QUANTITATIVE SKILLS AND TOOLS

QUANTITATIVE SKILLS

Out-of-academia context:

- apply **quantitative methods** to (business) problems in order to obtain **actionable insight**
- difficult for any given individual to have expertise in **every** field of mathematics, statistics, computer science, data science, data engineering, etc.

With a graduate degree in math/stats, for instance:

- **expertise** in 2-3 areas
- **decent understanding** of related disciplines
- **passing knowledge** in various domains

Flexibility is an ally, perfectionism… only up to a point.

QUANTITATIVE SKILLS

Suggestions:

- **keep up with trends**
- become **conversant in your non-expertise areas**
- know **where to find information**

In many instances (70%?), only the basics (2nd–3rd year mandatory courses at uOttawa, say) are sufficient to meet government/industry needs.

Focus: make sure you really **understand** the basics, stepping stones.

In the rest of the cases, more sophisticated knowledge is required.

QUANTITATIVE SKILLS

- survey sampling and data collection
- data processing and data cleaning
- data visualization
- mathematical modelling
- statistical methods
- regression analysis
- queueing models
- machine learning
- deep learning
- reinforcement learning
- stochastic modelling (MC simulations)
- optimization and operations research
- survival analysis
- Bayesian data analysis
- anomaly detection and outlier analysis
- feature selection/dimensions reduction
- trend extraction and forecasting
- cryptography and coding theory
- design of experiment
- graph and network theory
- text mining/natural language processing
- etc.

SOFTWARE AND TOOLS

Modern quantitative work typically involves **programming** (or the use of point-and-click software, at the very least).

But programming languages **go in and out of style.**

It is important not just to understand the syntax of a particular language, but also how computer languages and computing infrastructure work in general.

ALSO: avoid getting caught up in programming/tool wars … they're more or less all functionally equivalent!

SOFTWARE AND TOOLS

Programming (and Related)

- Python, R, C/C++/C#, Perl, Julia, regexps (, Visual Basic?), Java, Ruby, etc.

Database Management

- SQL and variants, ArangoDB, MongoDB, Redis, Amazon DynamoDB (, Access?), Big Query, Redshift, Synapse, etc.

Data Visualization

- ggplot2, seaborn, plot.ly, Power BI, Tableau, D3.js, Google Data Studio, proprietary software, etc.

Simulations, Statistical Analysis, Data Analysis, Machine Learning

- tidyverse, scikit-learn, numpy, pandas, scipy, MATLAB, Simulink, SAS, SPSS, STATA (, Excel?), Visio, TensorFlow, keras, Spark, Scala, etc.

Typesetting and Reporting

- LaTeX, R Markdown, Adobe Illustrator, GIMP (, Word?, PowerPoint?), etc.

SOFTWARE AND TOOLS

Q: At StatCan, R or SAS?

A: Not easy to answer as StatCan is in a slow transition period. The Agency is better equipped for SAS (with “Big Data” options, such as SAS Grid).

R is […] not as ideal for large files (e.g., Census data), so it is not an option in such cases because it is still too slow (unless you have very powerful servers). But we would prefer to use the R packages, so it’s a dilemma.

TL;DR: R is our future, but SAS is still very much our present. In times of transition, **analysts/employees who know both are better positioned**.

QUANTITATIVE SKILLS AND TOOLS