Recitation 4

Measurements

Shreyans Sheth
June 10th, 2020

What is Measurement in formal terms ?

IEEE 1061 says: “A software quality metric is a function whose inputs are software data and whose output is a single numerical value that can be interpreted as the degree to which software processes a given attribute that affects its quality.”

Can we be more generic ?

A quantitatively expressed reduction of uncertainty based on one or more observations. – Hubbard, “How to Measure Anything …”

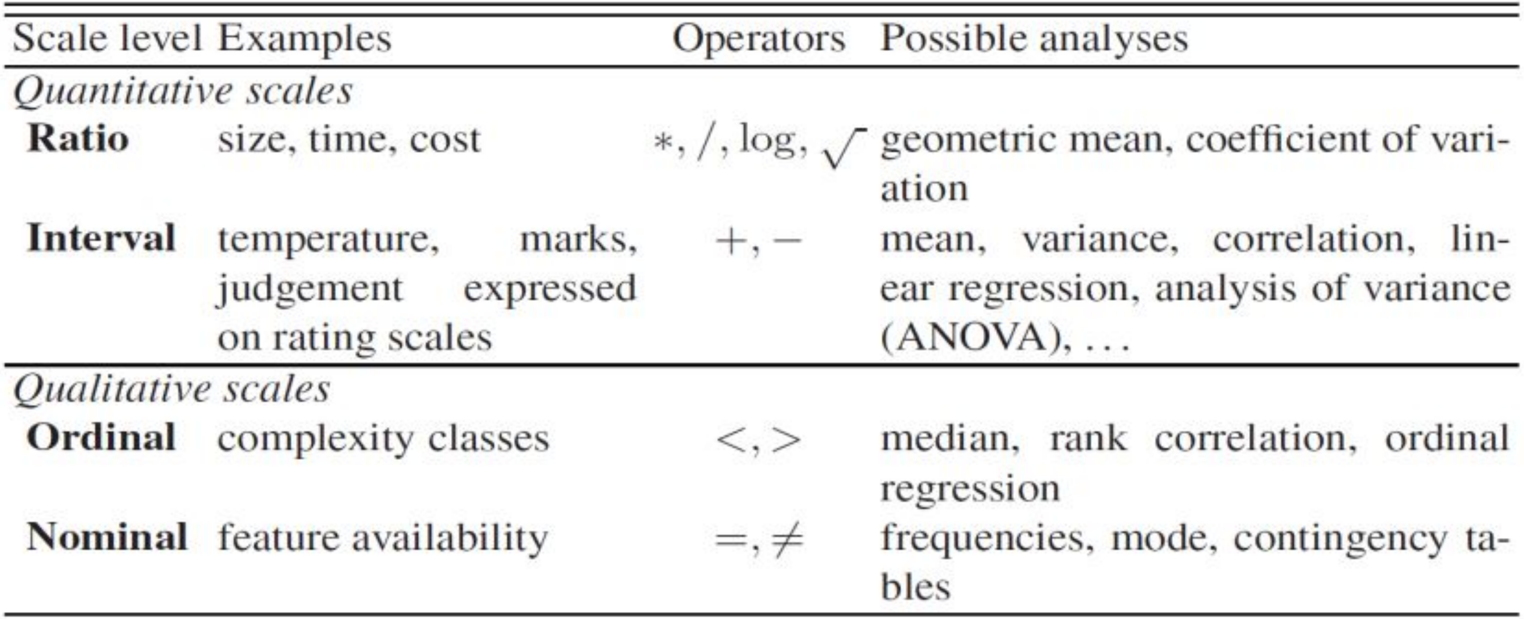
Why is this ***‘better’*** ?

- If we incorrectly think that measurement means meeting some nearly unachievable standard of certainty, then few things will seem measurable

- The fact that some amount of error is unavoidable but can still be an improvement on prior knowledge is central to experiments, surveys etc.

- “uncertainty reduction” is what is critical to business. Major decisions (worth millions) made under a state of uncertainty can be made better, even if just slightly, by reducing uncertainty.

