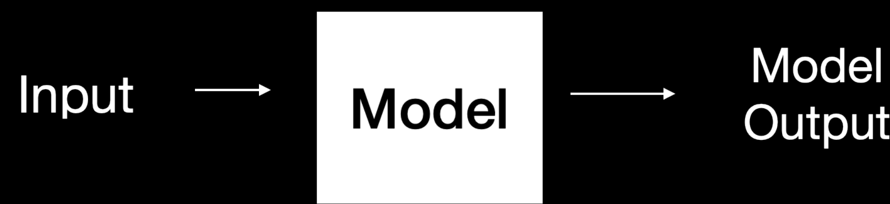
Quality of Machine-Learned Models

Jin Guo
SOCS McGill University

Machine Learning

- Constructing and/or learning the parameters of a specified model given existing data

Supervised Learning



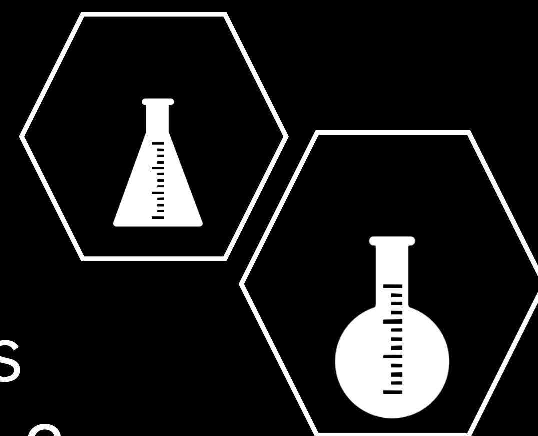
Input					Output
ID	Temperature	Cough	Sore throat	Headache	Flu
1	*37*	*Yes*	*No*	*No*	*No*
2	*39*	*No*	*Yes*	*No*	*No*
3	*39.2*	*Yes*	*No*	*Yes*	*Yes*

Features

How do you know the model is doing what you intended to do?

