017-player-statistics.csv")
```

Select what we want to manipulate, this case, our salaries.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

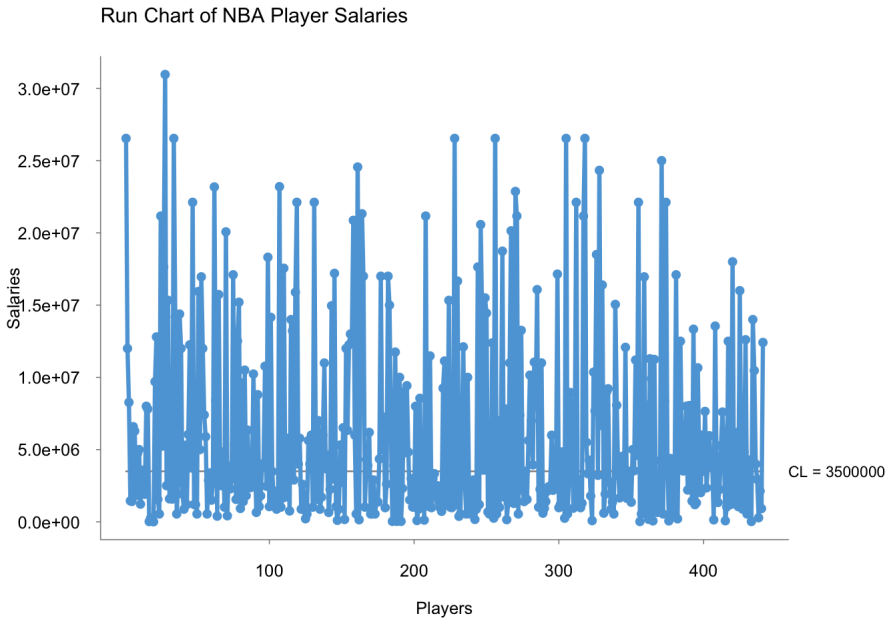
```
sal <- dat$Salary
```

Now we can start graphing

```
library(qicharts)
```

```
## qicharts will no longer be maintained. Please consider moving to qicharts2: https://anhoej.github.io/qicharts2/
.
```

