Review: Applied ML Projects

Rayid Ghani and Kit Rodolfa

Carnegie Mellon University

ML
MACHINE LEARNING
DEPARTMENT

Heinz College
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Things to remember

This Week:

- Midterm – we will post tonight, due by Friday evening on Canvas
- No Wednesday or Thursday class sessions this week

Coming Up Next Week:

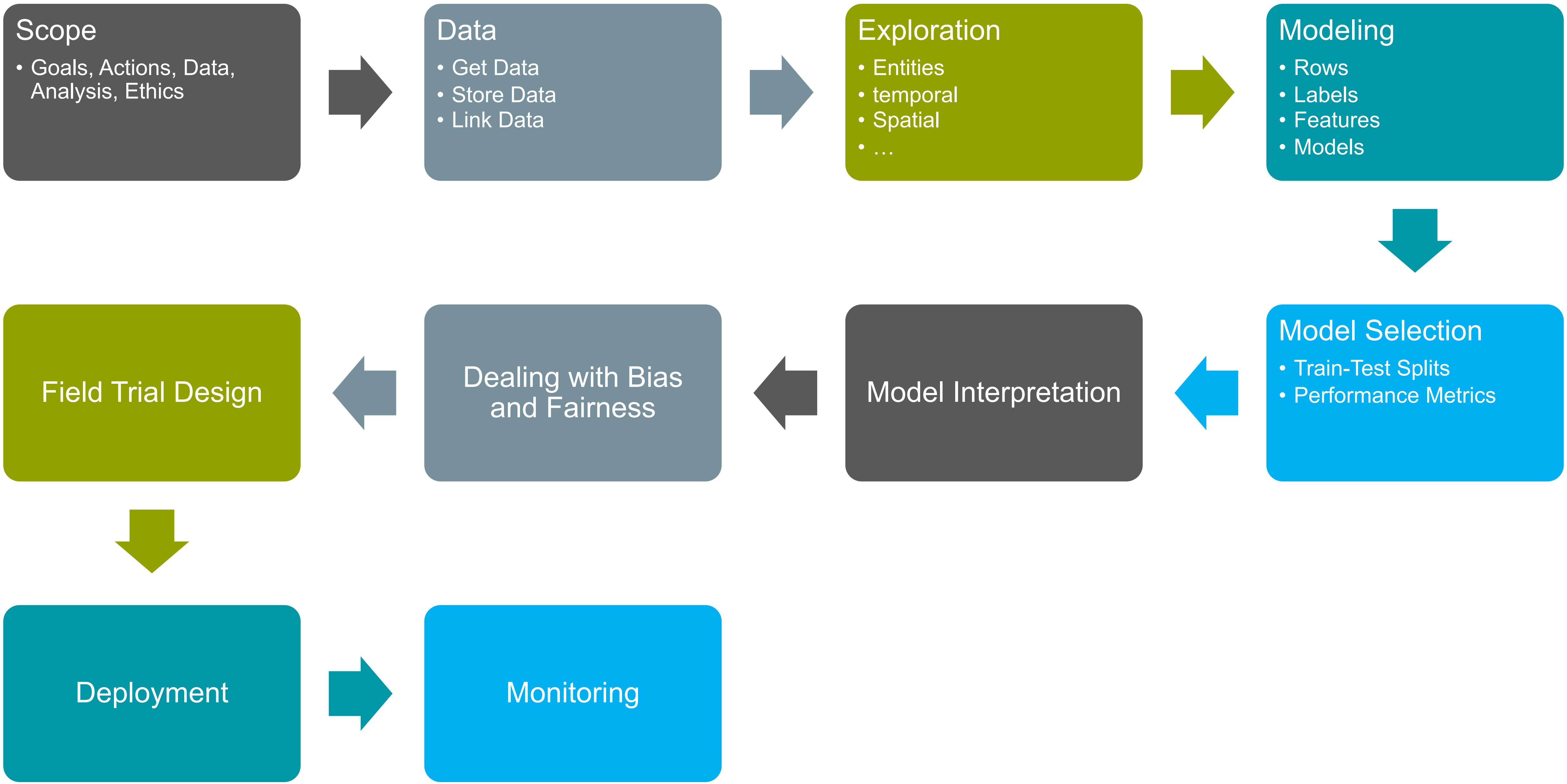
- Tuesday: Ethics Discussion
    - Note the change from interpretability overview
    - Be sure to do the readings, we'll spend time in class discussing the case study
- No Monday Update Assignment