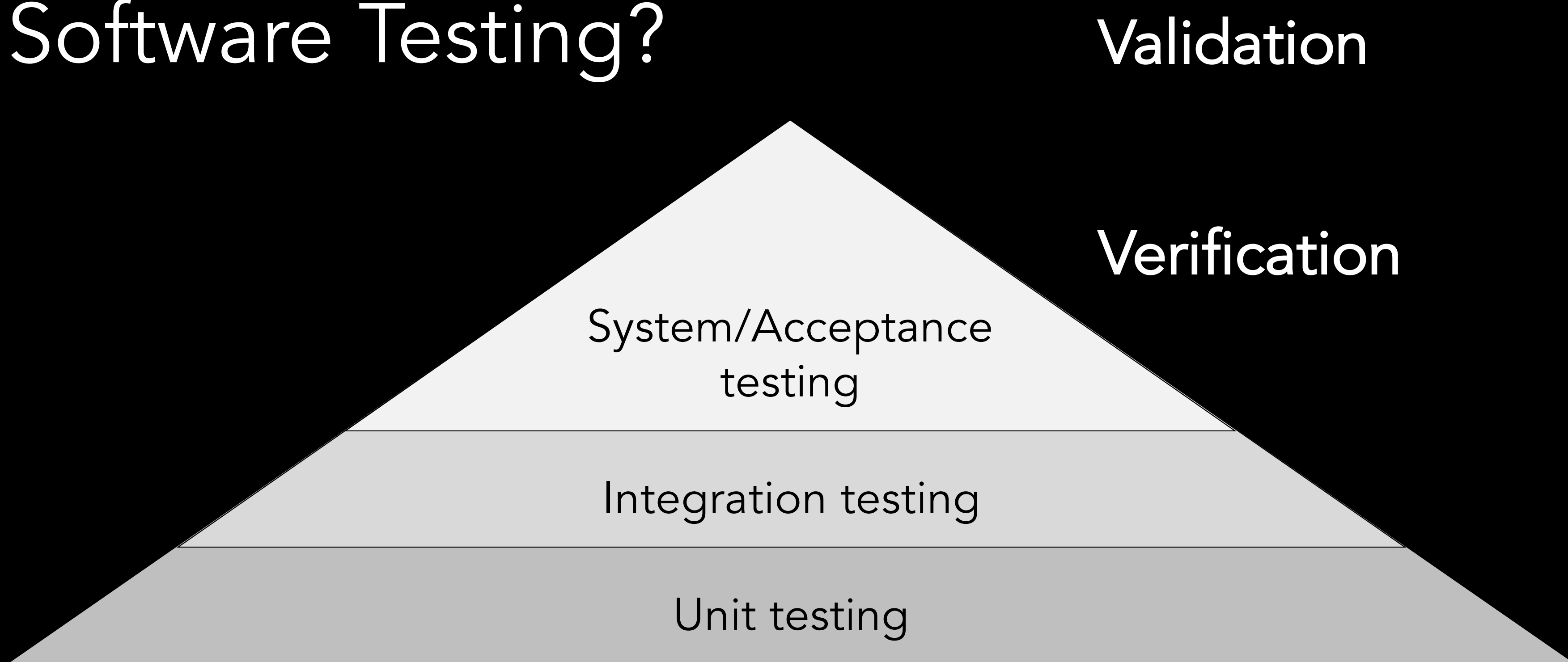
Data Scientists/Model Developers

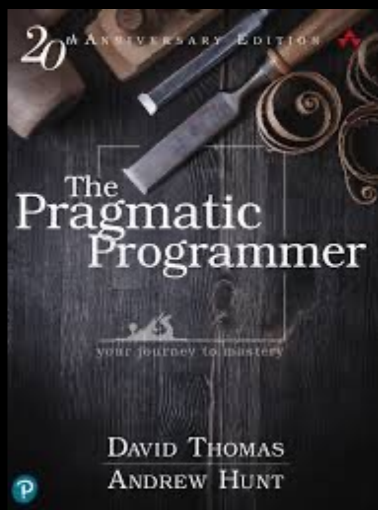
How do you know the model is doing what you intended to do?



How do you know a program is doing what you intended to do?

Software Testing?





"All software you write *will* be tested—if not by you and your team, then by the eventual users—so you might as well plan on testing it thoroughly … "

Test Case Example during Unit Test

```java
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.*;


class UndergradTest {

    @Test
    void getFirstName() {
        Student s = new Undergrad("001","Lily", "Joe");
        assertEquals("Lily", s.getFirstName());
    }
}
```

assertEquals method

Oracle

```
public static void assertEquals(Object expected,
                                Object actual)
```

What is the “unit” for ML software?

