IBM Analytics

Machine Learning in Master Data Management Systems

Lars Bremer
lbremer@de.ibm.com
Senior Software Engineer, IBM

Mariya Chkalova
mariya.chkalova1@ibm.com
Software Engineer, IBM

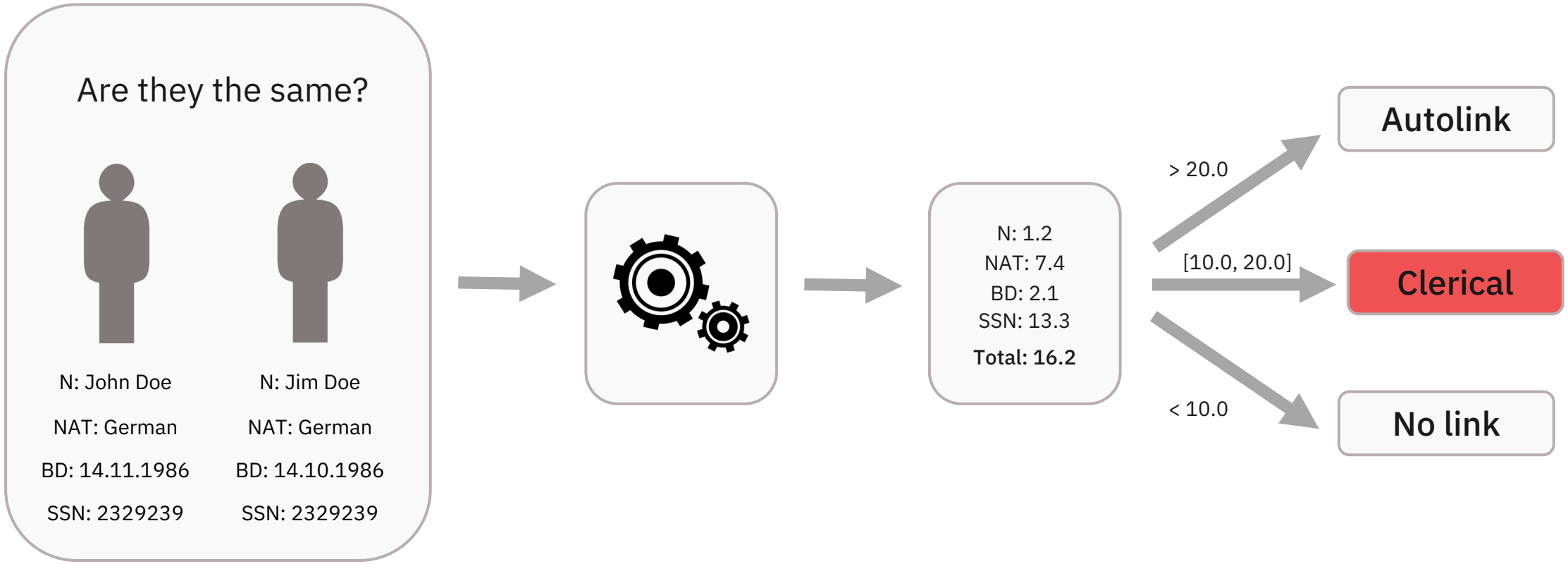
IBM

Legal Disclaimer

- The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

- References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

Clerical Task Management



Clerical Task Management

Business Executive

„It‘s expensive“

Data Steward

„It‘s boring“

Can we reduce the number of tasks with Machine Learning?

