Team DDI

Mengqiao (MQ) Liu
Rachel King
Evan Sinar

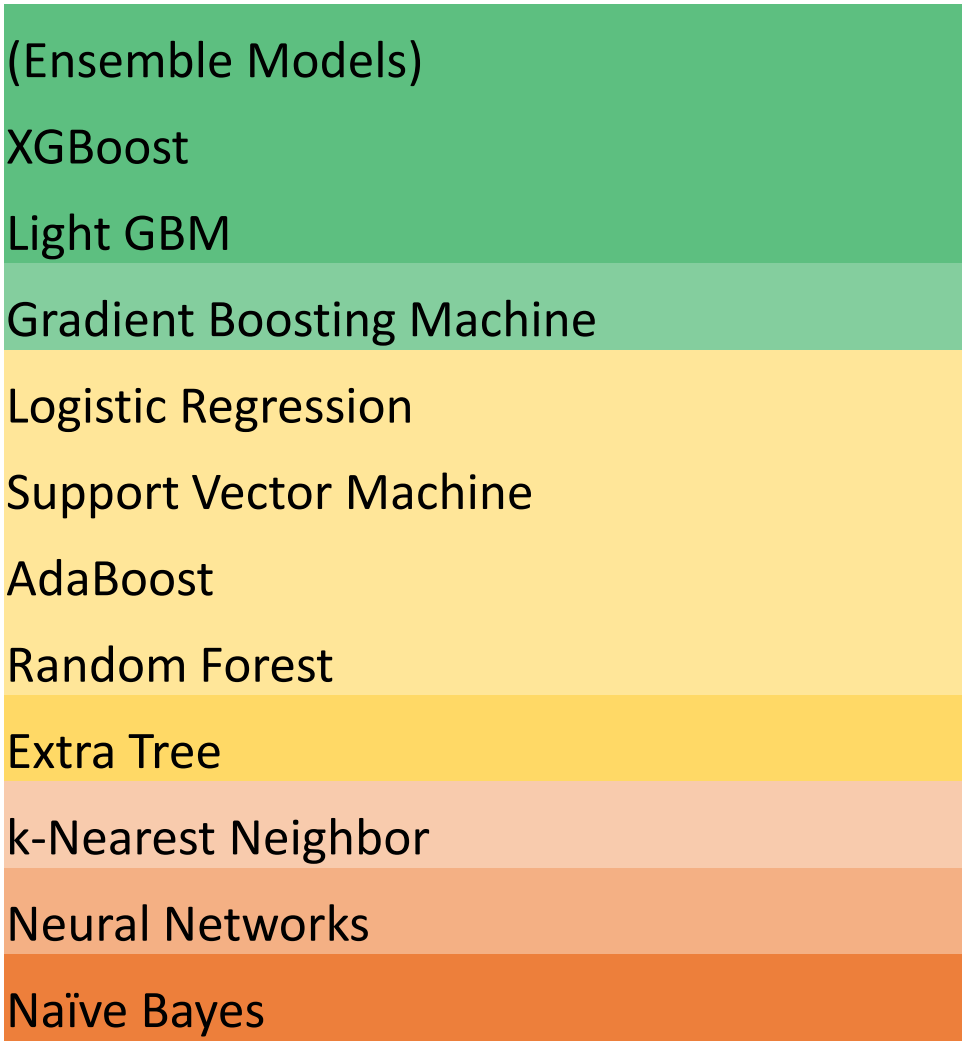
Problem

- Supervised learning
- Classification problem
- A few “strong” features and many weak features
- Redundant/interdependent features
- Sparse data
- A lot of noise
- Missing data
- …

Approach

- Bring in external data
- Data imputation
- Tried **many** models, with heavy focus on boosting models

Classifier Performance

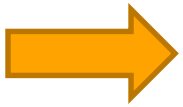


Our Winning Solution

Data Pre-Processing

- Categorical variables
  - Transformed to numeric variables (e.g., potential level)
  - Encoded to one column per category, with a 1 or 0 in each cell

EmployeeID	Country
1	US
2	US
3	Japan
4	China
5	Spain
6	China



EmployeeID	US	Japan	China	Spain
1	1	0	0	0
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	0	0	0	1
6	0	0	1	0

Data Pre-Processing

- Missing data
  - Replaced with column means
  - Tried the following methods; did not see sig. improvement in prediction:
    - Mean across time points
    - Imputation based on ML methods (e.g, KNN)
    - MICE imputation (Multiple Imputation by Chained Equations)

Feature Engineering & Selection

- Add features
  - External variables theoretically related to turnover
    - ✓ Country-level unemployment rate
    - ✓ Consumer Confidence Index
    - ✓ Composite Leading Indicator
  - Combinations of original features (the most predictive ones)
    - ✓ Small increment in prediction but less variance