Going Forward: Modules 2 and 3

- Last project update Assignment October 18 (week after next) with final models
- Module 2 & 3 classes focus on 2-3 methods/approaches each day
- Each group responsible for applying one approach from each class (may implement from scratch or use existing packages)
- “Extended Abstract” (3-4 pages) at end of the module comparing these results
- Presentations and “discussants” (15 minutes)
  - We will assign one of each, other methods up to you
  - Method overview and preliminary results

Recap: What we want you to learn from this class

- How to responsibly and effectively solve real-world problems using ML
  - Understand the *entire* Machine Learning process (and get hands-on experience doing most of it)
  - Build (and use) reusable ML pipelines
  - Learn how to formulate ML problems, use, understand, evaluate, and communicate ML methods (that you have covered in earlier classes) in the context of a real problem


Scope
• Goals, Actions, Data, Analysis, Ethics
Data
• Get Data
• Store Data
• Link Data
Exploration
• Entities
• temporal
• Spatial
• …
Modeling
• Rows
• Labels
• Features
• Models
Model Selection
• Train-Test Splits
• Performance Metrics
Model Interpretation
Dealing with Bias and Fairness
Field Trial Design
Deployment
Monitoring

Recap so far

- **Scope:** Goals, Actions, Data, Analysis, Ethics
- **Data:** Getting, storing, linking, exploring, and understanding
- **Formulation**: Rows, Labels, Time, Metric, Baselines
- **Pipeline**: Rows, Labels, Features, Train-Validation Pairs, Metrics, Models + hps
- **Model Selection:**
  - Run Experiments
  - Analyze results to choose best model
  - Iterate

Actionable and Goal-Driven Project Scope

1. **Goals**: Define the goal(s) of the project

2. **Actions**: What actions/interventions will you inform?

3. **Data**: What data do you have internally? What data do you need? What can you augment from external and public sources?

4. **Analysis**: What analysis needs to be done? How will it be validated?

Analytical Formulation Examples

How often is the recommendation/decision being made?

Who/what is included in the cohort?

What is the output?

What outcome are you predicting/estimating?

For what purpose?

On the first of every month, for all the individuals who have been released from Johnson County Jail during the past 2 years and have demonstrated mental health needs, can we identify the 200 highest risk individuals who are likely to return to jail in the next 6 months to prioritize for proactive mental health interventions?

Baseline Options

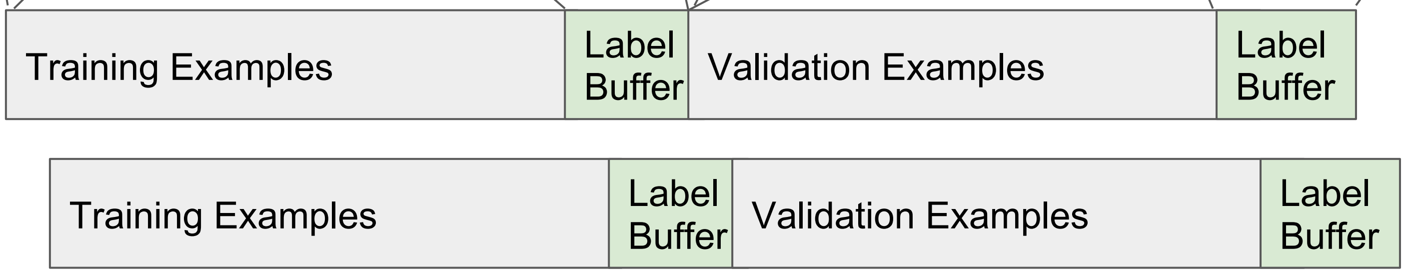
- Common Sense
- What they do today
- What they could do today easily (without any or very simple ML involved)
- Prior/Base Rate
  - What expected value would you get if you just choose at random (based on the data distribution)?