Input for Machine Learning

- Learn from task resolution history to auto-classify future potential duplicates
- Resolution history stored in our clients databases
- Matching engine can create comparison data

```
MEMRECNO,MEMRECNO2,CAUDTIME,RULETYPE,XNM,AXP,SSN,DOB,SEX,FPF2,OVERALL_CMPSCORE
29955364,45928598,2015-01-02 08:07:44,S,+0.66,+0.13,+0.00,+4.47,+0.26,-3.00,2.5
33087603,45928598,2015-01-02 08:07:44,S,+0.66,+0.13,+0.00,+4.47,+0.26,-3.00,2.5
32192384,45928598,2015-01-02 08:07:44,S,+0.66,+3.20,+0.00,+4.47,+0.26,-3.00,5.5
30214332,46274721,2015-01-02 08:10:07,S,+8.27,+1.33,+0.00,+4.55,+0.26,-2.00,12.4
46274721,46331036,2015-01-02 08:10:07,S,+8.27,+4.71,+5.01,+4.55,+0.26,+0.00,22.8
30214332,46331062,2015-01-02 08:10:07,S,+8.27,+4.71,+0.00,+4.55,+0.26,-2.00,15.7
46220762,46315567,2015-01-02 09:35:55,D,+8.07,+4.71,+0.00,+4.45,+0.35,-6.00,11.5
25754083,46264503,2015-01-02 15:32:23,D,+2.28,+1.33,+0.00,+4.53,+0.35,-3.00,5.4
25754083,46262360,2015-01-02 15:32:23,S,+8.27,+1.33,+0.00,+4.53,+0.35,-2.00,12.4
```

Evaluation Environment

Implementation

- Runtime: Python 3.7
- ML Libraries:
  - Scikit Learn 0.20.0
  - Xgboost 0.81
  - Imbalanced-learn 0.4.3

Environment

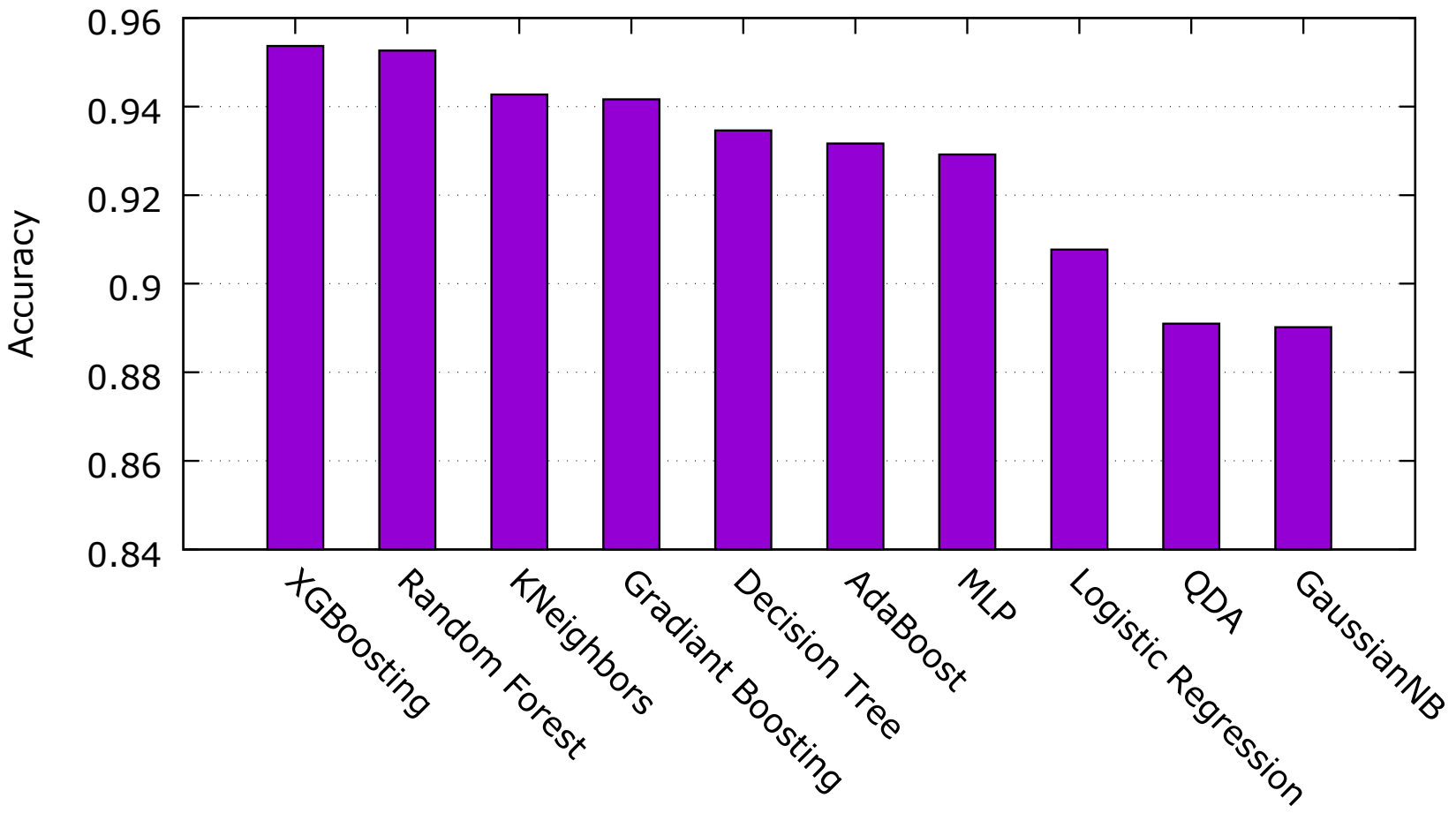
- Ubuntu 18.04 Virtual Machine
- CPU: 2.4 GHz with 8 cores
- Memory: 16GB