- Remove features
  - Low variance
  - Supervisor ID and City
  - Machine learning based methods (Lasso)

Model

- Python 3.5

- XGBoost
  - Distributed gradient boosting system that is highly *efficient, accurate, and flexible*
  - Tianqi Chen and Carlos Guestrin, UW

Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In 22nd SIGKDD Conference on Knowledge Discovery and Data Mining, 2016

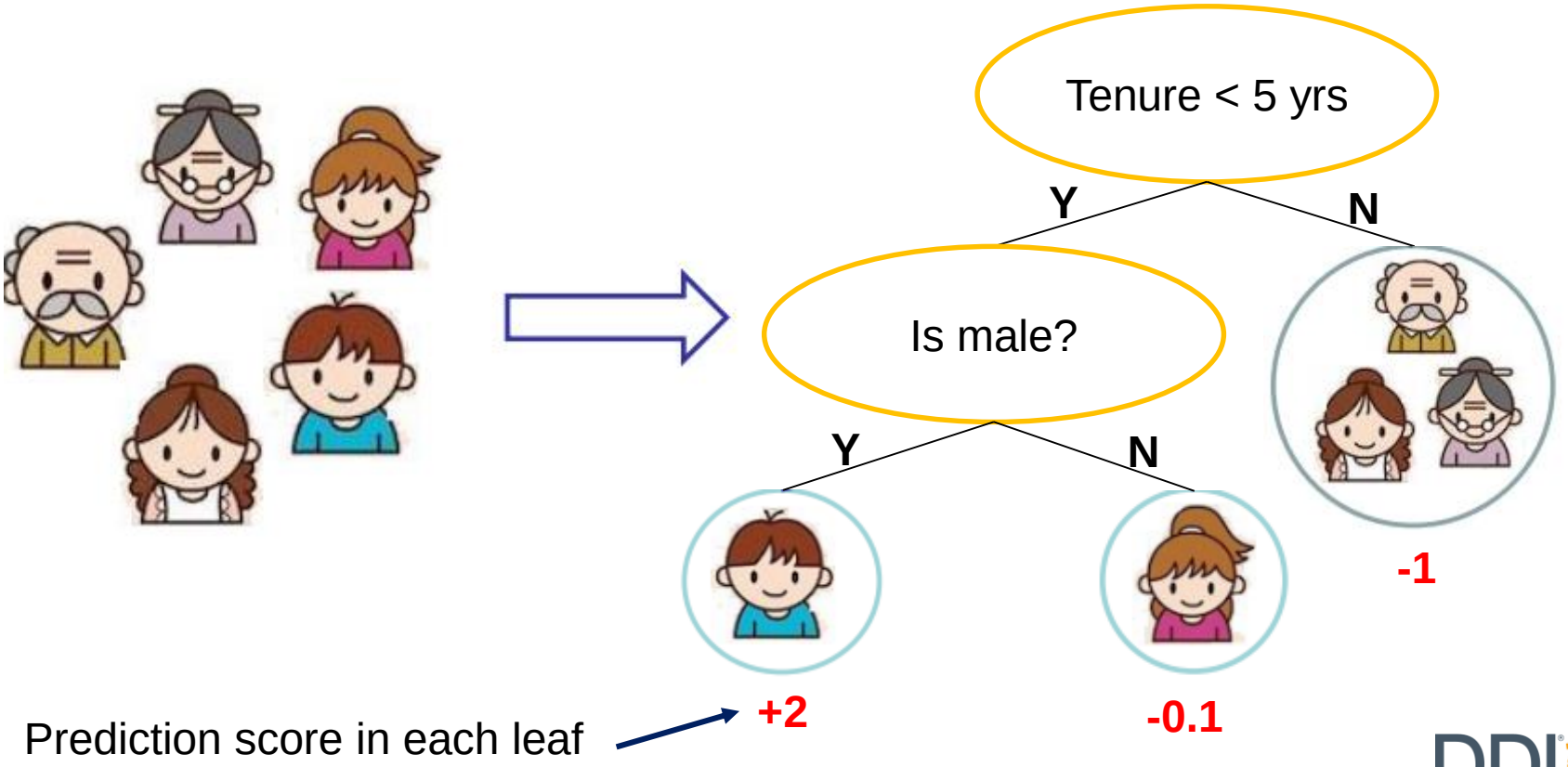
XGBoost

