Carnegie Mellon University

QA: ML

17-313 Fall 2022

Rohan Padhye, Michael Hilton, Chris Timperley, and Daye Nam

S3D

Software and Societal Systems Department

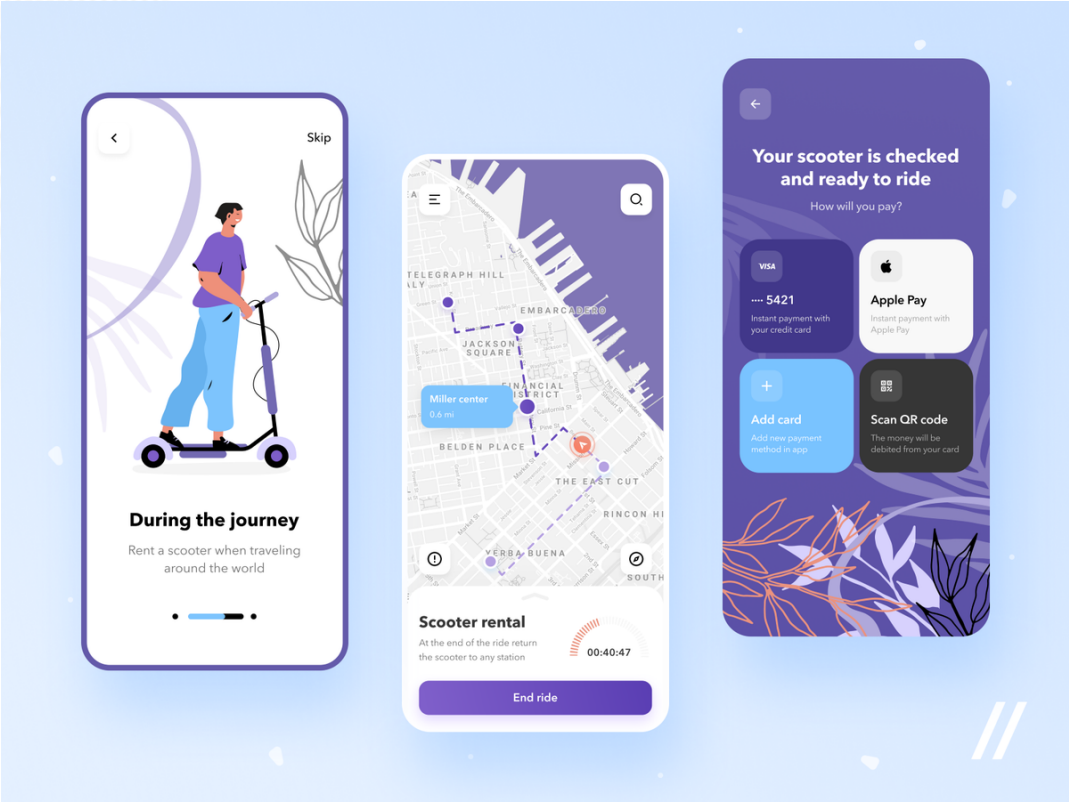
Learning Goals

- Understand challenges for QA of ML systems
- Be able to test assumptions about the data
- Detect changes in the model over time
- List quality attributes to consider in building ML systems
- Understand the importance of interpretability

S3D

Activity

- Develop one candidate AI feature of the scooter app (e.g., surge prediction)
- Come up with (at least 3) test plans.



What does it mean to do QA for a ML System?