Data

- Healthcare Customer
- 1.3 million steward decisions

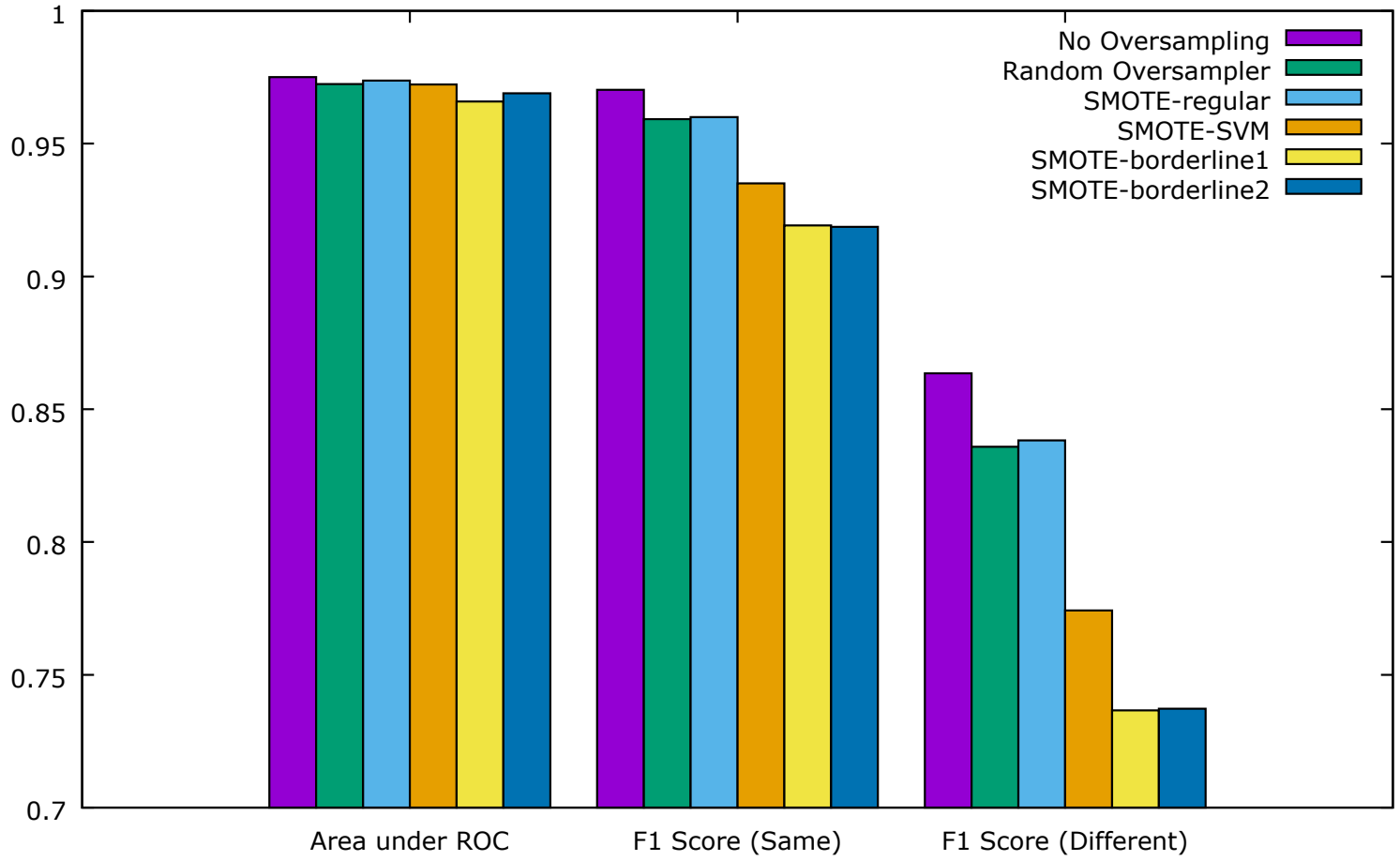
Exploration of different Machine Learning Algorithms

