Guerrilla Automation with R

*R automation despite resource constraints*

**Introduction**

Guerrilla warfare is a type of asymmetric warfare wherein a smaller, less powerful military uses unconventional tactics to fight a stronger, more traditional military. The definition via Google of the adjective form of guerilla is “Referring to actions or activities performed in an impromptu way, often without authorization.” When a problem calls for a powerful technological solution, but no such solution is currently in-place, actuaries may require a low-cost, light-weight approach to meeting the needs of their stakeholders.

That’s where R comes in. R is a free and open source programming language with a powerful and continually-expanding set of libraries and packages. It is highly versatile and compatible with a wide variety of other systems. I am going to walk through a hypothetical use-case for R that highlights R’s advantages. Example code will be interspersed throughout the article. The full code, complete with example excel files can be accessed via Github at the following url:

<https://github.com/TimothyQuast/Guerrilla_Automation>

**The Use-Case**

* Describe the use-case and highlight key features.
  + Need to gather a moderate amount of data stored in a regular format
  + Lack of centralized data repository or need to refer to non-centralized data
  + Data is stored in excel or another annoying file-type/structure
  + Lack of impetus and/or resources to fix the previous two problems
* Why is R suitable for the task?
  + It’s free!
  + The open source nature and broad user base makes for a prolific developer community which continues to produce marvelous and effective packages.
  + Extremely compatible with a wide variety of data-types and systems, especially due to the large number of packages available
  + Well-suited for rapid prototyping
  + Some disadvantages
    - R can struggle with processing speed, so scalability is an issue.
    - R can be difficult to learn, making portability of the process (to other personnel) challenging.

The Solution

* Organize inputs by putting information need to gather data (e.g. file paths) into a spreadsheet or data table.
* Validate the inputs (e.g. are the file paths correct? Which ones are wrong?).
* Pull the data using the validated inputs.
* Organize/Summarize the data.
* Additional Commentary
  + If the data is not in a regular format, can use inputs to correct the process
  + Can summarize at different levels of aggregation, but should ensure ability to trace data back to inputs for ease of validation and auditability

Appendix

* Example Code