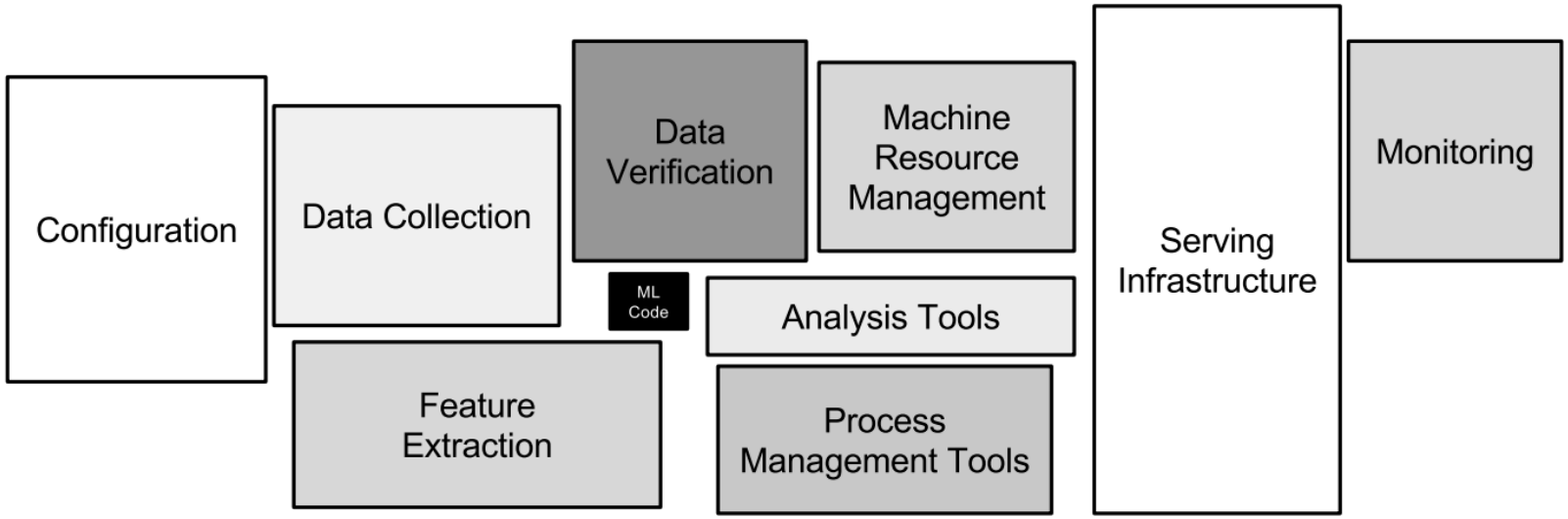
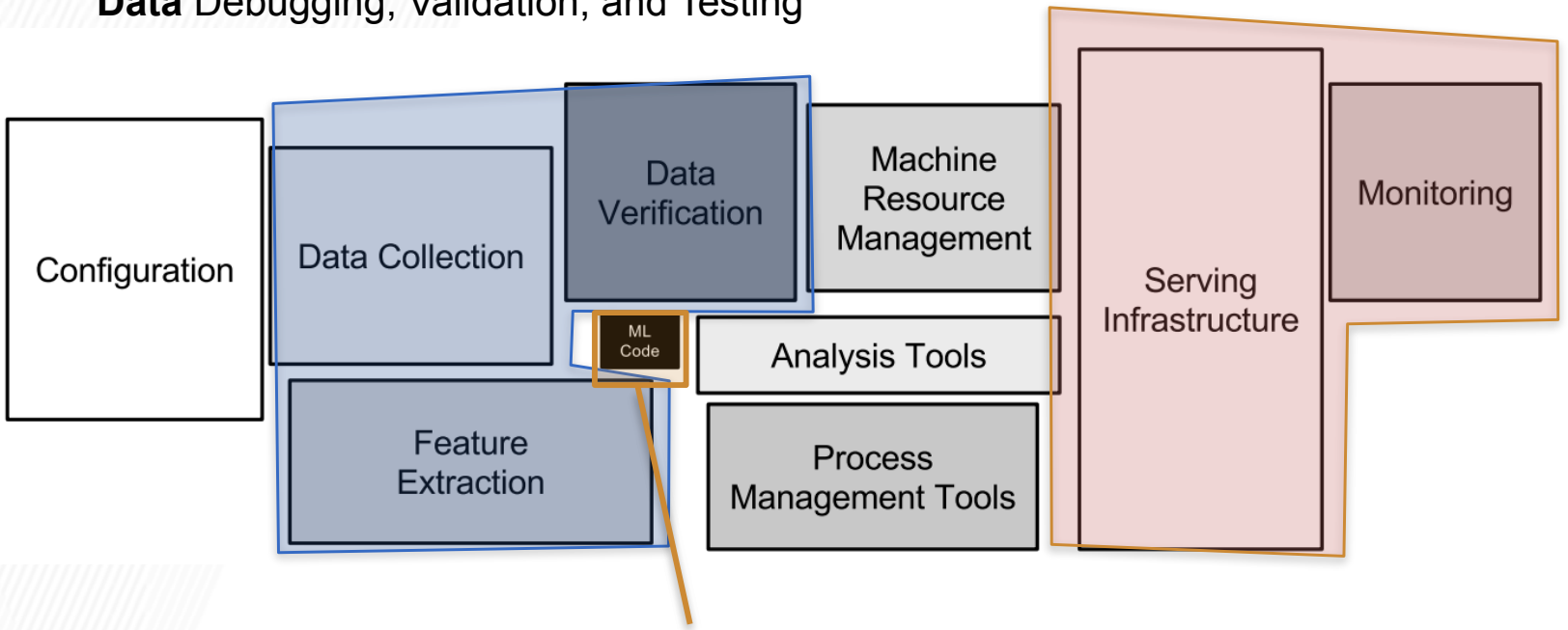


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

What does it mean to do QA for a ML System?

Service Debugging, Validation, and Testing

Data Debugging, Validation, and Testing



Model Debugging, Validation, and Testing

S3D

Data Debugging

- Data Collection: Validate Input Data Using a Data Schema
  - For your feature data, understand the range and distribution. For categorical features, understand the set of possible values.
  - Encode your understanding into rules defined in the schema.
  - Test your data against the data schema.

S3D

QA for Data

- Data Verification:
  - All numeric features are scaled, for example, between 0 and 1.
  - One-hot encoded vectors only contain a single 1 and N-1 zeroes.
  - Missing data is replaced by mean or default values.
  - Data distributions after transformation conform to expectations.
  - Outliers are handled, such as by scaling or clipping.
- Feature Extraction:
  - Are any features in your model redundant or unnecessary?