- Splitting data
  - Used 80% of randomly selected data to train model
  - Used remaining 20% to verify ML results
- Evaluated multiple classifiers
  - Random forest showed best results w.r.t. quality of predictions and training speed



Skewed Data

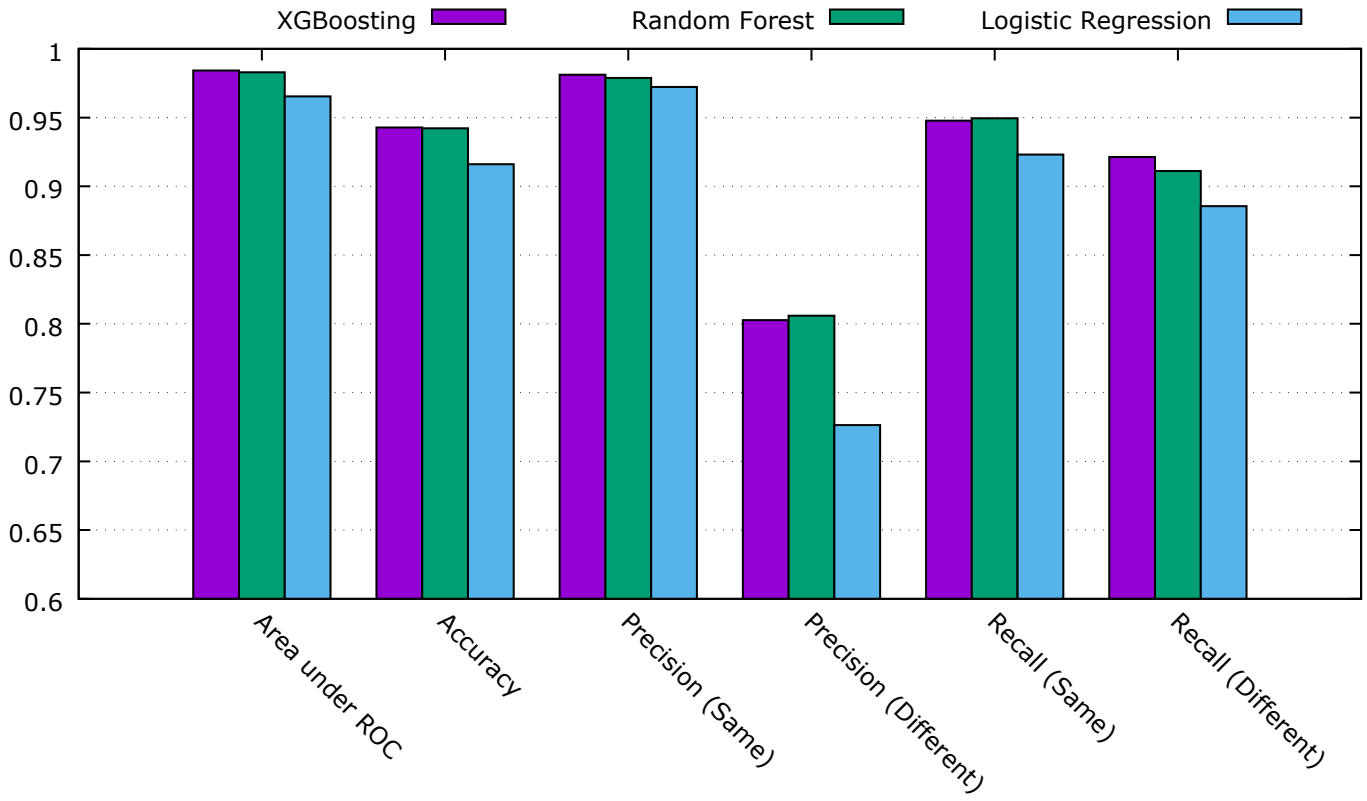
- The clerical data is often skewed
  - 75% same, 25% different
- To compensate, we evaluated different sampling methods
- No oversampling yields best results overall.
- Random oversampling and SMOTE regular perform best among sampling algorithms and show better precision for majority class.