Train Validation Pairs

	Train Set				**Validation Set**			
Train-Validation Pair ID	Earliest project submission date	Latest project submission date	Start date for labels	End date for labels	Start date for rows	End date for rows	Start date for labels	End date for labels
1								
2								
3 (most recent)								

Train-Validation Pair 1

Train-Validation Pair 2

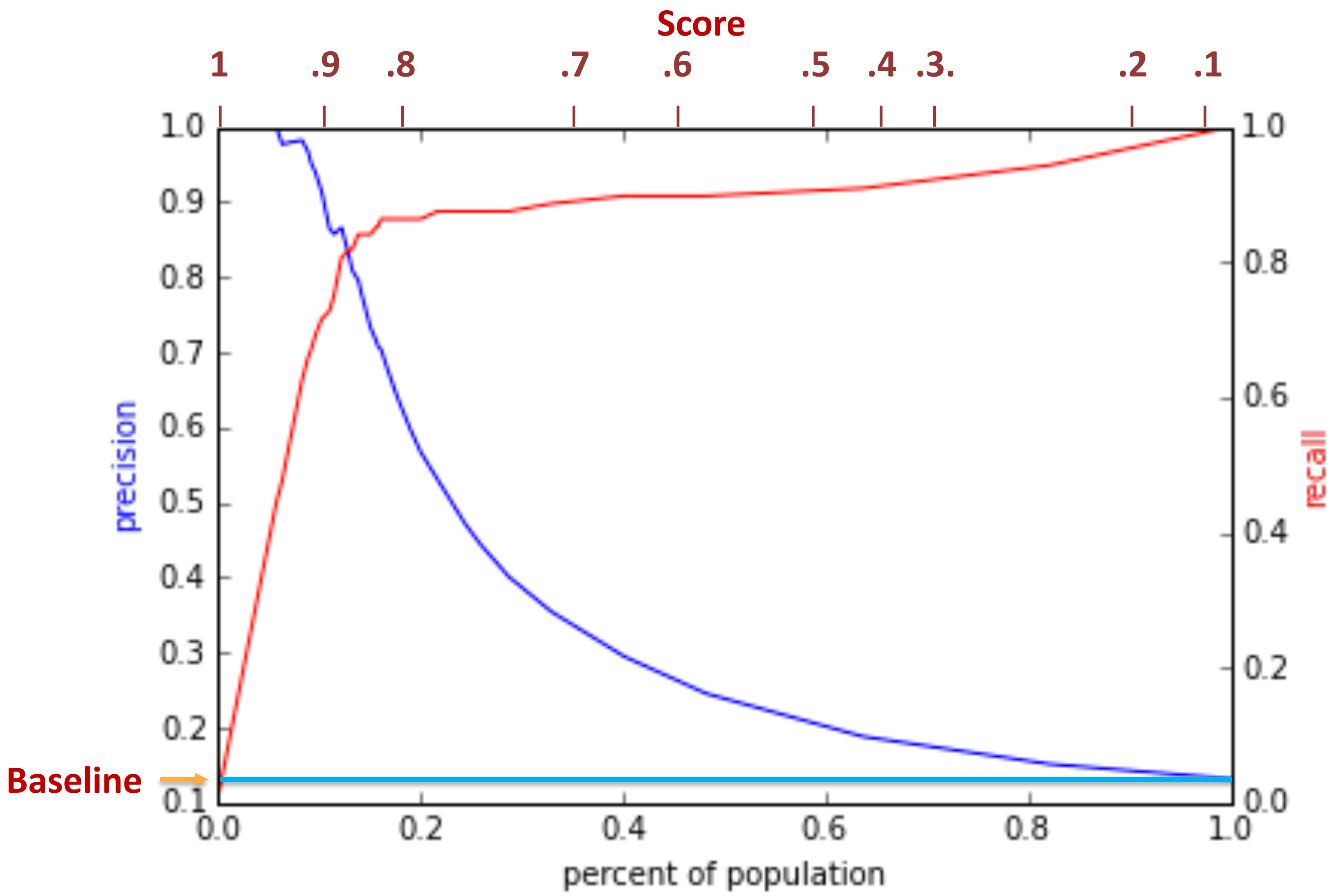


Confusion Matrix-based Metrics Cheatsheet

<table>
<tr><td></td><td></td><td colspan="2">True condition</td><td></td><td colspan="2"></td></tr>
<tr><td></td><td>Total population</td><td>Condition positive</td><td>Condition negative</td><td>Prevalence = $\frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$</td><td colspan="2">Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$</td></tr>
<tr><td rowspan="2">Predicted condition</td><td>Predicted condition positive</td><td>True positive, Power</td><td>False positive, Type I error</td><td>Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Predicted condition positive}}$</td><td colspan="2">False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Predicted condition positive}}$</td></tr>
<tr><td>Predicted condition negative</td><td>False negative, Type II error</td><td>True negative</td><td>False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Predicted condition negative}}$</td><td colspan="2">Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Predicted condition negative}}$</td></tr>
<tr><td colspan="2" rowspan="2"></td><td>True positive rate (TPR), Recall, Sensitivity, probability of detection = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$</td><td>False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$</td><td>Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$</td><td rowspan="2">Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$</td><td rowspan="2">$F_1$ score = $\frac{2}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}}$</td></tr>
<tr><td>False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$</td><td>Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$</td><td>Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$</td></tr>
</table>