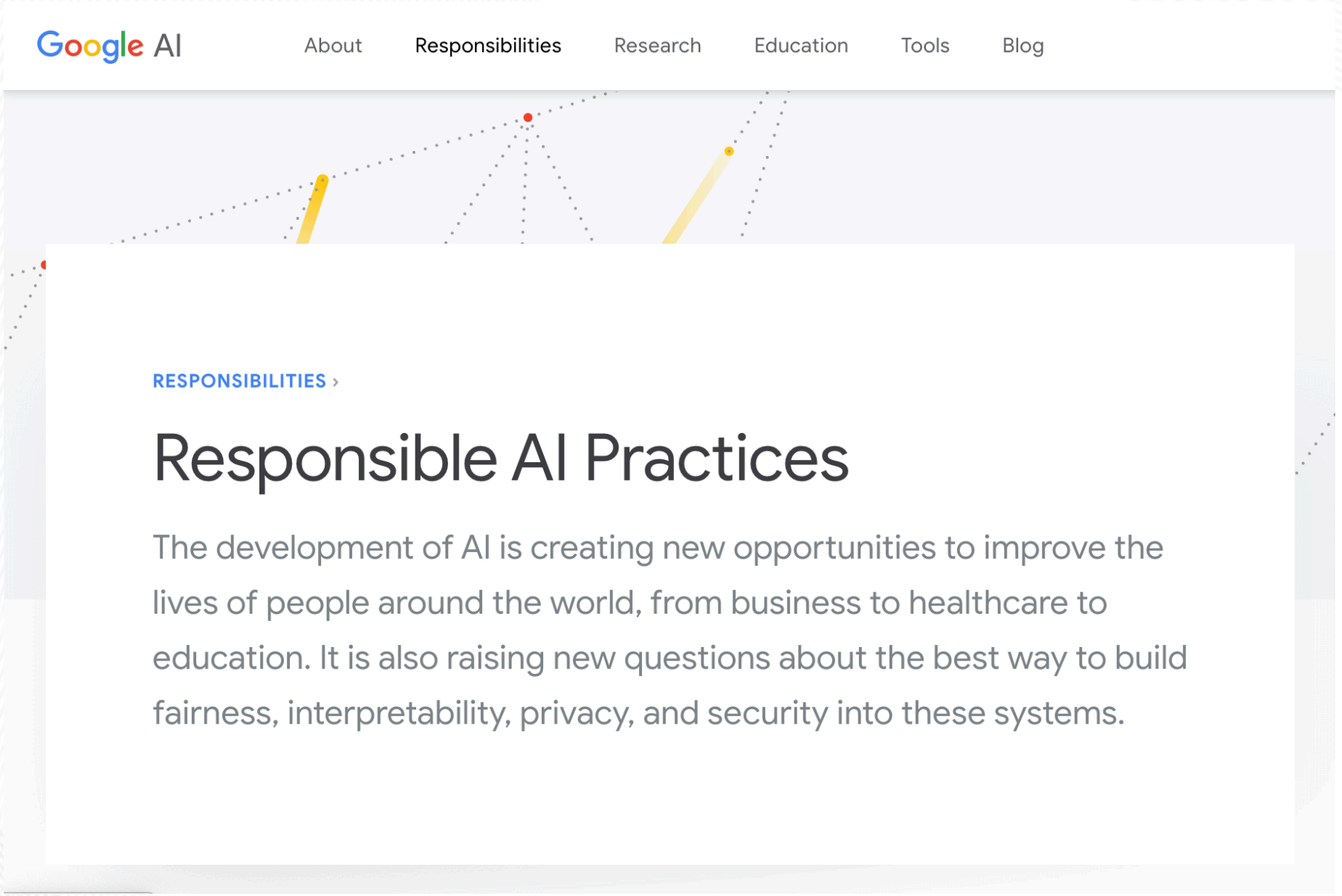
S3D

QA for ML Model

- Check that the data can predict the labels.
  - Use some examples from your dataset that the model can easily learn from. Alternatively, use synthetic data.
- Establish a baseline
  - Use a linear model trained solely on most predictive feature
  - In classification, always predict the most common label
  - In regression, always predict the mean value

S3D

https://ai.google/responsibilities/responsible-ai-practices/



S3D

Test, Test, Test

- Conduct rigorous **unit tests** to test each component of the system in isolation.
- Conduct **integration tests** to understand how individual ML components interact with other parts of the overall system.
- Proactively detect **input drift** by testing the statistics of the inputs to the AI system to make sure they are not changing in unexpected ways.

S3D

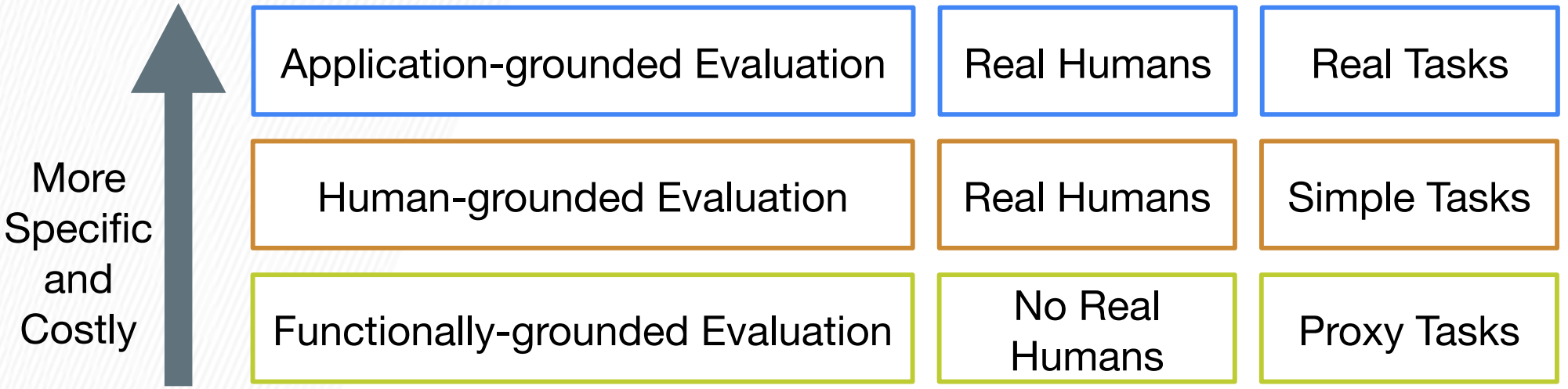
Test, Test, Test

- Use a gold standard dataset to test the system and ensure that it **continues to behave as expected**. Update this test set regularly in line with changing users and use cases, and to reduce the likelihood of training on the test set.
- Conduct iterative user testing to incorporate a diverse set of users' needs in the development cycles.
- Apply the quality engineering principle of poka-yoke: build quality checks into a system, so that unintended failures either cannot happen or **trigger an immediate response** (e.g., if an important feature is unexpectedly missing, the AI system won't output a prediction).