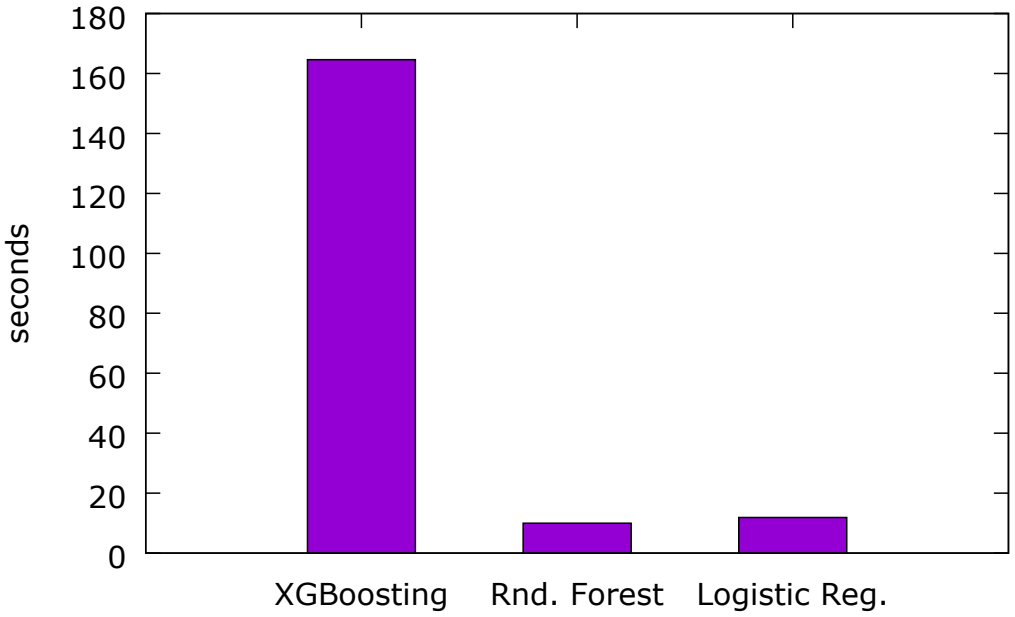
Artificial Features

- Zero values are categories
- To reduce the impact we added a additional columns
- No affect on tree-based classifiers
- Logistic regression benefits