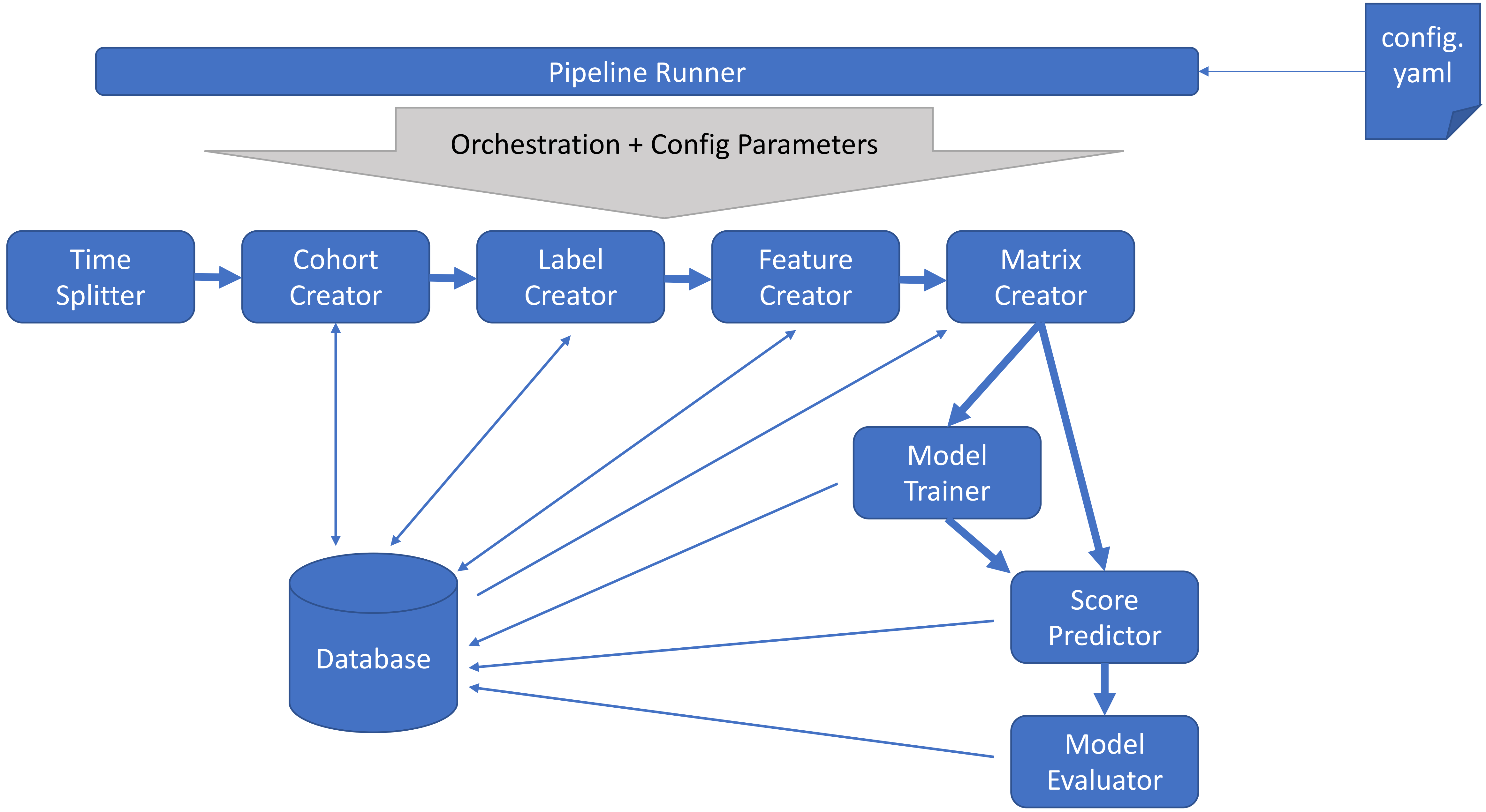
Source: https://en.wikipedia.org/wiki/Sensitivity_and_specificity