S3D

Adapted from Interpretability and Explainability in Machine Learning
Lakkaraju, Himabindu (Harvard University)

Test, Test, Test



Identify multiple metrics to assess training and monitoring

- Does your data contain any mistakes (e.g., missing values, incorrect labels)?
- Is your data sampled in a way that represents your users, and the real-world setting? Is the data accurate?
- Training-serving skew—the difference between performance during training and performance during serving—is a persistent challenge.
- Are any features in your model redundant or unnecessary? Use the simplest model that meets your performance goals.
- Data bias is another important consideration; learn more in practices on AI and fairness.

S3D

Understand the limitations of your dataset and model

- A model trained to detect correlations should not be used to make causal inferences, or imply that it can.
- It is important to communicate the scope and coverage of the training, hence clarifying the capability and limitations of the models.

Continue to monitor and update the system after deployment

- Issues will occur: any model of the world is imperfect almost by definition.
- Consider both short- and long-term solutions to issues.
- Before updating a deployed model, analyze how the candidate and deployed models differ, and how the update will affect the overall system quality and user experience.

S3D

Use a human-centered design approach

- Design features with appropriate disclosures built-in
- Consider augmentation and assistance
- Model potential adverse feedback early in the design process, followed by specific live testing and iteration for a small fraction of traffic before full deployment.
- Engage with a diverse set of users and use-case scenarios, and incorporate feedback before and throughout project development.

S3D

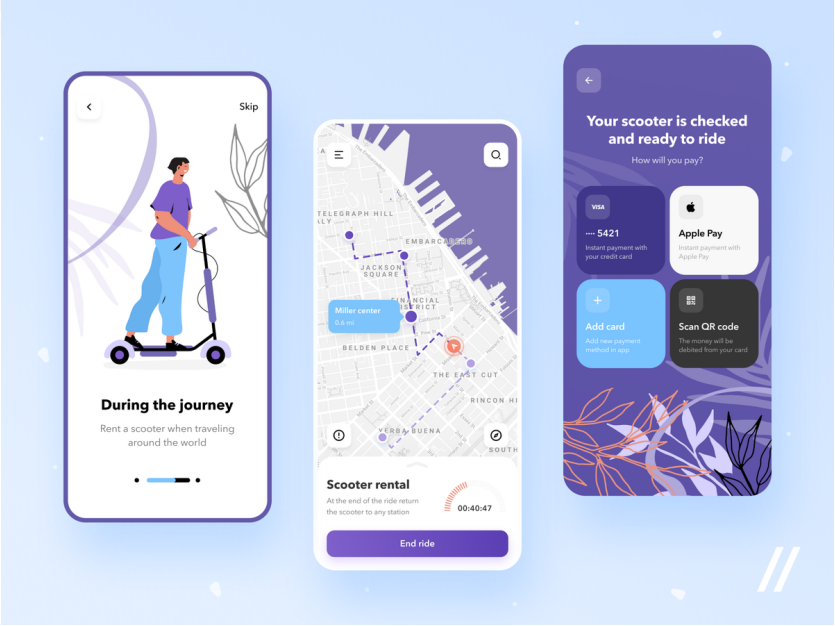
Software qualities of ML systems

What software qualities do we care about? (examples)

- Scalability
- Security
- Extensibility
- Documentation
- Performance
- Consistency
- Portability
- Installability
- Maintainability
- Functionality (e.g., data integrity)
- Availability
- Ease of use

Activity

- What are the quality attributes do you need to consider when building the new AI feature?



https://ckaestne.medium.com/quality-drivers-in-architectures-for-ml-enabled-systems-836f21c44334

Quality attributes of ML models

- **Interpretability (Explainability)**
- **Fairness**
- Inference latency
- Inference throughput
- Scalability
- Model size
- Energy consumption
- Determinability
- Cost
- Robustness
- Privacy