Performance Comparison

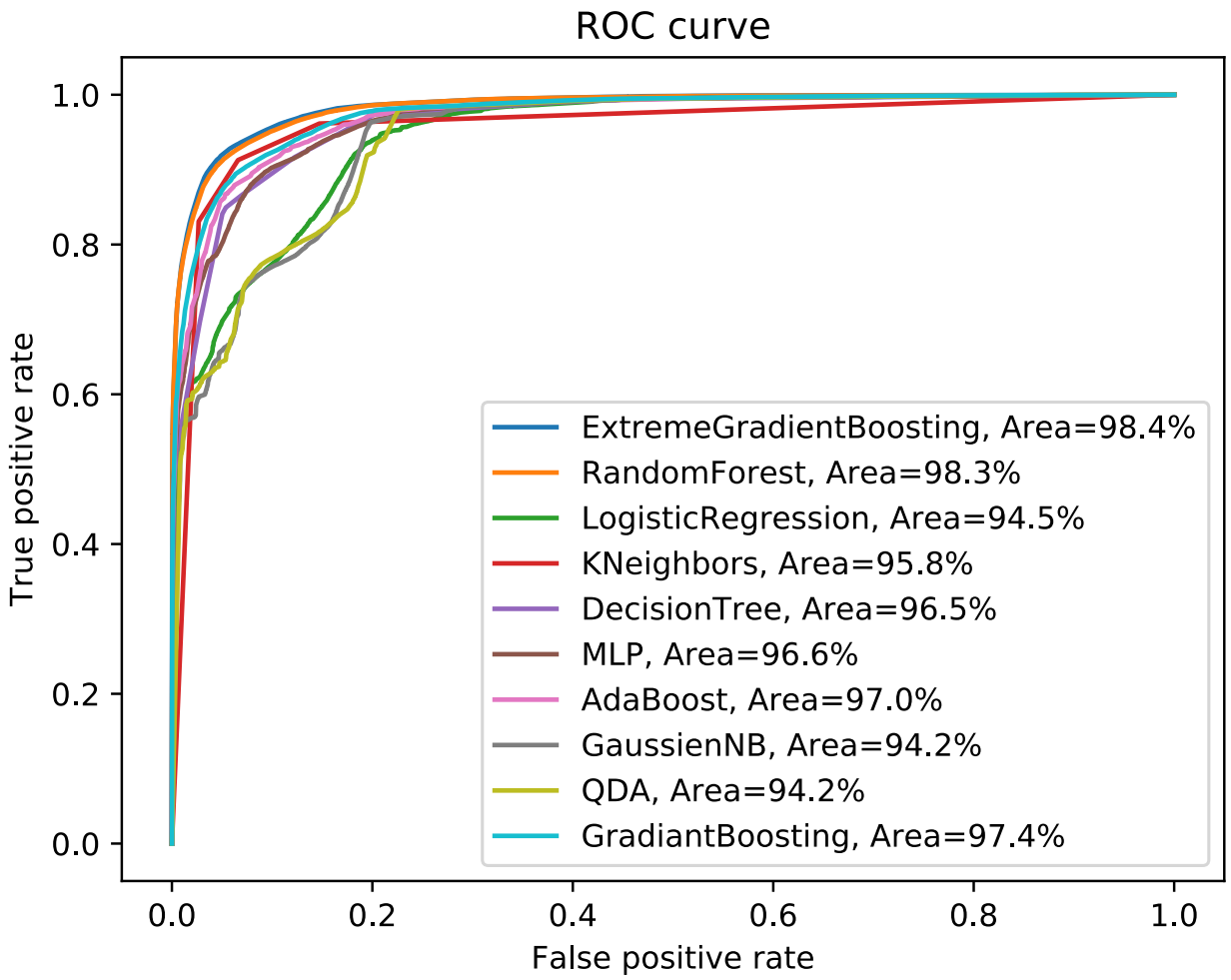


