Motivation
- With streaming data coming at a fast pace, it becomes critical to build systems that can process it efficiently and make real-time decisions.
- As data changes over time, a model learned in the past may be obsolete, need to continuously learn.

Traditional Pipeline

- Limitations:
  - Train a single model (e.g. Logistic Regression).
  - Difficult to do hyper-parameter tuning for model selection.
  - Might predict with a suboptimal model.

Our Approach

- Advantages:
  - Train multiple models in parallel (e.g. Logistic Regression, Trees, Neural Networks, etc.).
  - Automatic model selection in real-time (the deployer chooses the current best model to predict).
  - Always predict with the best model so far.

Design and Implementation
- Shared-nothing Architecture. RPC-based communication with Apache Thrift.
- Push (emit) instead of pull (getNext).
- Model check-pointing (PostgreSQL database).
- Registry service to register/unregister modules.
- Historical points in order to catch up Learners added on the fly.
- Holdout and Prequential evaluation.
- Everything configurable.
- Built entirely from scratch for this class.

Preliminary Results
- KDD Click-Through Rate dataset
  - ~2.4M examples, 100K sparse features.
- Prequential Evaluation

Conclusions
- Built a Streaming ML system that adapts to changes in data by automatically switching between models on real-time.
- Preliminary empirical results showed improvements over traditional streaming pipelines.

Future Work
- Changing feature dimensionality on the fly.
- Incremental generation of complex features (will require domain specific approximation algorithms).
- Test with more models (e.g. Neural Nets, Trees).
- Additional datasets (e.g. Criteo CRTR).
- Runtime results in a real distributed scenario (AWS).
- Different Deployer policies (e.g. voting).
- Learning from multiple streams.
- Real-time visualization.

Acknowledgments
Special thanks to Tianqi Chen and Rahul Kidambi for the helpful discussions about the project. Thanks to the CSE544 teaching staff for their guidance and support during the quarter.

1 https://github.com/nachocano/asml.git