Introduction to Machine Learning

Practical Advice for Building Machine Learning Systems

Varun Chandola

Computer Science & Engineering
State University of New York at Buffalo
Buffalo, NY, USA
chandola@buffalo.edu

University at Buffalo
Department of Computer Science and Engineering
School of Engeering and Applied Sciences

Outline

From Theory to Practice

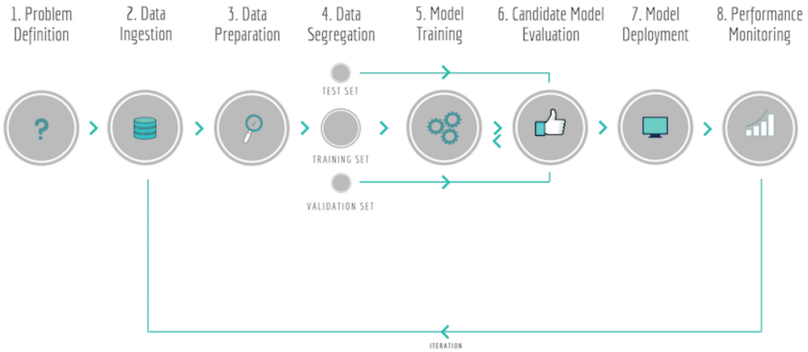
Problems that we can solve

- Given $(\mathbf{X}, \mathbf{y})$, learn a model to predict y^* for a test $\mathbf{x}^*$
- Or the unsupervised or RL variant
- Many tools at our disposal

Problems that we need to solve

- Predict outages in a massive data center
- Build an automated insulin pump
- Design a next generation space propulsion system

A General ML Pipeline



- Sometimes we need to go back to the problem definition itself

Debugging an ML Pipeline

- Is the pipeline **good**?
- How do we define goodness?

Performance

- Cross-validation performance
- Generalizability

Costs

- Computing
- Data

Acceptance

- Fairness
- Interpretability
- Privacy preserving
- Ethical

How to measure goodness?

- Ideally we want the model to be **generalizable**
- Two things that we need:
  - Good validation data (out of sample)
    - Random sampling is not always enough
  - Robust evaluation metric
- What if we do not have enough validation data?
  - Get more data (manual work, Mechanical Turk, synthetic)
  - Test for stability

What do we do if the model is not good? I

- Change the model
  - Make model more complex or simpler (??)
  - Incorporate domain knowledge
    - e.g., physics inspired neural networks
  - Handle structural dependencies
- Change the data
  - Feature selection/reduction
  - Representation learning (embedding)
  - More data
- Change the problem
  - New problem formulation

Finally - Is the model useful?

- Domain intepretation
- Stability of the model
- What do we if not useful?
  - Maybe solve a different (better) problem

Correlation is not causation

References