* Part of this content is based on these slides by the author of XGBoost

- Based on classification trees

Input: tenure, gender, income

Does the person leave

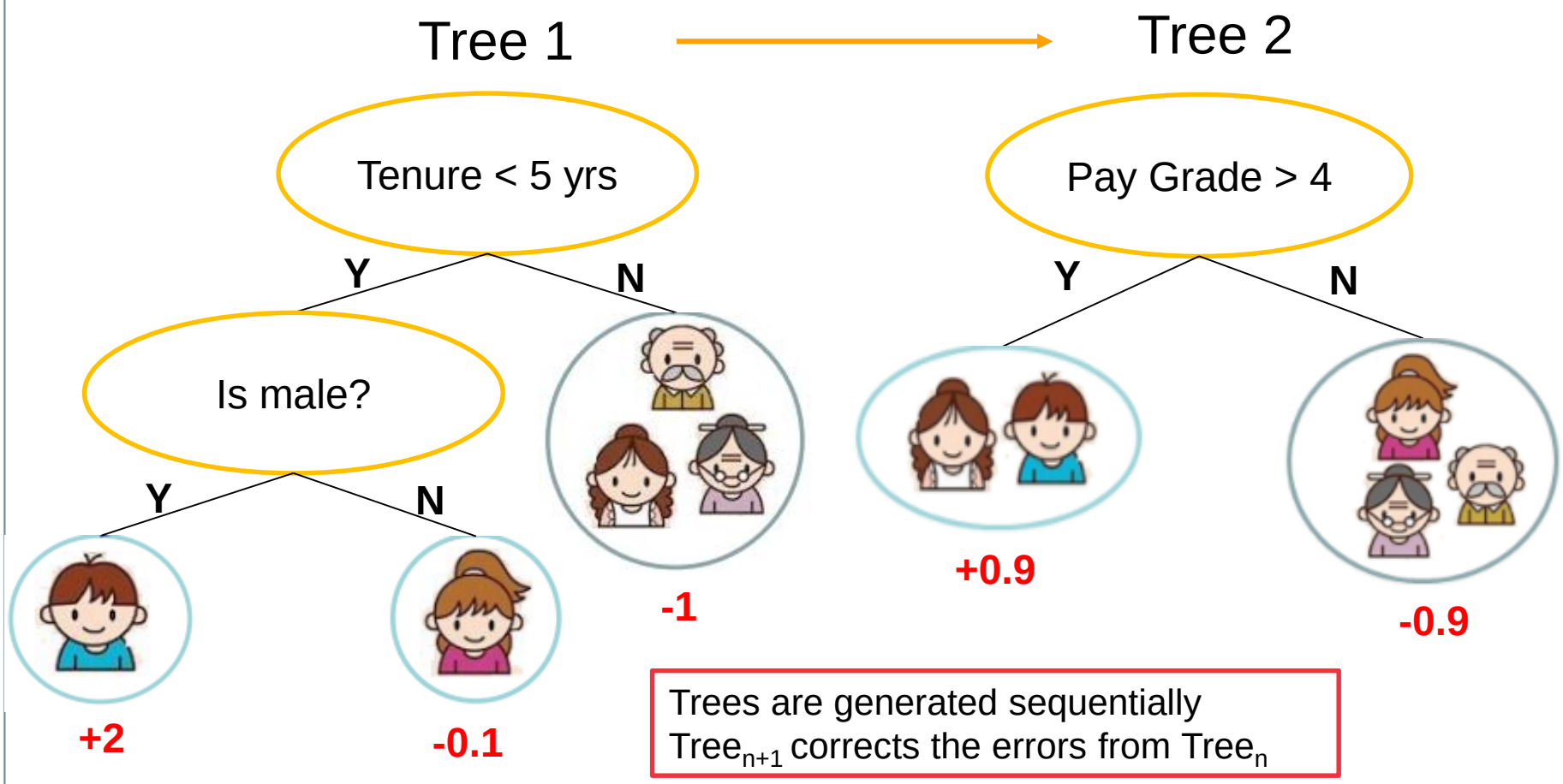


Prediction score in each leaf

XGBoost

* Part of this content is based on these slides by the author of XGBoost

- Boosting: an iterative process

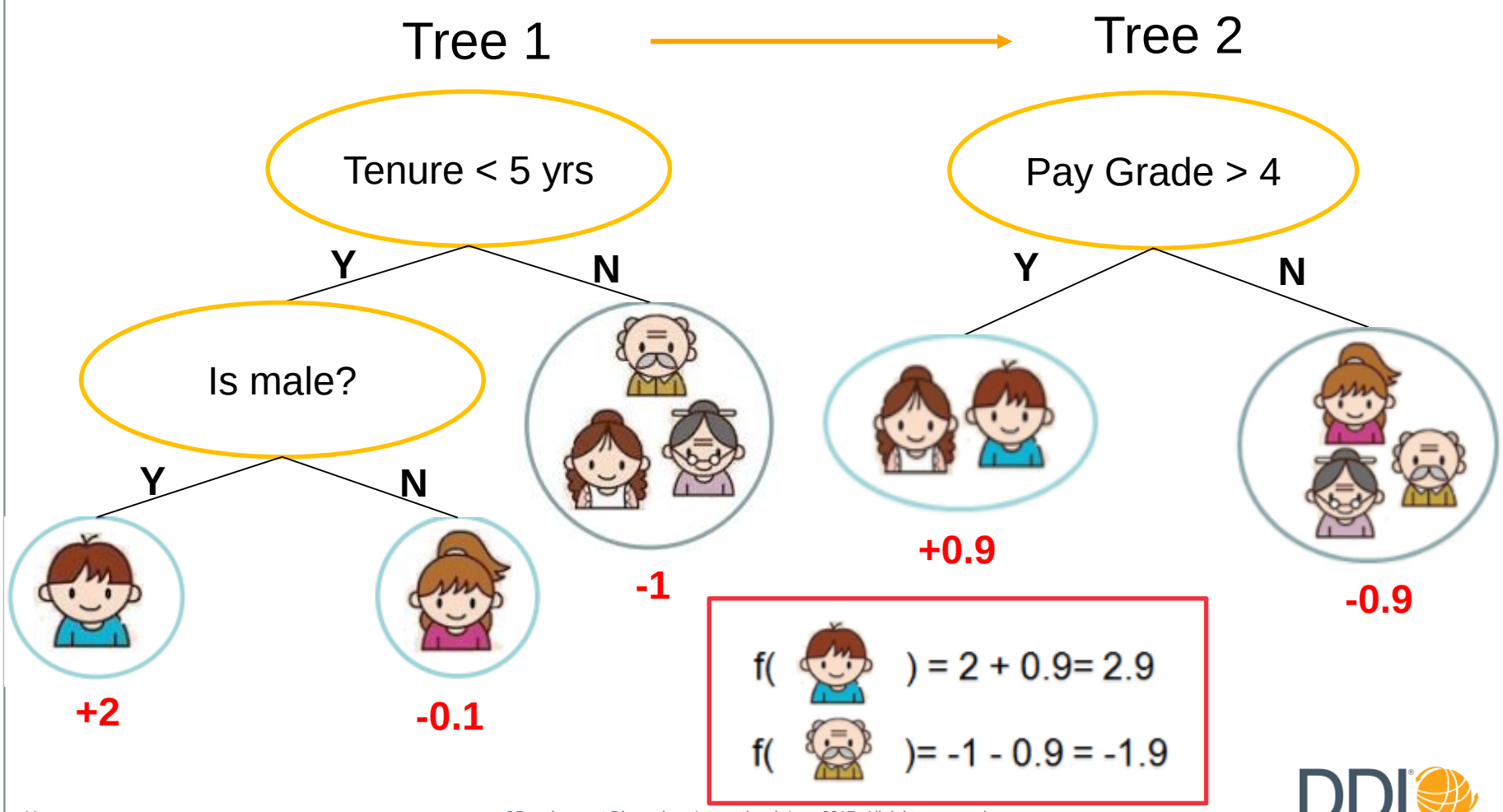


Trees are generated sequentially
$Tree_{n+1}$ corrects the errors from $Tree_n$

XGBoost

* Part of this content is based on these slides by the author of XGBoost

- Tree ensemble: sums the prediction of multiple trees



XGBoost: Advantages

- Boosting → accuracy
- Approximation algorithm → speed
- Greedy algorithm for tree learning → speed

- 10x faster than some other boosting methods ☺ (e.g., GBM)
- Most popular method in machine learning competitions (e.g., Kaggle, KDDCup)