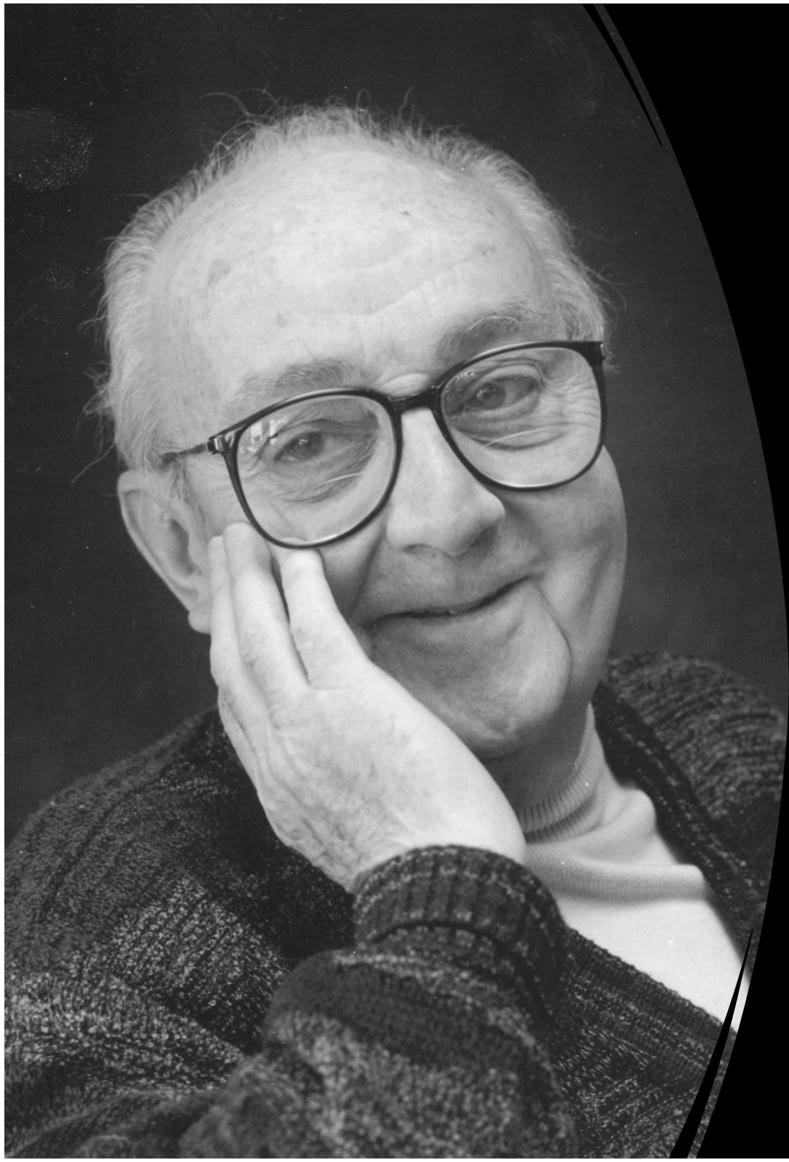
Activity 1

- How do you, as a model developer, consider a model is performing reasonable? You can draw from your experience.
- How do that compare model evaluation with the unit test practice for traditional software source code? What are the transferable consideration, and what are not?
- Summarize your comparison on Miro.

Model Evaluation VS Software Unit Testing

- Evaluation Means
- Evaluation Objective
- What do to in the case of unsatisfied evaluation outcome
- Quality of the evaluation itself

…

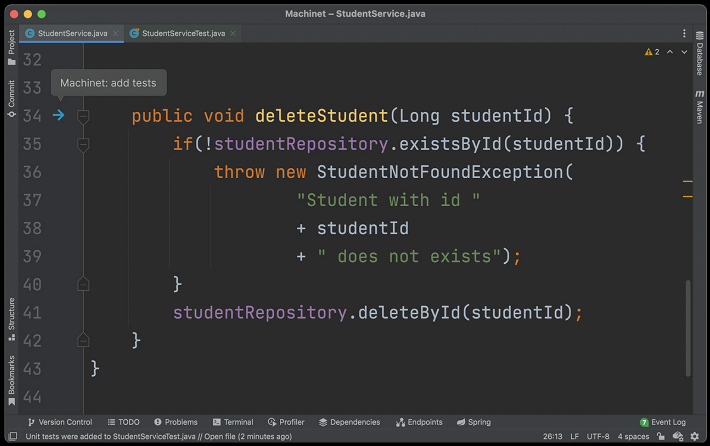


“All models are wrong, but some are useful.”