Varying the Threshold

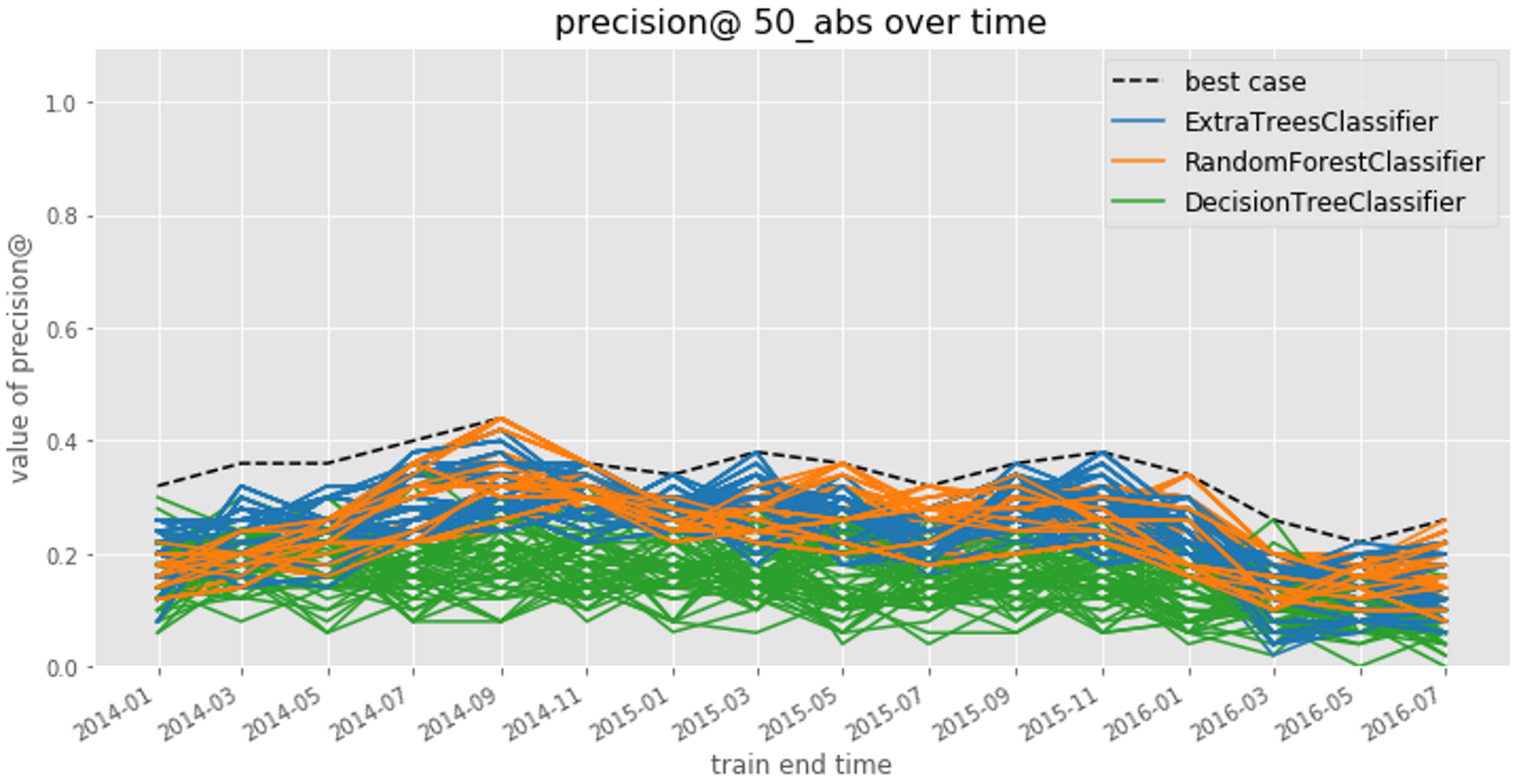


Feature Generation

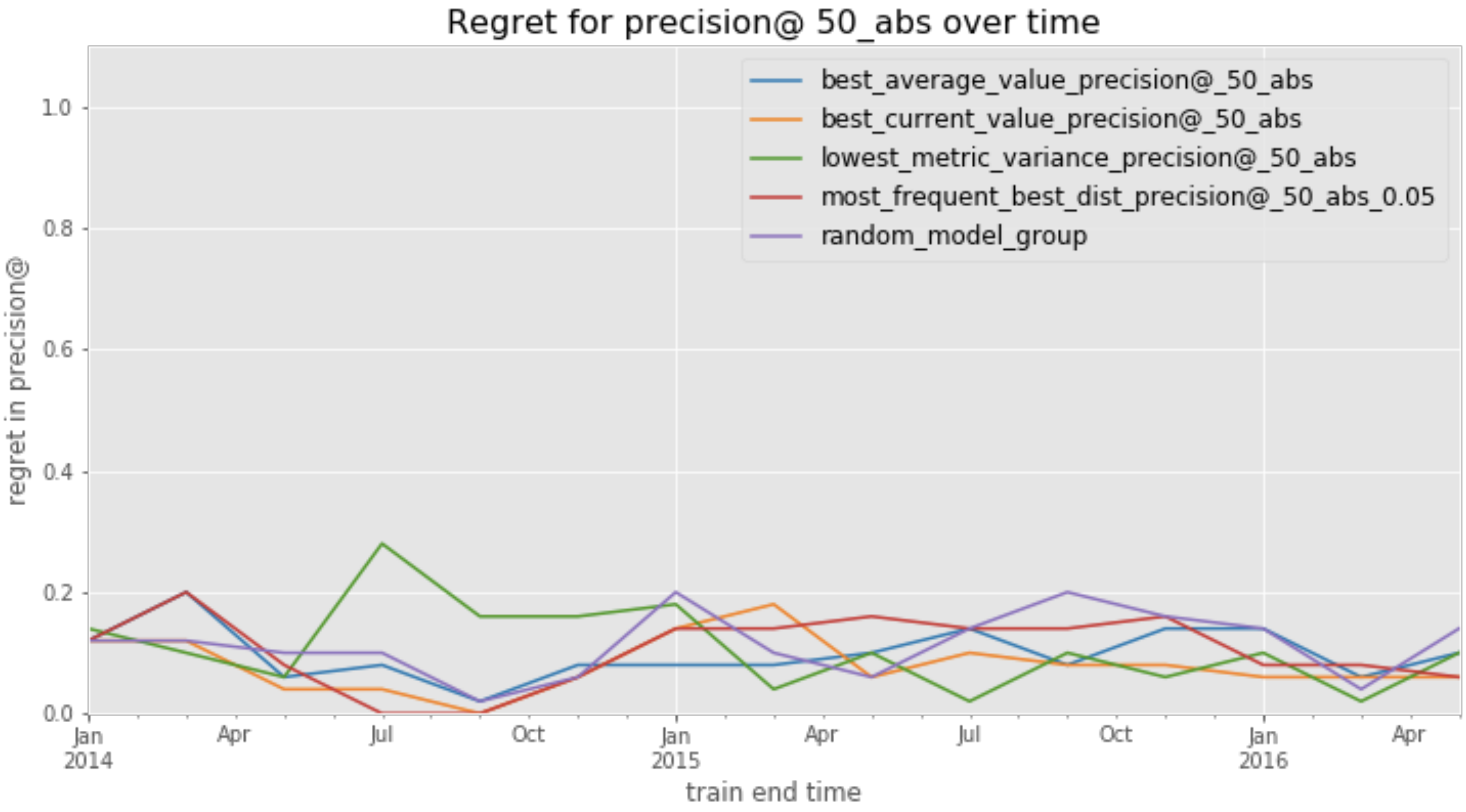
- Categorical to Binary (Dummies)
- Features for missing values
- Discretization
- Date/Time Features
- Scaling/Normalizing
- Transformations
- **Aggregations (space, time, space and time)**
- **Relative (compared to the average…)**
- Interactions


config.
yaml
Pipeline Runner
Orchestration + Config Parameters
Time
Splitter
Cohort
Creator
Label
Creator
Feature
Creator
Matrix
Creator
Model
Trainer
Score
Predictor
Model
Evaluator
Database