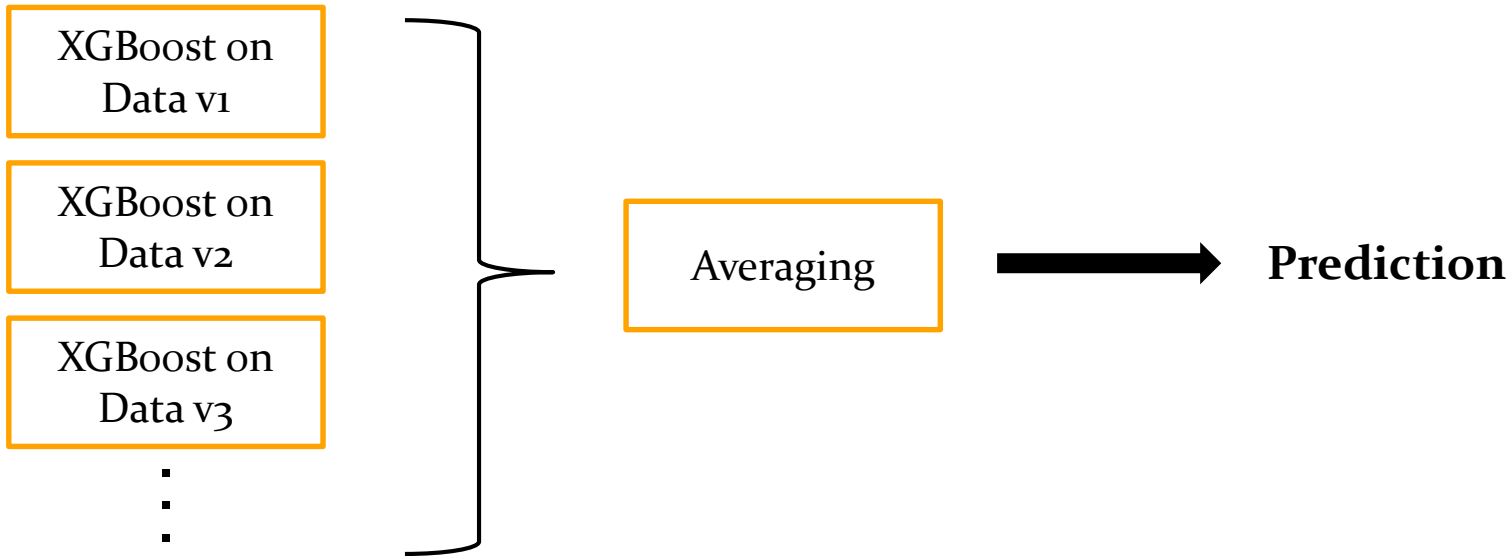
Hyperparameter Tuning

- Focused on hyperparameters that had the biggest impact on accuracy

Hyperparameter	Description
learning_rate	step size shrinkage used in update to prevents overfitting
n_estimators	number of boosted trees to fit
max_depth	maximum depth of a tree, increase this value will make the model more complex / likely to be overfitting.