S3D

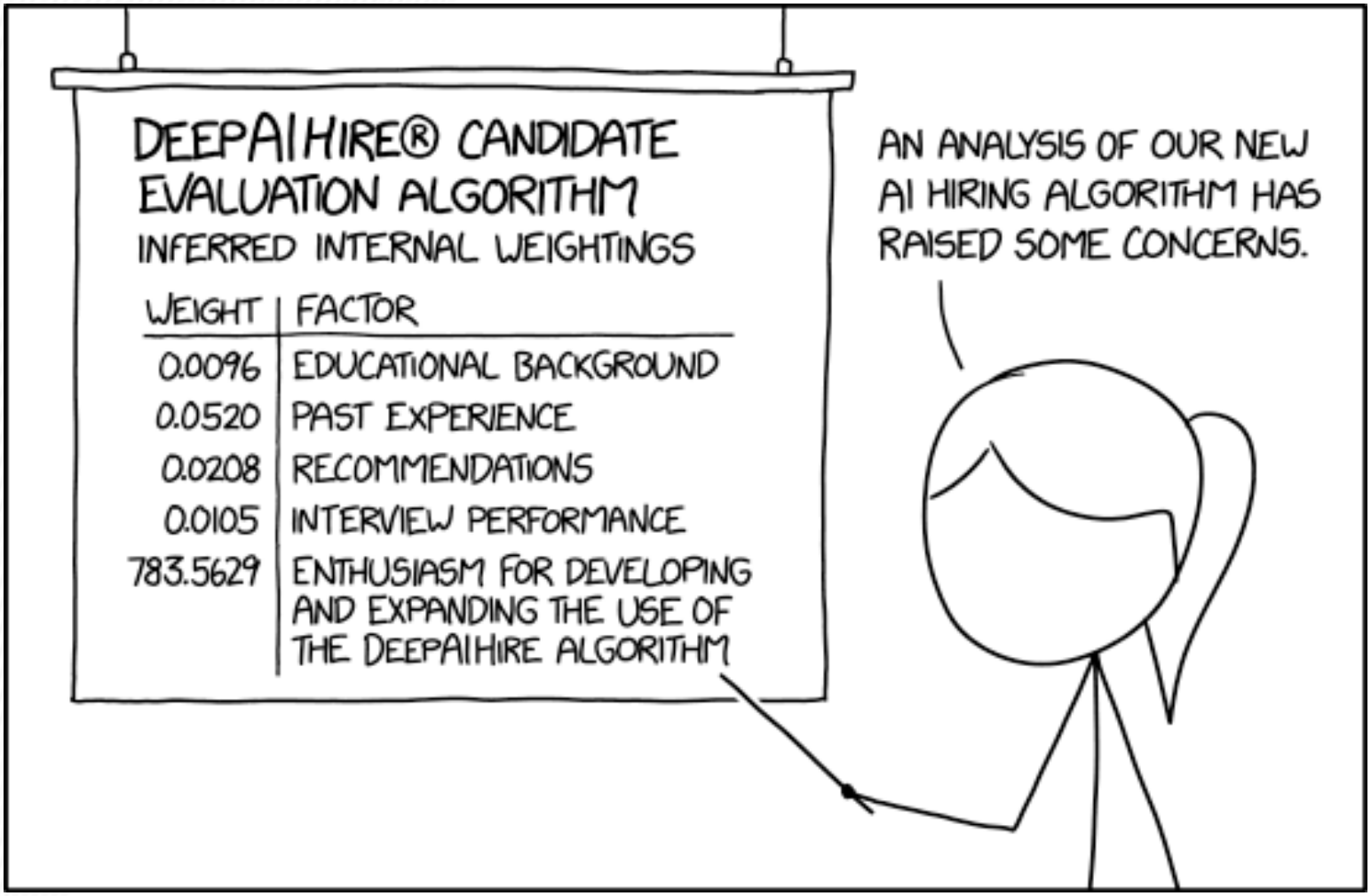
Interpretability

- ML systems are being deployed in complex high-stakes settings
- Safety, nondiscrimination … are often hard to quantify
- Fallback option: interpretability/explainability
    - If the system can explain its reasoning, we can verify if that reasoning is sound