- George Box

Usefulness

- Model makes assumptions

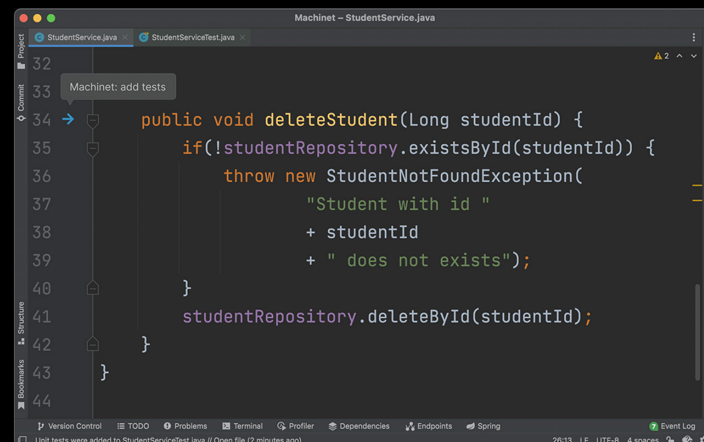


https://openai.com/blog/codex-apps/

https://machinet.net/#know-more

Usefulness

- Model makes assumptions
- Selection of the baseline

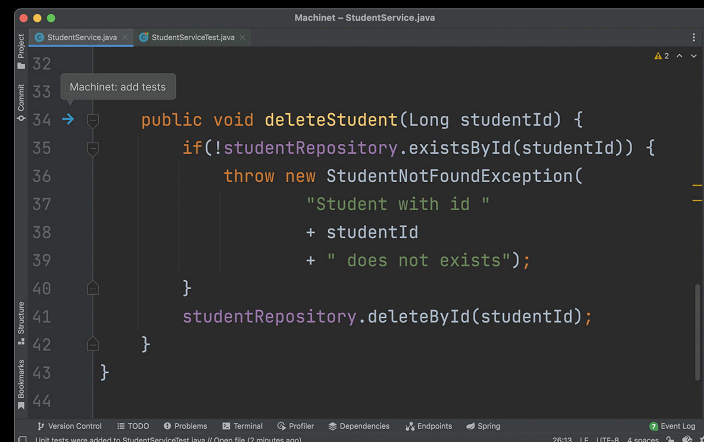


https://openai.com/blog/codex-apps/

https://machinet.net/#know-more

Usefulness

- Model makes assumptions
- Selection of the baseline
- Selection of evaluation methods



https://openai.com/blog/codex-apps/

https://machinet.net/#know-more

Activity 2

- What kind of metrics do you think are suitable to evaluate the model for Unit Test generation application?
- How are you going to calculate and optimize the selected metrics?
- What else are you going to test the model?

Selection of Evaluation Methods

- Choose Metrics
- Training/Validation/Testing split

Selection of Evaluation Methods

- Choose Metrics

- Training/Validation/Testing split

Pitfalls

Test data not representative

Misleading aggregated metrics

	Overall accuracy
Model A	96.2%
Model B	95%

Identify critical slices in your data.

Selection of Evaluation Methods

- Choose Metrics
- Training/Validation/Testing split

Pitfalls

Test data not representative

Misleading aggregated metrics

Data Leaking

Overfitting testing data

Capability	Min Func Test	INVariance	DIRectional
Vocabulary	Fail. rate=15.0%	16.2%	C 34.6%
NER	0.0%	B 20.8%	N/A
Negation	A 76.4%	N/A	N/A
...			

Test case	Expected	Predicted	Pass?
A Testing **Negation** with ***MFT*** Labels: negative, positive, neutral			
Template: I {NEGATION} {POS_VERB} the {THING}.			
I can't say I recommend the food.	neg	pos	X
I didn't love the flight.	neg	neutral	X
...			
Failure rate = 76.4%			
B Testing **NER** with ***INV*** Same pred. (inv) after removals / additions			
@AmericanAir thank you we got on a different flight to [Chicago → Dallas].	inv	pos neutral	X
@VirginAmerica I can't lose my luggage, moving to [Brazil → Turkey] soon, ugh.	inv	neutral neg	X
...			
Failure rate = 20.8%			
C Testing **Vocabulary** with ***DIR*** Sentiment monotonic decreasing (↓)			
@AmericanAir service wasn't great. You are lame.	↓	neg neutral	X
@JetBlue why won't YOU help them?! Ugh. I dread you.	↓	neg neutral	X
...			
Failure rate = 34.6%			

Figure 1: CheckListing a commercial sentiment analysis model (G). Tests are structured as a conceptual matrix with capabilities as rows and test types as columns (examples of each type in A, B and C).