Model Selection



Model Selection



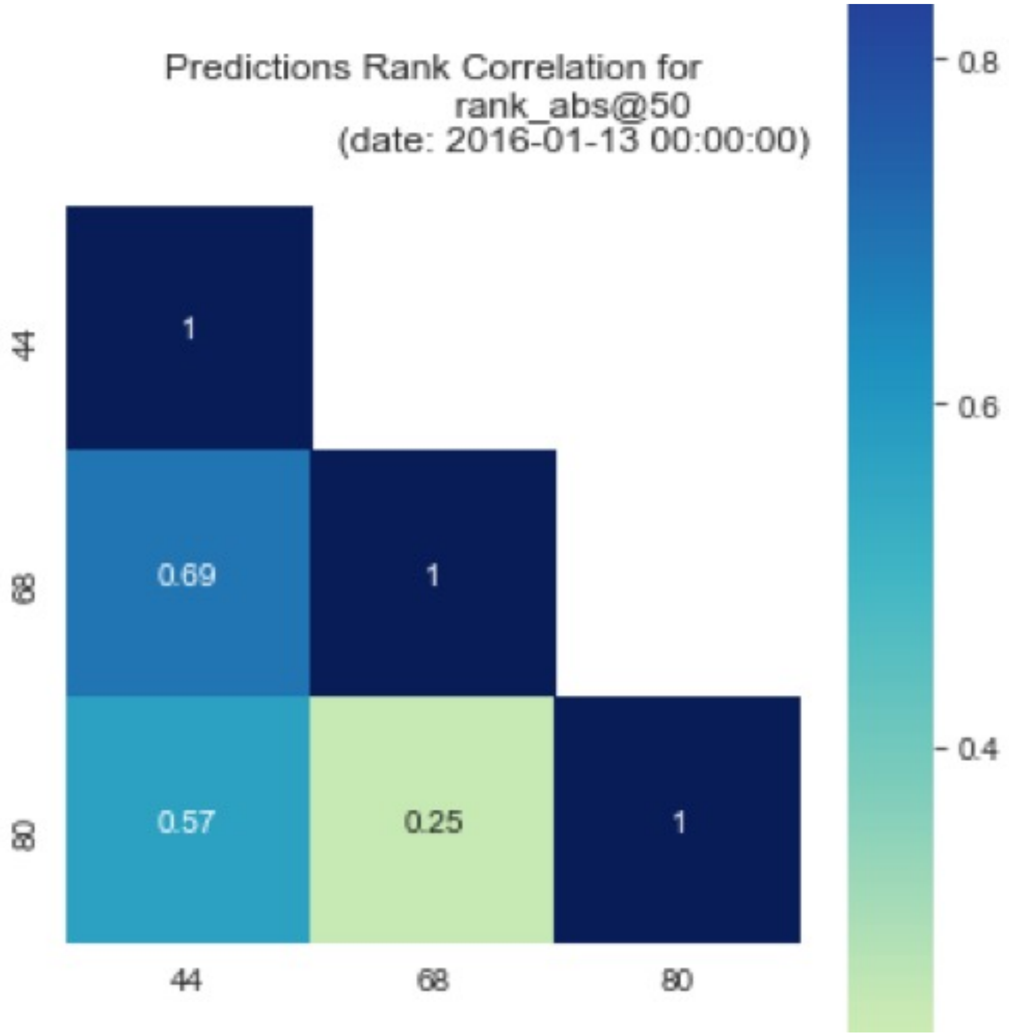
Finishing Up Model Selection?

Model Selection

- May not be obvious which strategy / model specification is "best"

- Among good candidates, may be instructive to ask how similar or different the lists each strategy would produce are

- May ultimately want to deploy (or at least test) a strategy that combines across several specifications



Some Open Research Questions

- What are the conditions under which temporal validation out-performs traditional cross-validation? By how much?
- Likewise, what can we learn about how well certain strategies perform in terms of regret under different real-world conditions?
- Many problems in policy settings involve resource constraints that require optimization at the top of the list, but few methods optimize for this directly.
    - e.g., Transductive Top k