S3D

https://xkcd.com/2237/

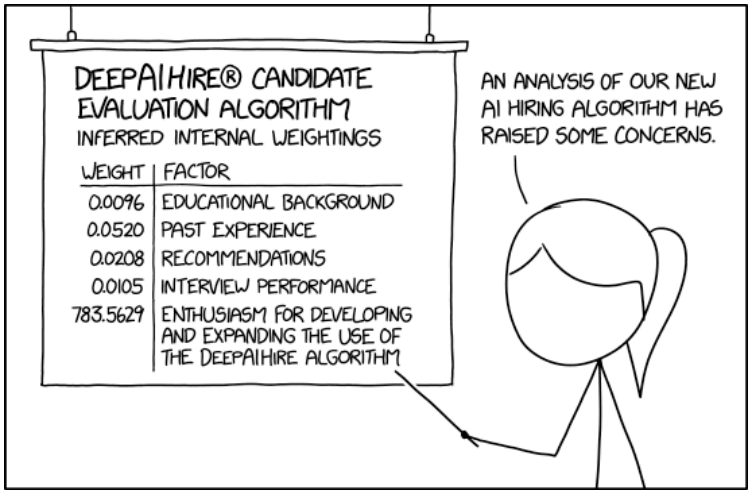
Interpretability



https://xkcd.com/2237/

Interpretability

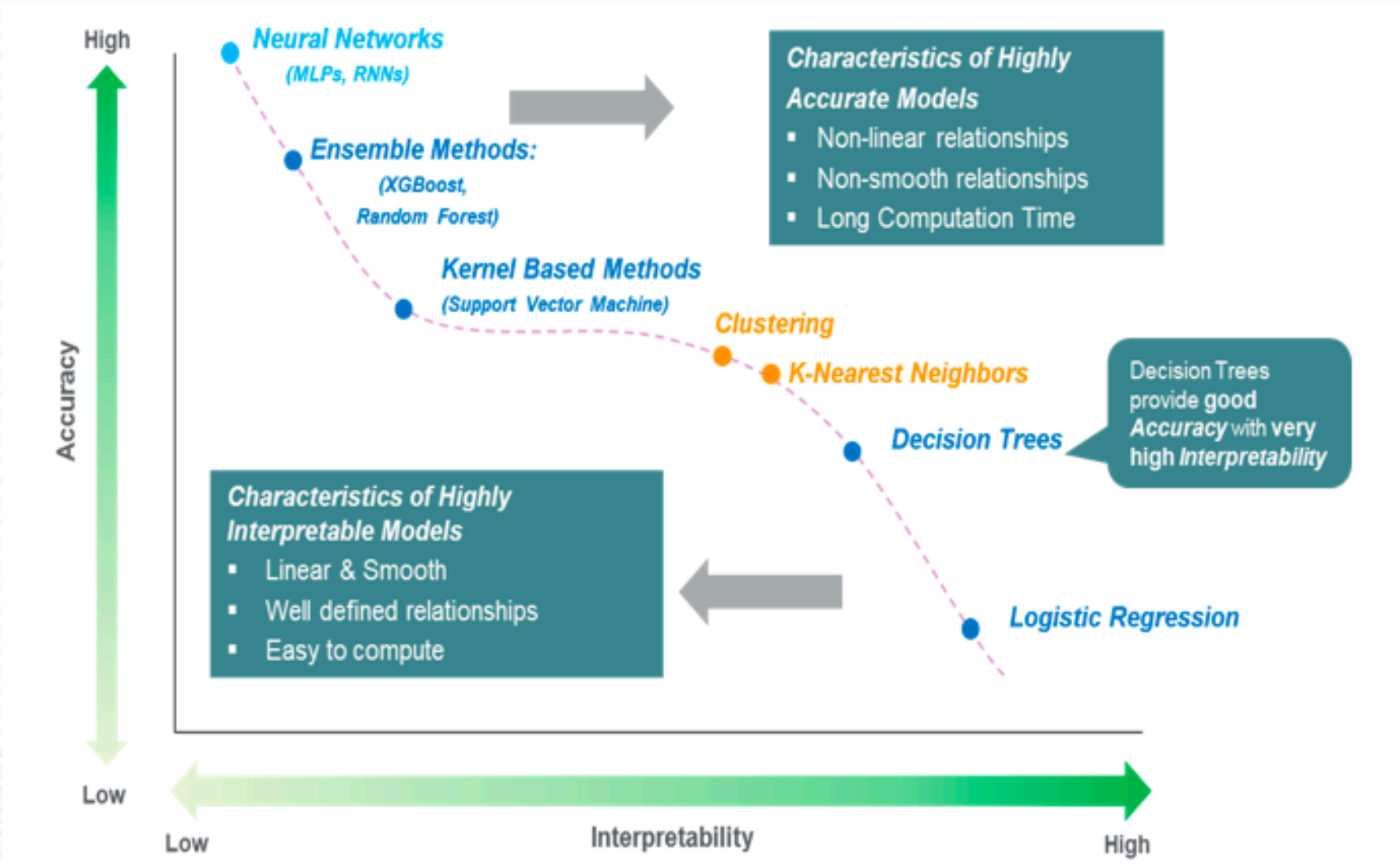
- Model debugging
- Auditing - fairness, safety, security
- Trust
- Actionable insights to improve outcomes
- Regulation



S3D

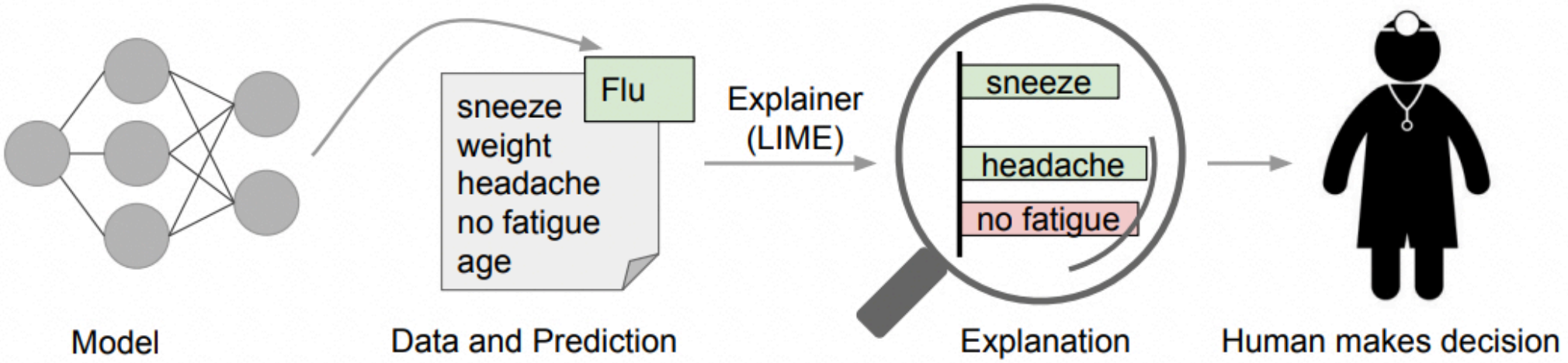
https://towardsdatascience.com/the-balance-accuracy-vs-interpretability-1b3861408062

Intrinsically interpretable models?



S3D

Explain models in a post-hoc manner



Post-hoc Explanation Techniques

- Typically consider the **complex model as a black box**
  - No internal details of the complex model required, only query access
- Several types of post hoc explanation techniques
  - Local vs. Global approximations, Gradient based vs. perturbation based.
- Examples: LIME, SHAP, Anchor, MUSE, Gradient times Input, Integrated Gradients etc.