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Usefulness

- Model makes assumptions
- Selection of the baseline
- Selection of evaluation methods

What else should we test for ML models

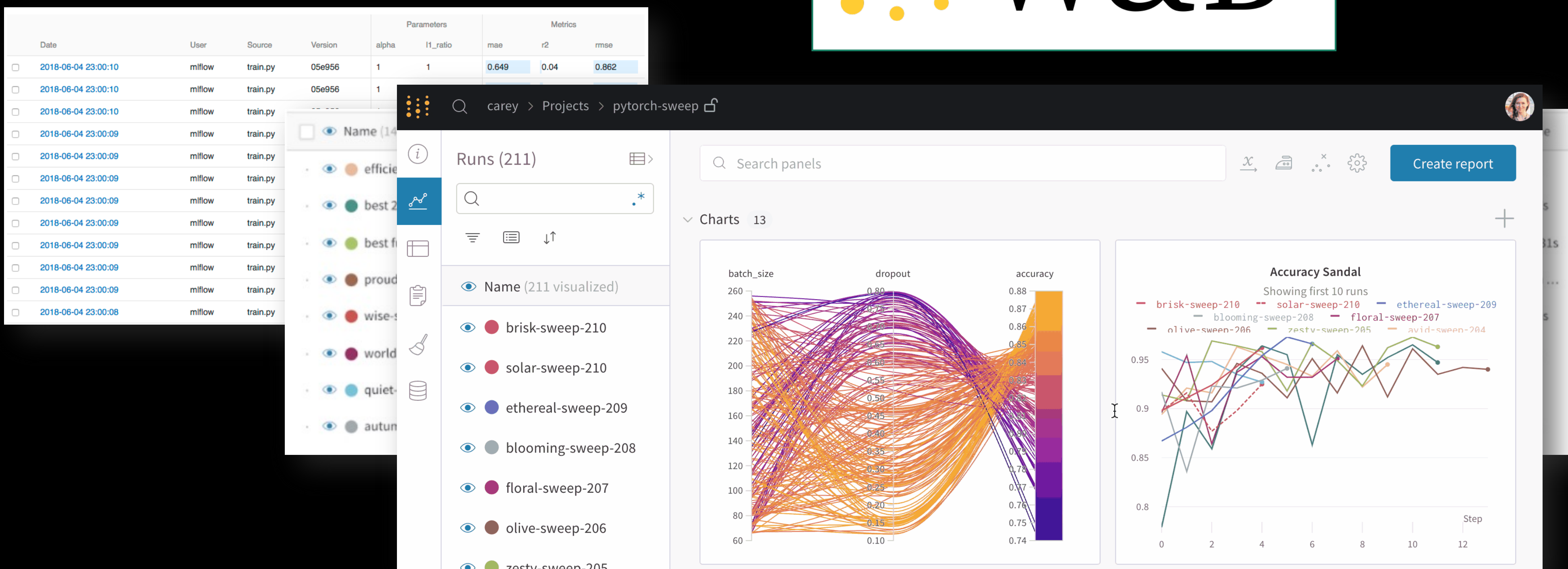
- Algorithmic Correctness (Design and Implementation)
  - Verify if the loss decreases after training for a few iterations
  - Overfit small dataset

What else should we test for ML models

- Algorithmic Correctness (Design and Implementation)
- Reproducible Training
    - Deterministically seed the random number generator
    - Initialize model components in a fixed order
    - Average several runs of the model.
    - Use version control (ML experiment management tools)

mlflow

W&B



Images from https://mlflow.org/docs/, https://docs.wandb.ai/

What else should we test for ML models

- Algorithmic Correctness (Design and Implementation)
- Reproducible Training
- Model Quality Degradation
  - Sudden or Slow degradation

Activity 3

- What content do/should you include in the model documentation?

- What content do you need as a model user?