```
qic(sal, main = "Run Chart of NBA Player Salaries", xlab = "Players", ylab = "Salaries")
```



This is just a very basic usage of the run chart; for a detailed documentation of run charts, check out https://cran.r-project.org/web/packages/qicharts/vignettes/runcharts.html.

Lag Plot

A lag plot is used to help evaluate whether the values in a dataset or time series are random. If the data are random, the lag plot will exhibit no identifiable pattern. If the data are not random, the lag plot will demonstrate a clearly identifiable pattern. The type of pattern can aid the user in identifying the non-random structure in the data. Lag plots can also help to identify outliers. Check out https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Lag_Plots.pdf.

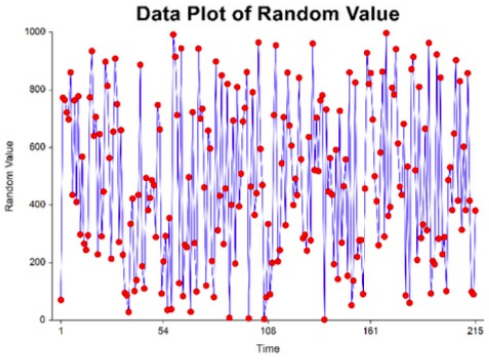
Definition of Lags

For data values Y1, Y2, …, YN, the k-period (or kth) lag of the value Yi is defined as the data point that occurred k time points before time i. That is Lagk(Yi) = Yi-k

Lag Plot Patterns

Random Data

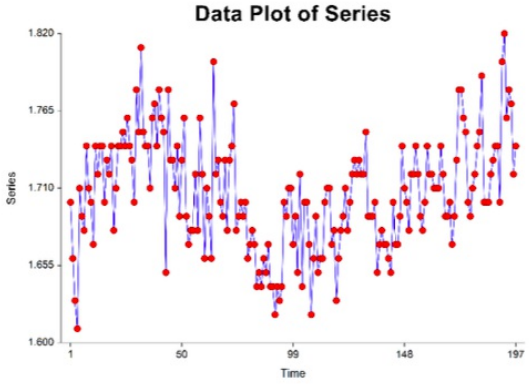
Random data gives rise to lag plots with no pattern. The points in the lag plot appear scattered from left to right and top to bottom.

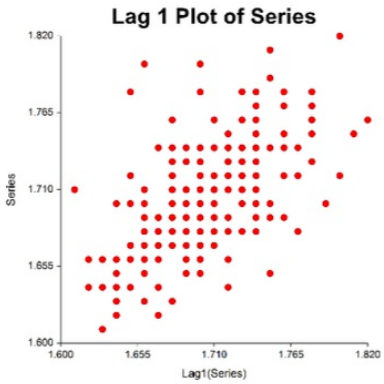


Data with Autocorrelation

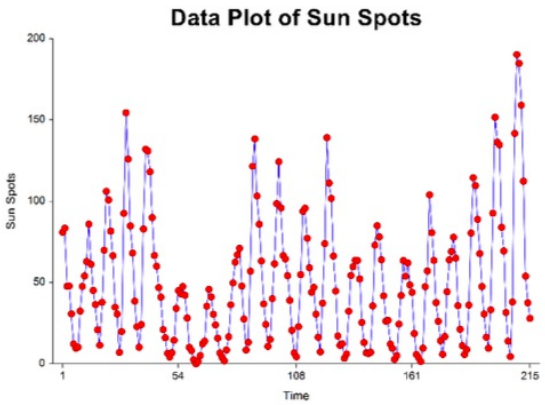
Data with autocorrelation gives rise to lag plots with linear patterns that follow the diagonal. As the level of autocorrelation increases, the points cluster more tightly along the diagonal.

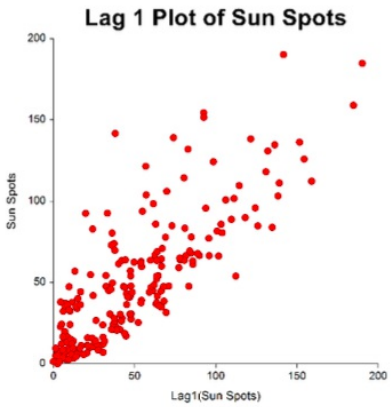
Data with Weak Autocorrelation



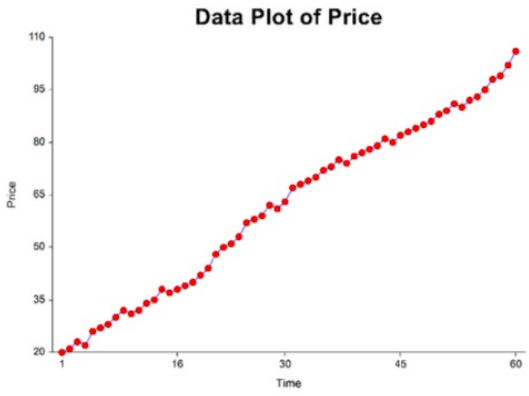


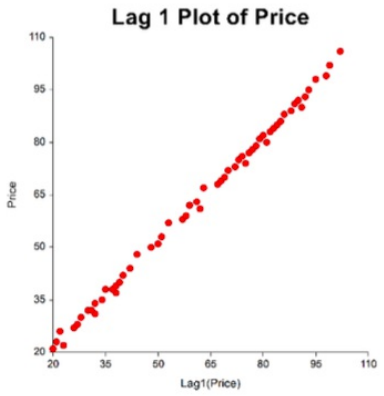
Data with Moderate Autocorrelation





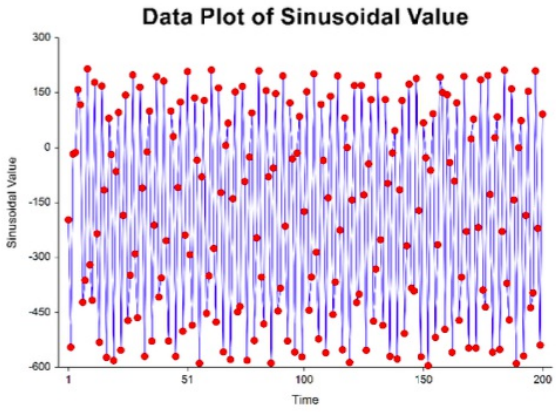
Data with High Autocorrelation

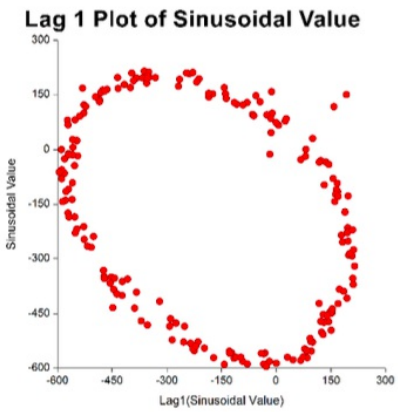




Sinusoidal Data

Single-cycle sinusoidal data gives rise to lag plots with circular or elliptical patterns. Values lying off the ellipse should be considered as potential outliers.





Youden Plots

Youden plots are a graphical technique for analyzing interlab data when each lab has made two runs on the same product or one run on two different products. The Youden plot is a simple but effective method for comparing both the within-laboratory variability and the between-laboratory variability. Check out http://www.itl.nist.gov/div898/handbook/eda/section3/youdplot.htm for more infomation.

Take Home Message

In conclusion, EDA is a useful step to take that tells us some important facts and relations between the variable before we dive deeply into the data. There are definitely a lot more about EDA than what I have put on for this post. Feel free to check out any of the reference that I mentioned and learn more about utilizing this powerful approach to dealing with real life data.