S3D

SHAP (SHapley Additive exPlanations)

Luca Pacioli

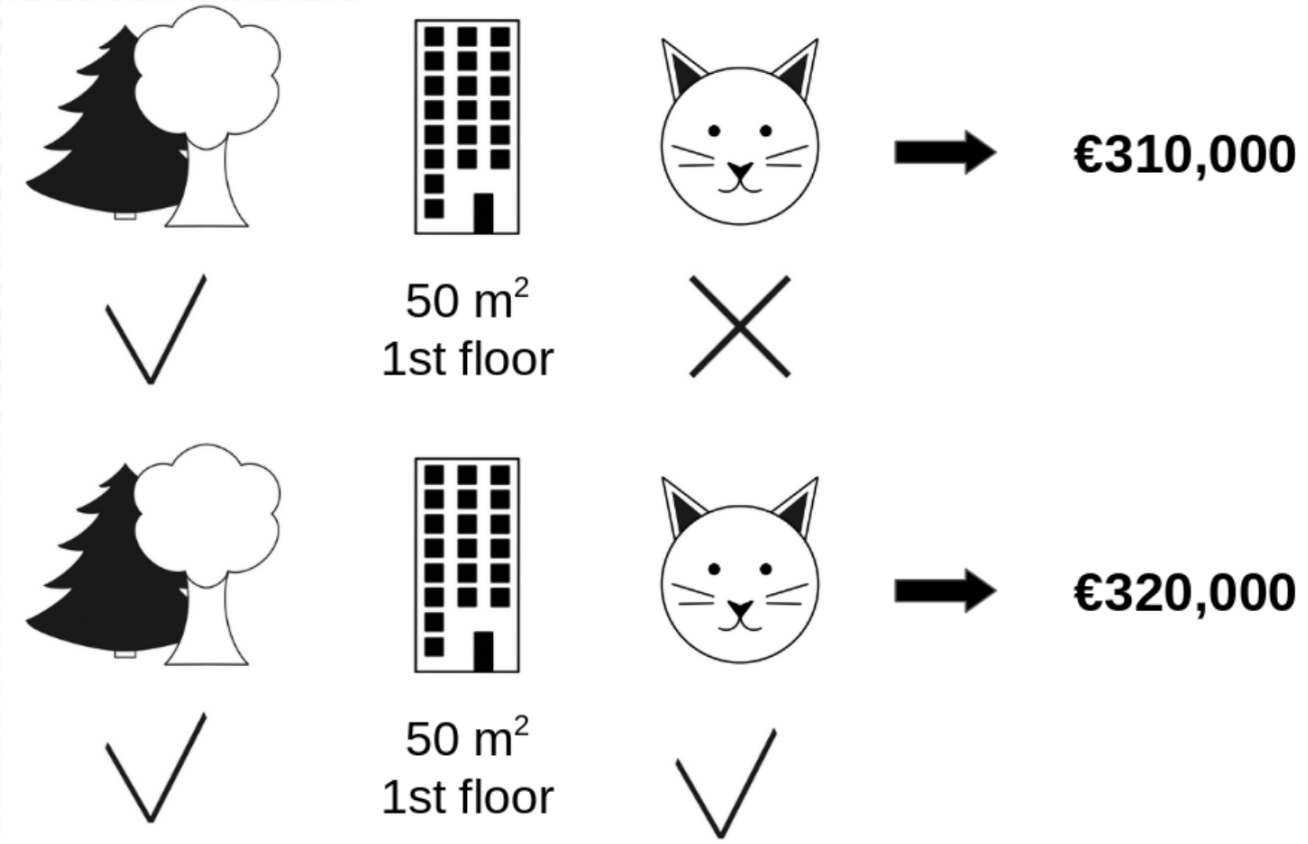
SHAP (SHapley Additive exPlanations)

Lloyd Shapley

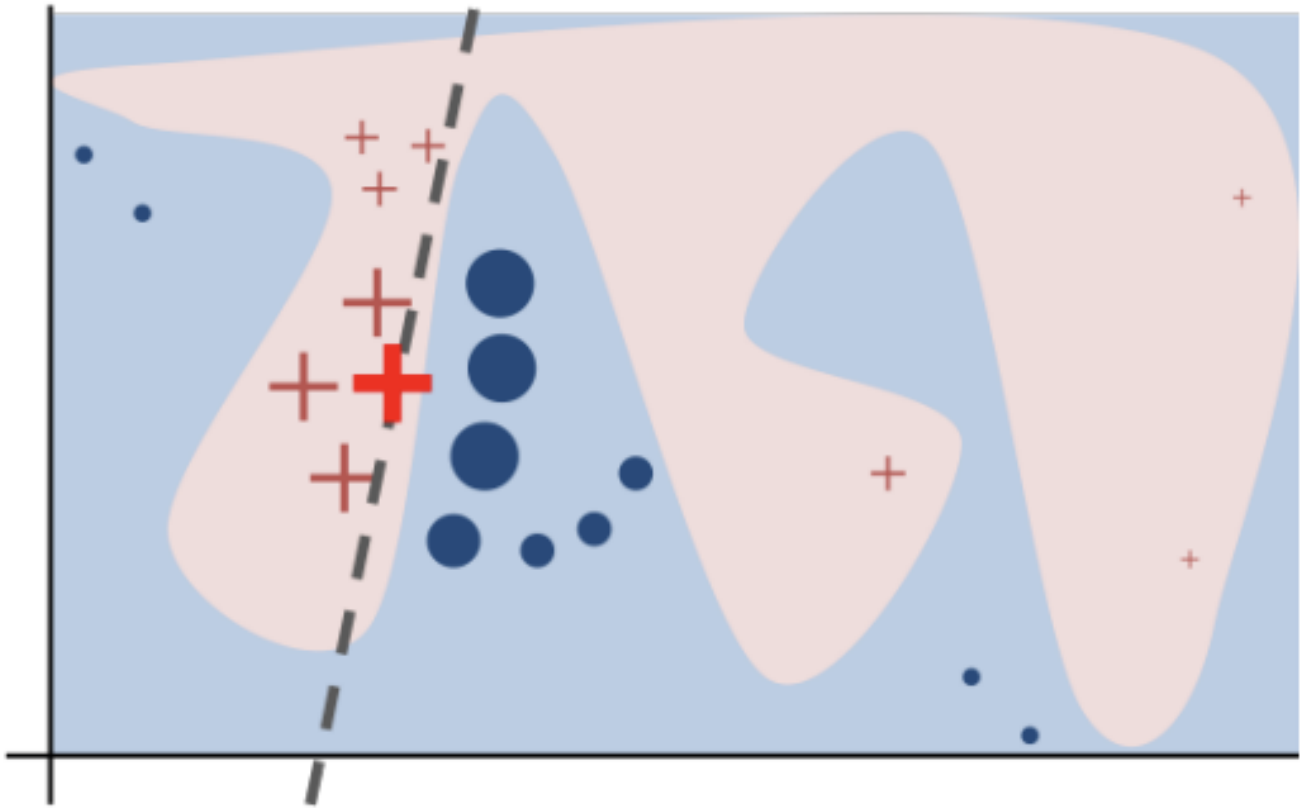
- The Shapley value:
  - assigns payouts to players depending on their contribution to the total payout
- → How much does each attribute contribute to the prediction?

