Hyperparameter tuning part II

Plan for the video

- Tree-based models
  - GBDT: `XGBoost, LightGBM, CatBoost`
  - RandomForest/ExtraTrees
- Neural nets
  - `Pytorch, Tensorflow, Keras...`
- Linear models
  - SVM, logistic regression
  - `Vowpal Wabbit, FTRL`
- Factorization Machines (out of scope)
  - `libFM, libFFM`

Tree-based models

Model	Where
GBDT	*XGBoost* (dmlc/xgboost) *LightGBM* (Mictrosoft/LighGBM) *CatBoost* (catboost/catboost)
RandomForest, ExtraTrees	*scikit-learn*
Others	*RGF* (baidu/fast_rgf)

GBDT

XGBoost	LightGBM

GBDT

XGBoost	LightGBM
• max_depth	• max_depth/num_leaves

GBDT

XGBoost	LightGBM
• max_depth • subsample	• max_depth/num_leaves • bagging_fraction

GBDT

XGBoost	LightGBM
• max_depth • subsample • colsample_bytree, colsample_bylevel	• max_depth/num_leaves • bagging_fraction • feature_fraction

GBDT

XGBoost	LightGBM
• max_depth	• max_depth/num_leaves
• subsample	• bagging_fraction
• colsample_bytree, colsample_bylevel	• feature_fraction
• min_child_weight, lambda, alpha	• min_data_in_leaf, lambda_l1, lambda_l2

GBDT

XGBoost	LightGBM
• max_depth	• max_depth/num_leaves
• subsample	• bagging_fraction
• colsample_bytree, colsample_bylevel	• feature_fraction
• min_child_weight, lambda, alpha	• min_data_in_leaf, lambda_l1, lambda_l2
• eta num_round	• learning_rate num_iterations

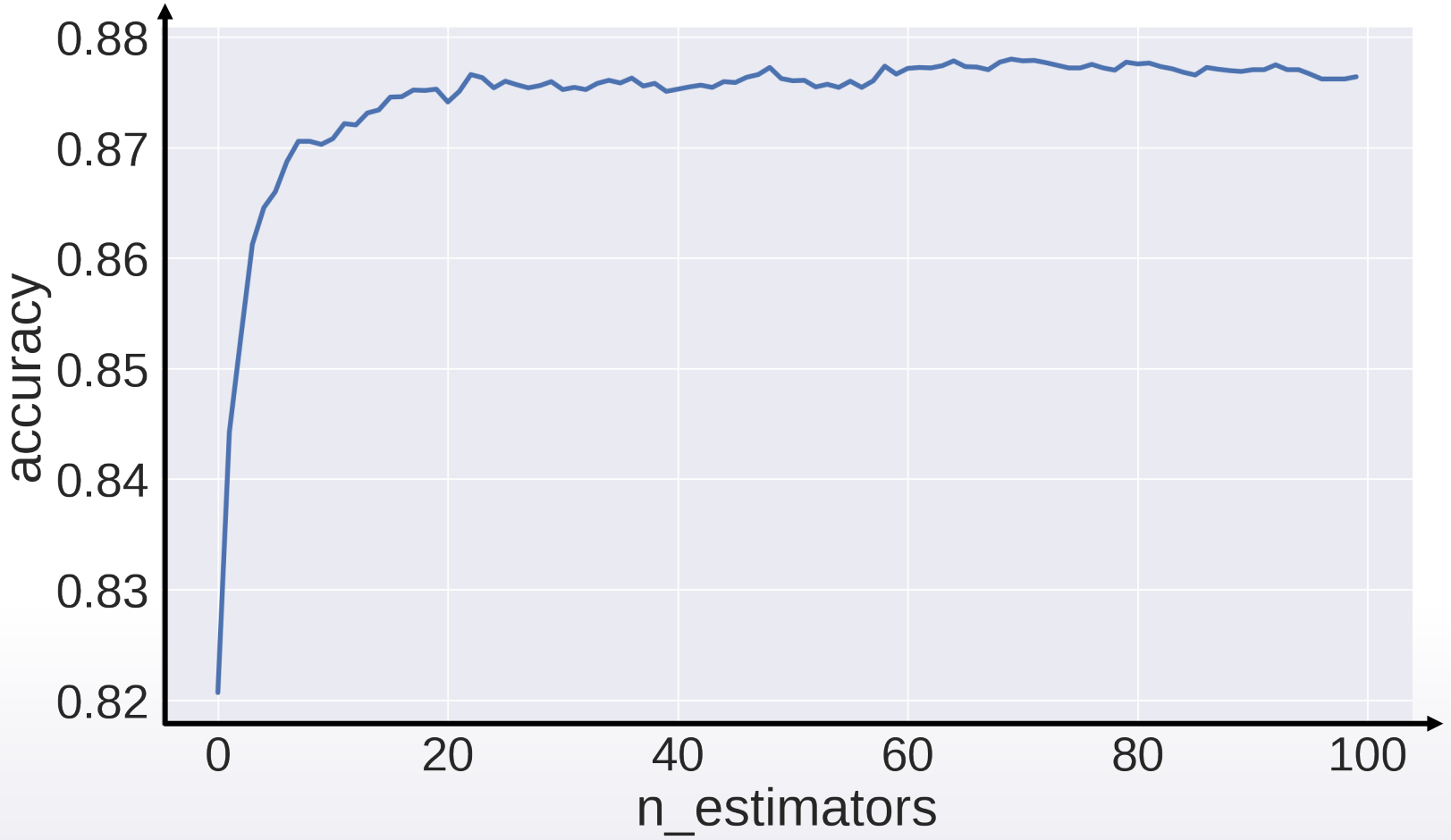
GBDT

XGBoost	LightGBM
• max_depth • subsample • colsample_bytree, colsample_bylevel • min_child_weight, lambda, alpha • eta num_round Others: • seed	• max_depth/num_leaves • bagging_fraction • feature_fraction • min_data_in_leaf, lambda_l1, lambda_l2 • learning_rate num_iterations Others: • *_seed

sklearn.RandomForest/ExtraTrees

sklearn.RandomForest/ExtraTrees

- N_estimators (the higher the better)



sklearn.RandomForest/ExtraTrees

- N_estimators (the higher the better)
- max_depth

sklearn.RandomForest/ExtraTrees

- N_estimators (the higher the better)
- max_depth
- max_features

sklearn.RandomForest/ExtraTrees

- N_estimators (the higher the better)
- max_depth
- max_features
- min_samples_leaf

sklearn.RandomForest/ExtraTrees

- N_estimators (the higher the better)
- max_depth
- max_features
- min_samples_leaf

Others:

- *criterion*

sklearn.RandomForest/ExtraTrees

- N_estimators (the higher the better)
- max_depth
- max_features
- min_samples_leaf

Others:

- *criterion*
- **random_state**
- **n_jobs**

Conclusion

- Tree-based models
  - GBDT: `XGBoost, LightGBM, CatBoost`
  - RandomForest/ExtraTrees
- Neural nets
  - `Pytorch, Tensorflow, Keras...`
- Linear models
  - SVM, logistic regression
  - `Vowpal Wabbit, FTRL`
- Factorization Machines (out of scope)
  - `libFM, libFFM`

Conclusion

- Hyperparameter tuning in general
  - General pipeline
  - Manual and automatic tuning
  - What should we understand about hyperparameters?
- Models, libraries and hyperparameter optimization
  - Tree-based models
  - Neural networks
  - Linear models