Model Documentation

- Examples:
  - https://keras.io/api/applications/vgg/
  - https://huggingface.co/bert-base-multilingual-cased
  - https://modelcards.withgoogle.com/object-detection

Model Documentation

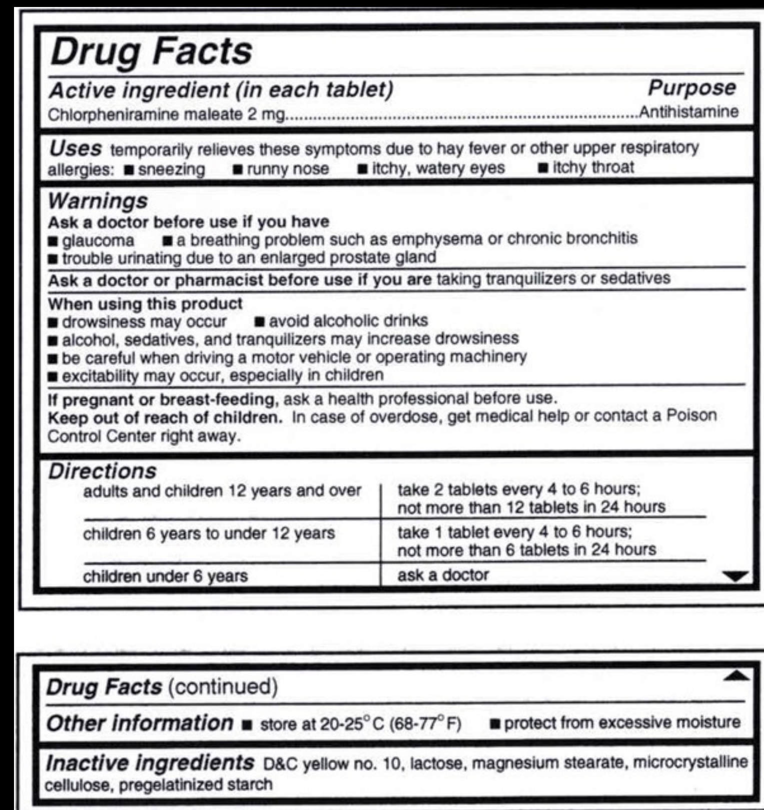


Image from: https://www.fda.gov/drugs/resources-you-drugs/over-counter-medicine-label-take-look

Model Card

Mitchell, Margaret, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. "Model cards for model reporting." In *Proceedings of the conference on fairness, accountability, and transparency*, pp. 220-229. 2019.

- **Model Details**. Basic information about the model.
  - Person or organization developing model
  - Model date
  - Model version
  - Model type
  - Information about training algorithms, parameters, fairness constraints or other applied approaches, and features
  - Paper or other resource for more information
  - Citation details
  - License
  - Where to send questions or comments about the model
- **Intended Use**. Use cases that were envisioned during development.
  - Primary intended uses
  - Primary intended users
  - Out-of-scope use cases
- **Factors**. Factors could include demographic or phenotypic groups, environmental conditions, technical attributes, or others listed in Section 4.3.
  - Relevant factors
  - Evaluation factors
- **Metrics**. Metrics should be chosen to reflect potential real-world impacts of the model.
  - Model performance measures
  - Decision thresholds
  - Variation approaches
- **Evaluation Data**. Details on the dataset(s) used for the quantitative analyses in the card.
  - Datasets
  - Motivation
  - Preprocessing
- **Training Data**. May not be possible to provide in practice. When possible, this section should mirror Evaluation Data. If such detail is not possible, minimal allowable information should be provided here, such as details of the distribution over various factors in the training datasets.
- **Quantitative Analyses**
  - Unitary results
  - Intersectional results
- **Ethical Considerations**
- **Caveats and Recommendations**

Recap

- Model Selection
  - Start from simpler models
  - Be aware of assumptions and compare the trade-offs
- Evaluation
  - Evaluate the quality of the models
  - Avoid the traps
  - Implementation Correctness
- Documentation
  - Document information beyond Model Performance

On Wednesday:

Model -> System