S3D

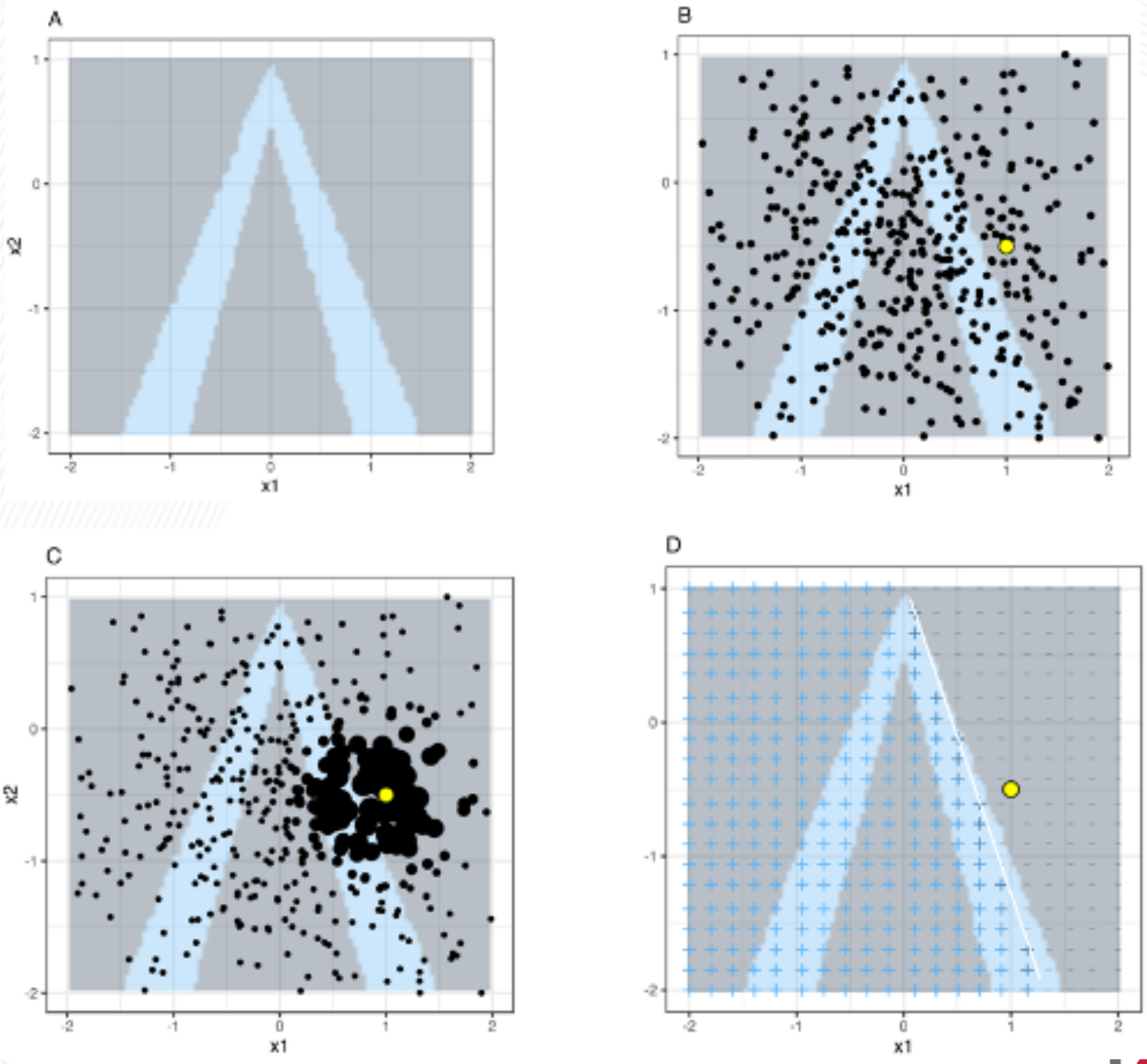
SHAP (SHapley Additive exPlanations)



LIME (Local Interpretable Model-agnostic Explanations)



While complex models have complex decision surfaces and are harder to explain globally, they may have simpler decision boundaries in local neighborhoods


A
B
C
D
x1
x2
1
0
-1
-2
-2
-1
0
1
2

Activity

- Inclass demo from: https://www.kaggle.com/vikumsw/explaining-random-forest-model-with-lime/notebook
- Working with your team, start trying to get LIME to run with your actual code (this will get you started on HW5)

S3D