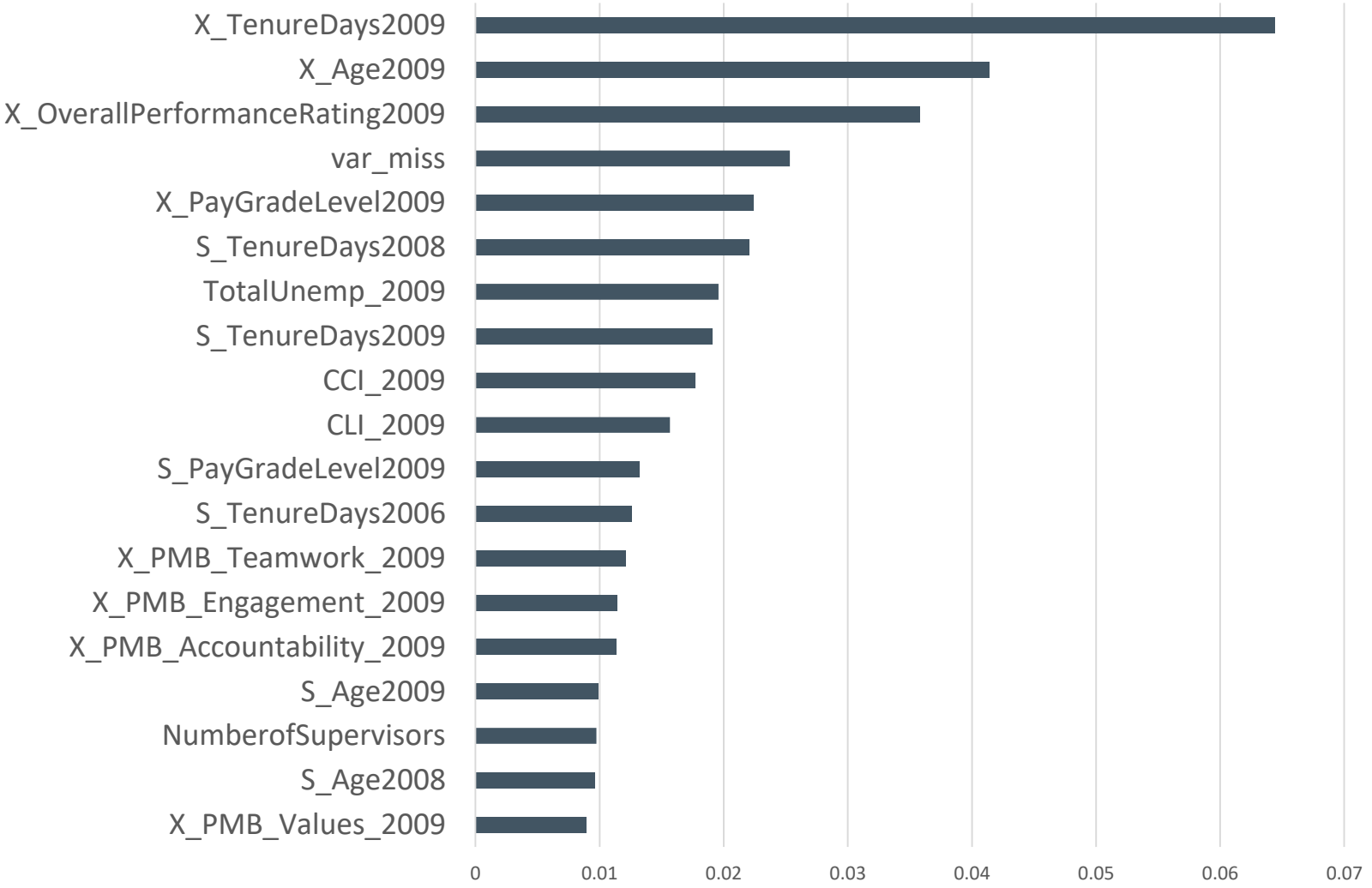
Ensemble

- Combine models in order to achieve more accurate and stable predictions
  - 4/5 of our top performing models are ensemble models

