Homework 3

Calin Capitanu

October 3, 2021

Chapter 1

Two Research Studies

1.1 Code Comments

a Can we draw the conclusion that comments improve the code? Explain

In my opinion, there is no good conclusion that can be drawn in a generalized form from a simple experiment like this. This is mostly because the dataset of the experiment is too little, and the conclusions drawn from it are too generalized.
Besides the term "best" being hardly described, and considered not relevant for this, which in my opinion, is wrong, as intuition could lead to "best" meaning performance in speed and capacity, which comments are unrelated to, but to the process of development, they could be. This is extrapolating the different known and unknown parts of the problem, but the main idea is that there can be no hard conclusions being drawn after such a small experiment, with only 2 data points.

b Suggest improvements to the study!

As the previous part suggests, that the wrong part here in my opinion is the size of the dataset. In this case, I think the best way of improving the experiment and the conclusions from it are to increase the size of the sample data, as well as to formulate the conclusion as if was part of an experiment and that the probability of a program being "better" is when it has comments.
Moreover, I think that defining the "best" term for a program is crucial, as one can also better describe which program is better with more scientific proof and less statistical results.

1.2 Functional Programming

a Can we draw the conclusion that functional programming languages give slower code? Explain!

Similar to the previous example, in my opinion, the problem space is too large, while the input data set is too little in this programming language example. That is, only one problem has been tested, only one functional programming language has been tested and also that only one solution to the problem is taken into account. With this little amound of information, there is no way we can draw definite conclusions to such generalized statements such as "functional programming languages being slower". At the same time, the fact that two different teams were assigned to this experiments also means that the solution conceptually could highly differ.

b Suggest improvements to the study!

In order to improve this experiment, I think that the best way is to define a smaller sample space of the problem, thus first things first trying to define a smaller goal conclusion. In order tu study if the wanted conclusion holds, we need to take into account less variables unrelated to the problem (such as different solutions from different teams).
First things first, I suggest that the same team, with the same conceptual solution codes the same problem into the two different programming languages. Secondly, I suggest that the conclusion data is restricted only to Haskell, and not to functional programming languages in general, since they could differ a lot. Finally, I think that the best way to rephrase the conclusion is to add some sort of statistical or probability factor. That is, even if after all the previous points, Haskell still yield slower times than Java for this specific problem and some others, there could be some other problem in which the result is the opposite, thus bringing the uncertainty factor of the term "on average", the conclusion could be more feasible.

Chapter 2

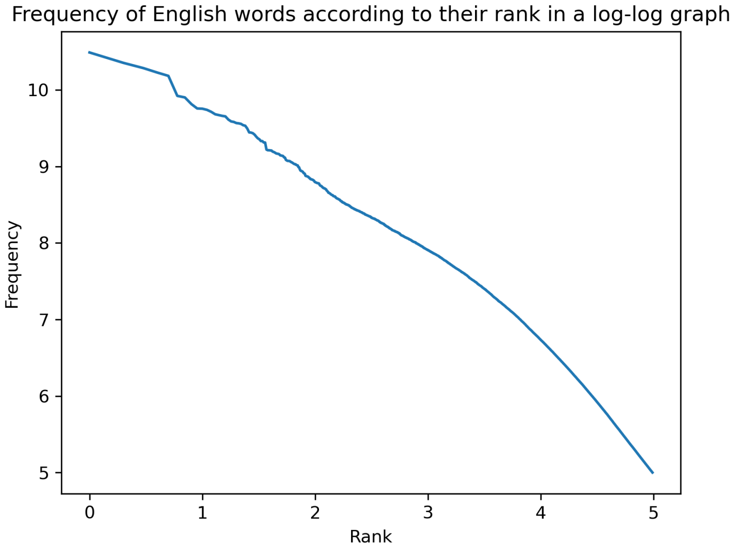
Zipf's law

a Dataset

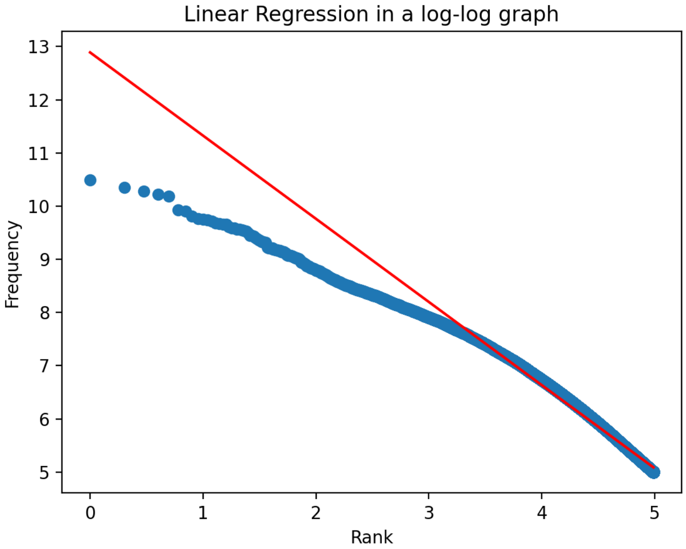
The dataset that I chose is the frequency of words in English, since the format offered allows for easy reading into a programming language of choice and proper formatting and analysis from that point.

b Graph

The graph below (Figure 2.1) represents the plot of English word frequency plotted against the rank of their appearance in a log-log plot.



c Linear Regression



The results here show a pretty well fit in the latter part of the graph. Function's coefficient sits around -1.563, since I inputted the points already under the log function. But then, since we want the un-logged coefficient, I will get 0.027307183730692054.
If we take the linearly fitted line, the data holds with Zipf's law.

d Scientific Article

In the scientific article "Zipf's Law for Word Frequencies: Word Forms versus Lemmas in Long Texts", teh results shown are a bit different than the one here. To start with, the graphs are relatively similar under the log function, but the coefficient resulted in my research is a bit below 10 times smaller than the cited paper. [1]

e Scientific Statement

In my opinion, I think that Zipf's Law is a conjecture. A conjecture is an opinion or a conclusion formed on the basis of inclomplete information. That is, this is not a theory that can always hold, since you do need to find a linear fit in order to make it "more true". In this case, this is just some sort of approximation of observable data, in which case, I think it is a conjecture.

References

[1] A. Corral, G. Boleda, and R. Ferrer-i Cancho, “Zipf’s law for word frequencies: Word forms versus lemmas in long texts,” *PLOS ONE*, vol. 10, no. 7, pp. 1–23, 07 2015. doi: 10.1371/journal.pone.0129031. [Online]. Available: https://doi.org/10.1371/journal.pone.0129031