

## Data Analysis Preliminaries

### Overview

This is a very simple “warm-up” exercise. The main purpose of the assignment is to begin thinking about and analyzing data and manipulating it via a program.

### Background

Suppose you have self-published a book on Amazon. Their Analytics division provides you with periodic summaries of the number of purchases of your book. The datafile consists simply of a list of “downloads per hour” for the entire previous month (31 days).

You would like to get a feel for the popularity of your book. Are more people buying it the longer it’s on the market? Has “word of mouth” had any effect, or has interest in it begun to tail off? What kind of sales can you expect in the future?

### Specifications

You decide to download the analytics data, visualize it, and produce some kind of “trend line” in order to predict future performance. That’s basically the assignment.

#### 1. Pre-processing: read in and clean the data

The datafile (`downloads.txt`) comes as a comma-separated list of values: each line contains the hour of the month ( $24 \times 31$ ) and the number of downloads that occurred during that hour (e.g. 1,2272). A quick glance at the file shows that some type of error has prevented the data from being measured and/or recorded at certain times, represented as a ‘nan’ (“not a number”) value in the datafile. You’re going to have to do something to deal with this problem. Develop, document, and justify a solution.

#### 2. Visualization: display the data

In order to get an initial feel for the data, the next step is to visualize it. Create a scatterplot (a Cartesian display of two-variable data). It would be nice if you could do this within your program, using a graphics library (e.g. matplotlib, R, gnuplot). But you can default to Excel if you do not already know a graphing API. Document and describe your approach. What does the visualization tell you about your book sales?

### 3. Analysis: perform simple linear regression on the data

You decide to begin with the simplest analysis: linear regression. Linear regression is a method for fitting a curve, in this case a straight line, to a set of points. The slope of the line represents the correlation between the  $x$  and  $y$  values; the intercept gives the center of mass of the data points.

There are several different ways of performing a linear regression, typically based on the *least-squares* method that attempts to minimize the sum of squared residuals (i.e. the error). A simple method follows.

First obtain/calculate:

- $\Sigma X$ : the sum of all  $X$  values
- $\Sigma Y$ : the sum of all  $Y$  values
- $\Sigma XY$ : the sum of the products of each  $X, Y$  pair
- $\Sigma X^2$ : the sum of the squares of every  $X$  value
- $\Sigma Y^2$ : the sum of the squares of every  $Y$  value

Suppose  $N$  is the number of data points. Then the relevant calculations are:

$$\text{slope} = \frac{(N \sum XY) - (\sum X \sum Y)}{(N \sum X^2) - (\sum X)^2}$$

$$\text{intercept} = \frac{\sum Y - (\text{slope} \sum X)}{N}$$

With these values you can create the regression equation:

$$Y = \text{intercept} + \text{slope} * X$$

and use it to make predictions about the future.

Perform the following:

- Using the regression equation, how many downloads would you expect at Noon on the fifth day of the next month?
- Create a visualization of the regression analysis (i.e. plot the trendline over the scatter plot of the data).

Note: apparently a well-read blogger made a favorable mention of your book towards the end of the month. Do you think this simple linear analysis captures the expected popularity of your book? Explain. What other analytical approaches might produce a better model?

Notes:

- You may use the programming language/platform of your choice. However, all computations of the basic assignment should be performed by your program, *not* by a built-in library routine. However, you are encouraged to compare your results to packaged routines (e.g. Excel, R, SAS).
- Be sure to demonstrate good programming style and practices.
- You may work together on this assignment.

Deliverables

- Submit a hard-copy of your source-code, sample output, and a design document describing your approach and justifying your choices.
- Be prepared to present and discuss your solution in class. E.g. what data structures did you employ? What graphing package/API did you use? What interesting problems did you encounter and how did you address them? What alternative analyses did you perform?