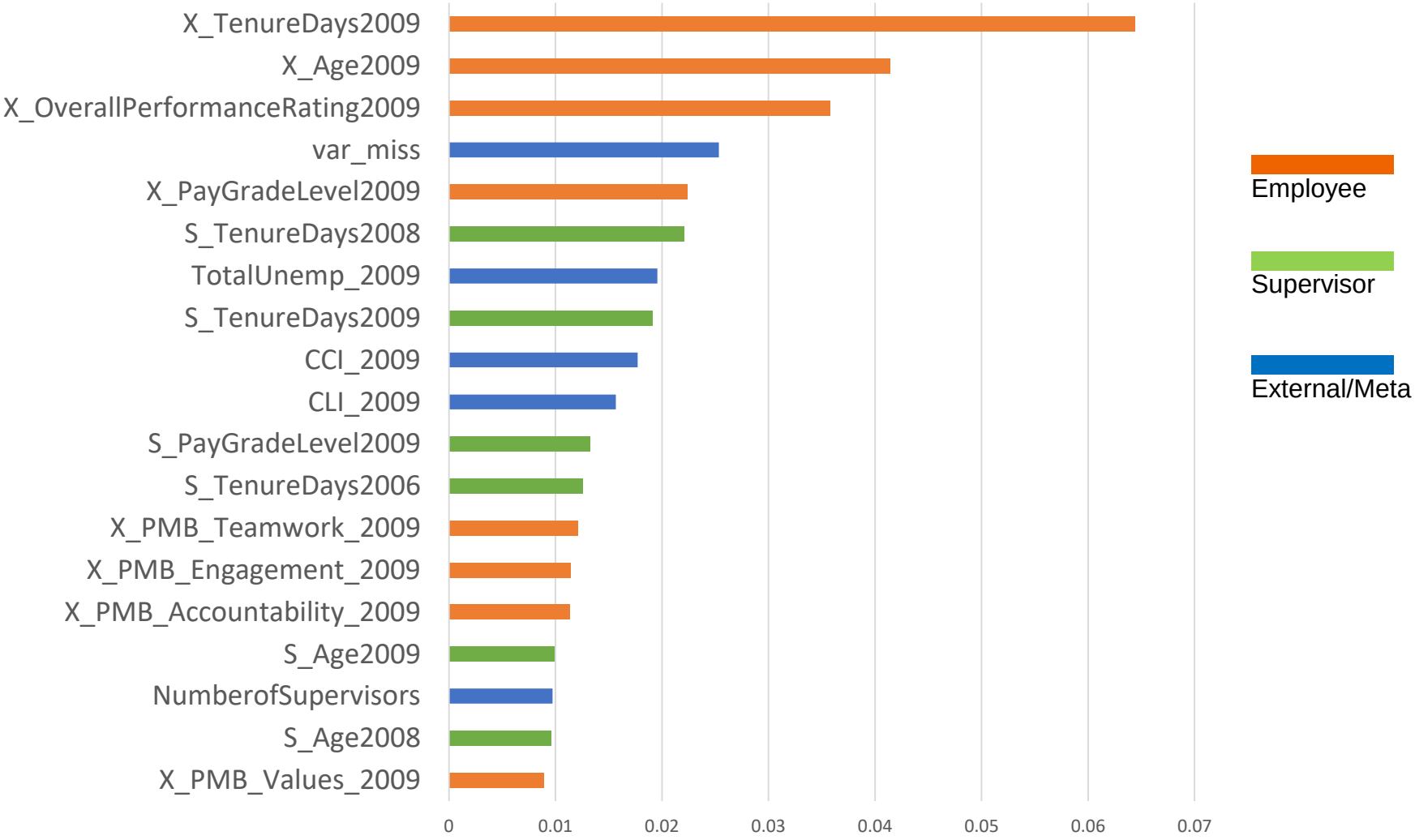


The Most Important Predictors of Turnover

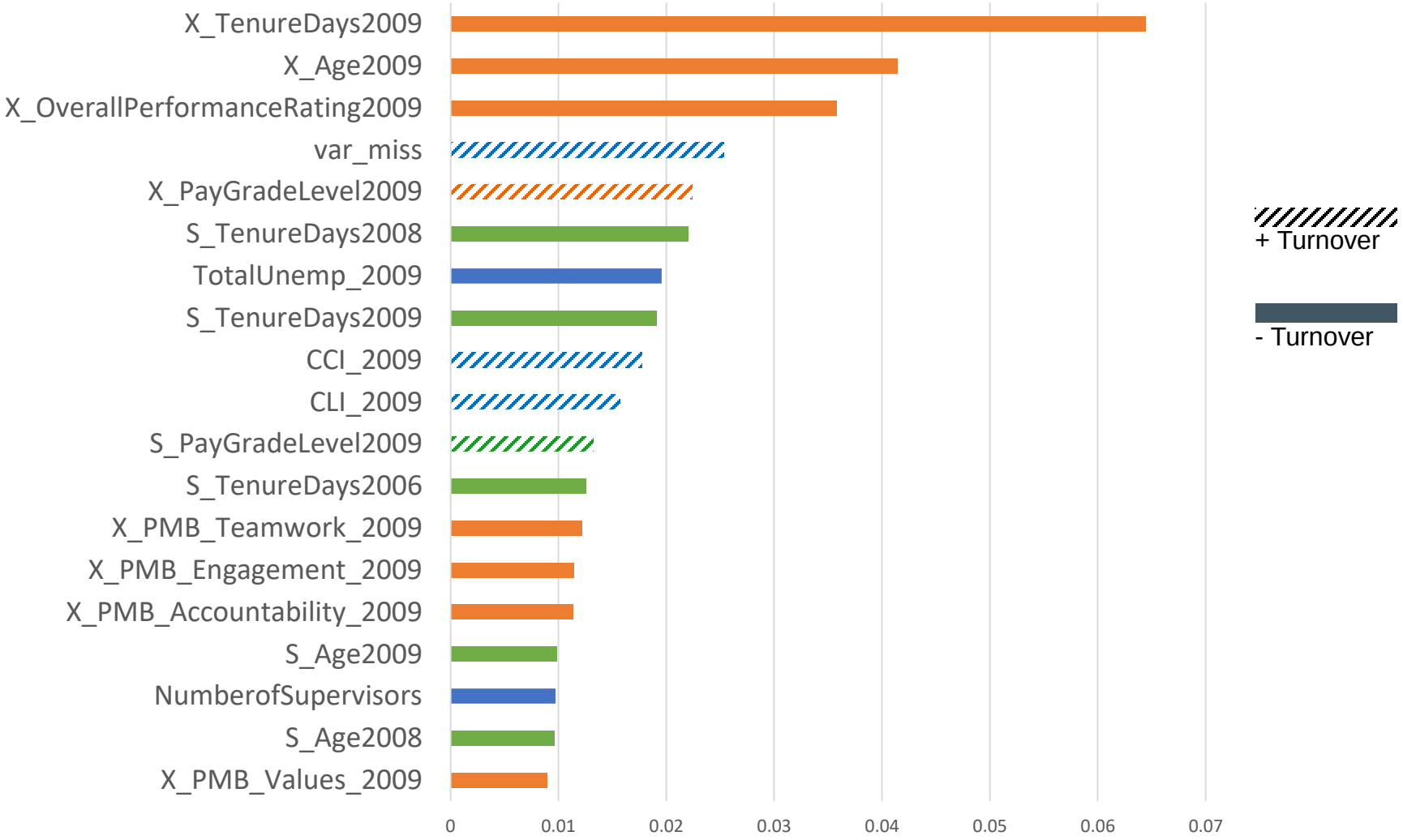
Feature Importance



Feature Importance



Feature Importance



Lessons Learned

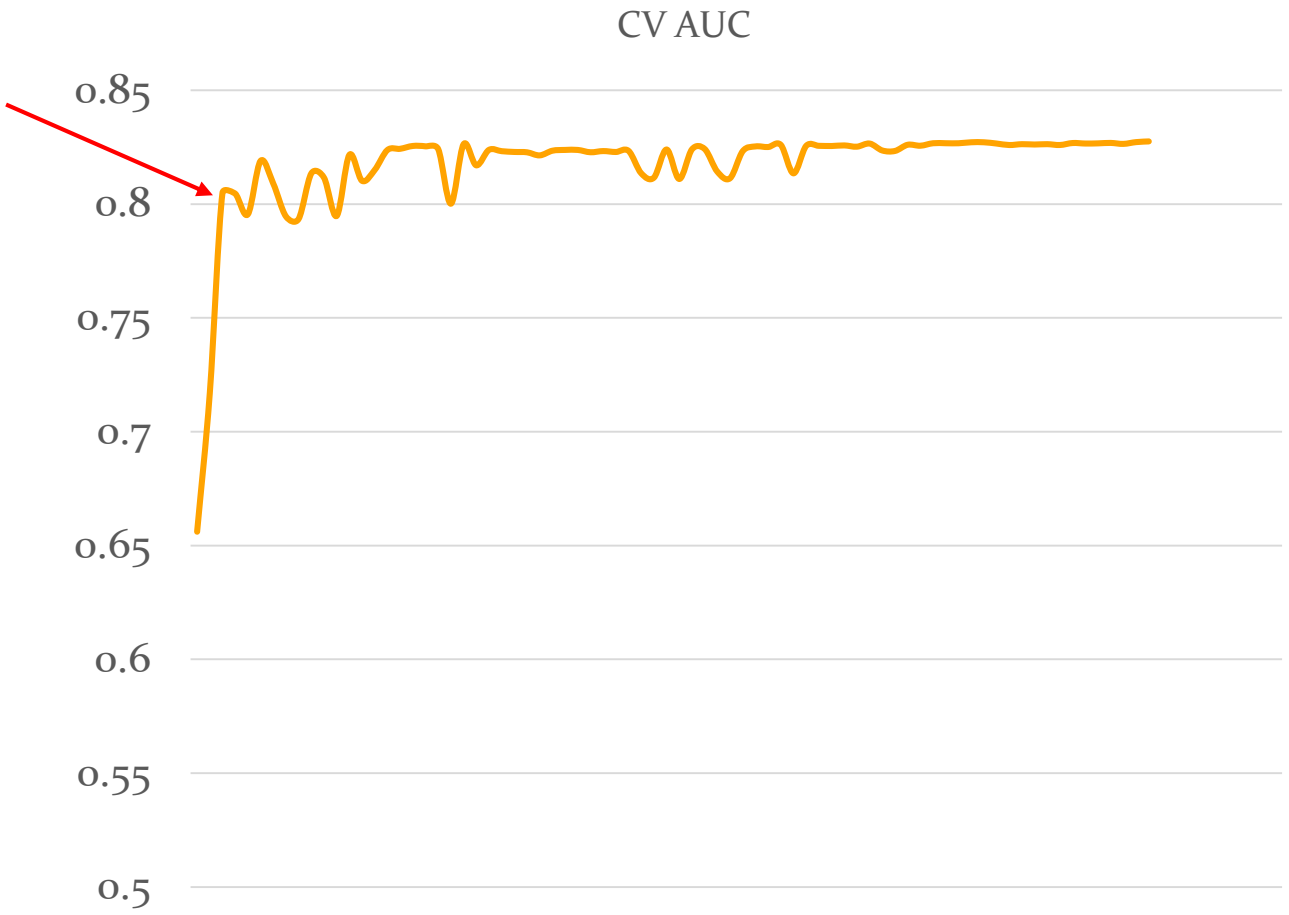
Lessons Learned

- A great opportunity to learn and practice machine learning techniques on a large organizational dataset!

- It was a lot of work… Time spent to validity gained ratio started to plateau quickly after Week 1. ☺

Lessons Learned

Week 1



Lessons Learned

- A great opportunity to learn and practice machine learning techniques on a large organizational dataset!

- It was a lot of work… Time spent to validity gained ratio started to plateau quickly after Week 1. ☺
- The impact of broader environmental variables on organizational phenomena.
- Considerations of model interpretability and fairness

Thank you!