Some Open Research Questions


Transductive Optimization of Top k Precision

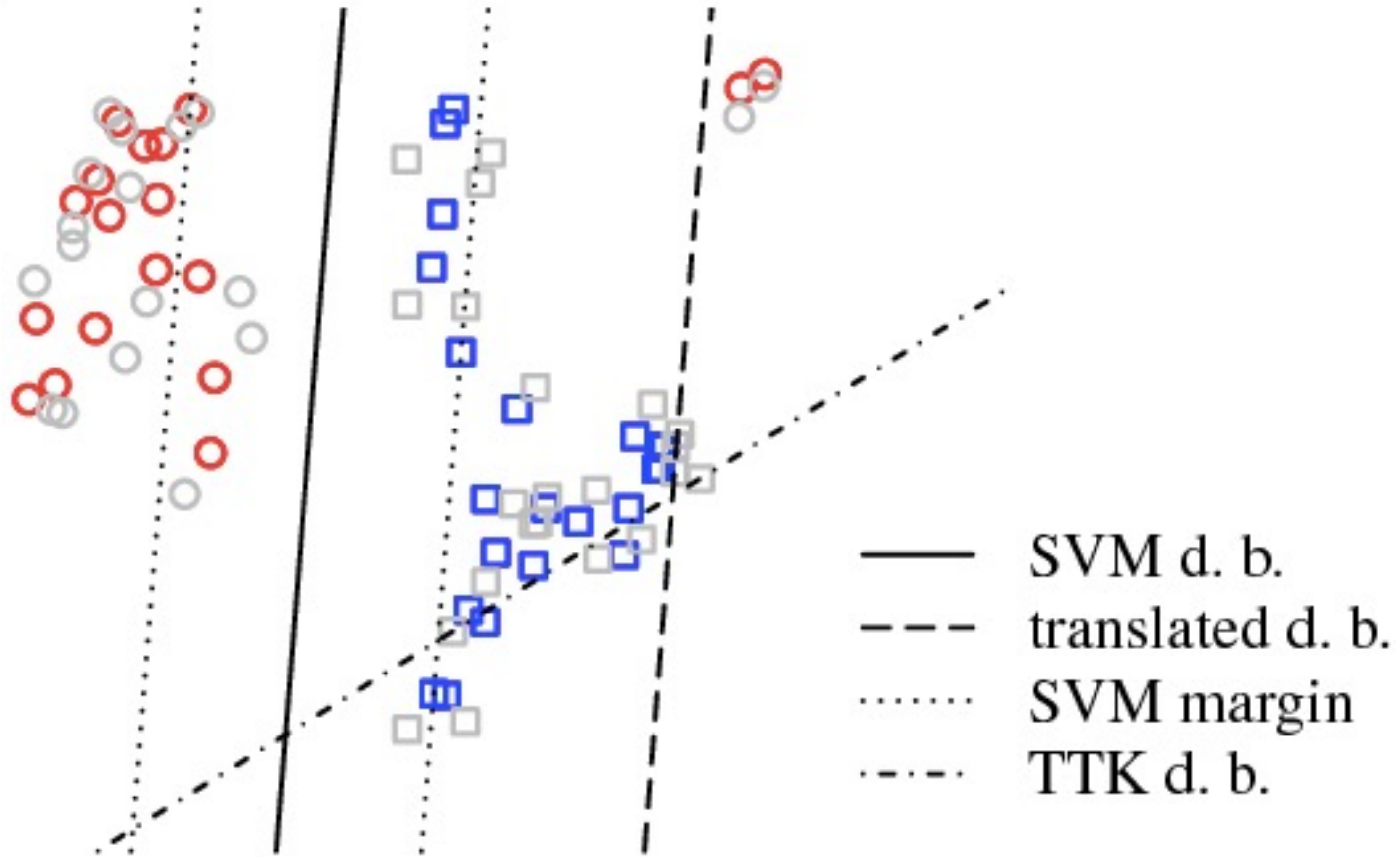
Li-Ping Liu Thomas G. Dietterich
EECS, Oregon State University
Corvallis, OR 97330, USA
{liuli@eecs.oregonstate.edu, tgd@oregonstate.edu }

Nan Li Zhi-Hua Zhou
Department of Computer Science & Technology, Nanjing University
Nanjing 210023, China
{lin, zhouzh}@lamda.nju.edu.cn

Some Open Research Questions

- The SVM loss function will find the “best” separating hyperplane overall, but perhaps we could draw a better hyperplane to separate just *k* positive examples?
- *Transductive* method: needs to be aware of the test set **without labels** to select just *k* test examples.
- Modified gradient descent procedure to project gradient direction for L2-regularized SVM loss onto a “feasible solution cone” such that <u>no more than</u> *k* test examples will be predicted positive after the step.

Some Open Research Questions



Some Open Research Questions

Paper shows improvements on synthetic examples and some “standard” datasets, but still more to investigate:

- Can be slow to converge on larger datasets
- “At most” *k* examples can yield many fewer than the desired *k*, particularly for rare events (why doesn’t the algorithm target *exactly k*?)
- Although creating a “top k” boundary, still penalizes false positives and false negatives equally during optimization
- Can we do better at the top, even if we don’t have access to the test list?

Things to remember

This Week:

- Midterm – we will post tonight, due by Friday evening on Canvas
- No Wednesday or Thursday class sessions this week

Coming Up Next Week:

- Tuesday: Ethics Discussion
  - Note the change from interpretability overview
  - Be sure to do the readings, we'll spend time in class discussing the case study
- No Monday Update Assignment