Scales of Measurement



Scale level	Examples	Operators	Possible analyses
Quantitative scales			
Ratio	size, time, cost	$*, /, \log, \sqrt{\ }$	geometric mean, coefficient of variation
Interval	temperature, marks, judgement expressed on rating scales	$+, -$	mean, variance, correlation, linear regression, analysis of variance (ANOVA), . . .
Qualitative scales			
Ordinal	complexity classes	$<, >$	median, rank correlation, ordinal regression
Nominal	feature availability	$=, \neq$	frequencies, mode, contingency tables

Nominal: categories • **Ordinal**: order, but no magnitude. • **Interval**: order, magnitude, but no zero. • **Ratio**: Order, magnitude, and zero

The Clarification Chain

1. If it matters at all, it is detectable/observable.
2. If it is detectable, it can be detected as an amount or range of possible amounts.
3. If it can be detected as a range of possible amounts, it can be measured.

Eg, *‘System should be scalable’*

What does this *really* mean ? Can we break it down using the clarification chain and measure scalability ?

Case Study: The PLASTER Framework

PLASTER [1] is a framework used by NVIDIA which describes the key elements for measuring deep learning performance.

- **P**rogrammability
- **L**atency
- **A**ccuracy
- **S**ize of model
- **T**hroughput
- **E**nergy efficiency
- **R**ate of learning

PLASTER - Programmability

- How ‘easy’ or ‘efficient’ is it to program in a given deep learning framework, say Caffe vs Tensorflow vs Pytorch ?
- The the platform support fast iteration and good integration with XYZ platform, allowing the business to be first to market ?
- **Interpretability**
  - How easy is it to debug code written in this language or framework
  - Is it comprehensible ? LoC ?
  - Could a decision tree provide more insight into decisions as opposed to a DL model in the first place ?

How can we measure this ? What scale would we choose ?

PLASTER - Latency

- Latency is the time between requesting something and receiving a response. With most human-facing software systems, not just AI, the time is often measured in millisecond (**what scale is this ?)**
- Voice recognition (unnatural lag?), Image classification?
  - Google has stated that 7 milliseconds is an optimal latency target for image-and video-based uses
- Jakob Nielsen’s **0.1 / 1 / 10 second limits**
  - 0.1 - user feelds system is instant
  - 1 - User’s flow of thought stays uninterruped but they notice the delay.
  - Somewhere between 2 to 10 seconds, people start wondering if the system is still working properly.

PLASTER - Accuracy

- Measuring model accuracy in runtime inferences
- Naturally, one model being more accurate under given circumstances with everything else held constant is a better choice.

- Traditional measurement techniques
  - Accuracy
  - Precision
  - Recall
  - MAE
  - MSE

PLASTER - Size

- With size and complexity of models increasing, we are able to detect far more detailed analysis.
- However, the size of a deep learning model and the capacity of the training infrastructure have impacts on performance.
- If an engineer constrains a model to be within a particular size bound, the accuracy could be affected as number of connections and layers is reduced..

- How Plaster measures size of DL networks:
    - Number of layers
    - Number of nodes (neurons) per layer
    - Complexity of computation per layer
    - Number of connections between a node at one layer and the nodes of neighboring layers

PLASTER - Throughput

- How much data can be predicted per unit of time for a given threshold of latency ?
- Without the appropriate balance of throughput and latency, the result can be poor customer service, missing (SLAs), and potentially a failed service.

- How can we measure throughput:
  - images-per-second for image-based networks
  - tokens-per-second for speech-based networks

PLASTER - Energy Efficiency

- Power consumption can hugely increase costs of delivering a service, driving a need to focus on energy efficiency in both devices and systems.
- Datacenter inference providing real-time processing for speech can easily involve large racks of machines that can impact a company's total cost of ownership (TCO).

- Measuring energy efficiency:
  - Inferences per watt. The higher the better
  - Note! if one processor is pulling 200W vs. another pulling 130W, that does not necessarily mean the 130W system is better. If the 200W solution completes the task 20x faster, it is more energy efficient.

PLASTER - Rate of Learning

- Focus on DevOps, continuous retraining and deployment.
- Faster training times mean that developers can retrain their networks more often to improve accuracy or maintain high accuracy
- Going to be covered in the future - Telemetry, Deployment etc.

- Measuring Rate of Learning:
  - Improvements in
    - Throughput
    - Programmability
    - size of model
    - energy efficiency

PLASTER - Conclusion

- PLASTER is greater than the sum of its parts. Not to be taken at face value
- Applying deep learning can be complex and is in the early stages of its life cycle. With a clear framework like PLASTER, organizations can take advantage of its vast potential.
- Measuring which DL model was ‘better’ than another was a hard problem, but defining metrics and creating a framework makes it a whole lot easier

Thanks!

References

1. Plaster Framework
2. Metrics to evaluate your ML model
3. How to Measure anything - The intangibles in business