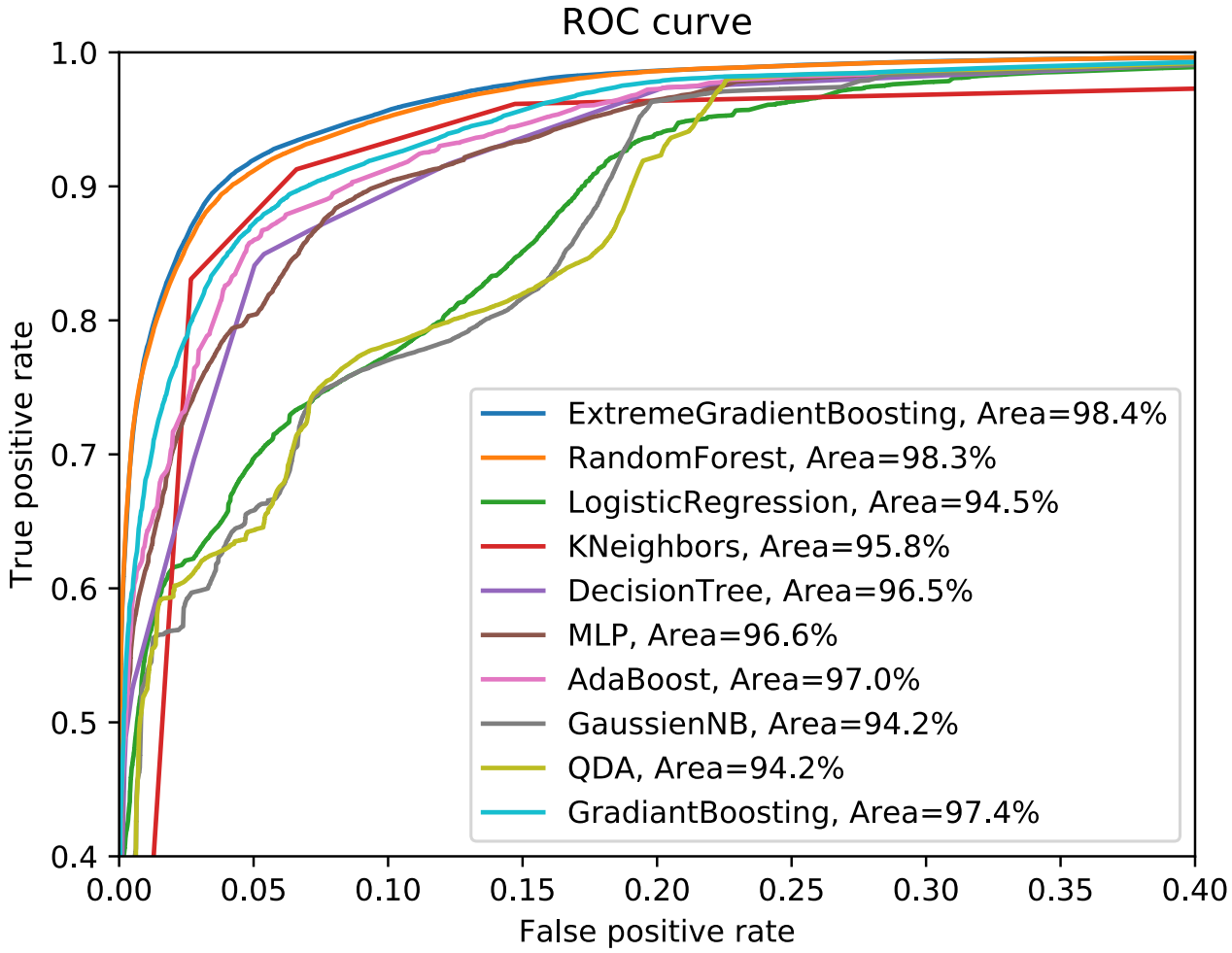
Prediction Quality



Training Speed

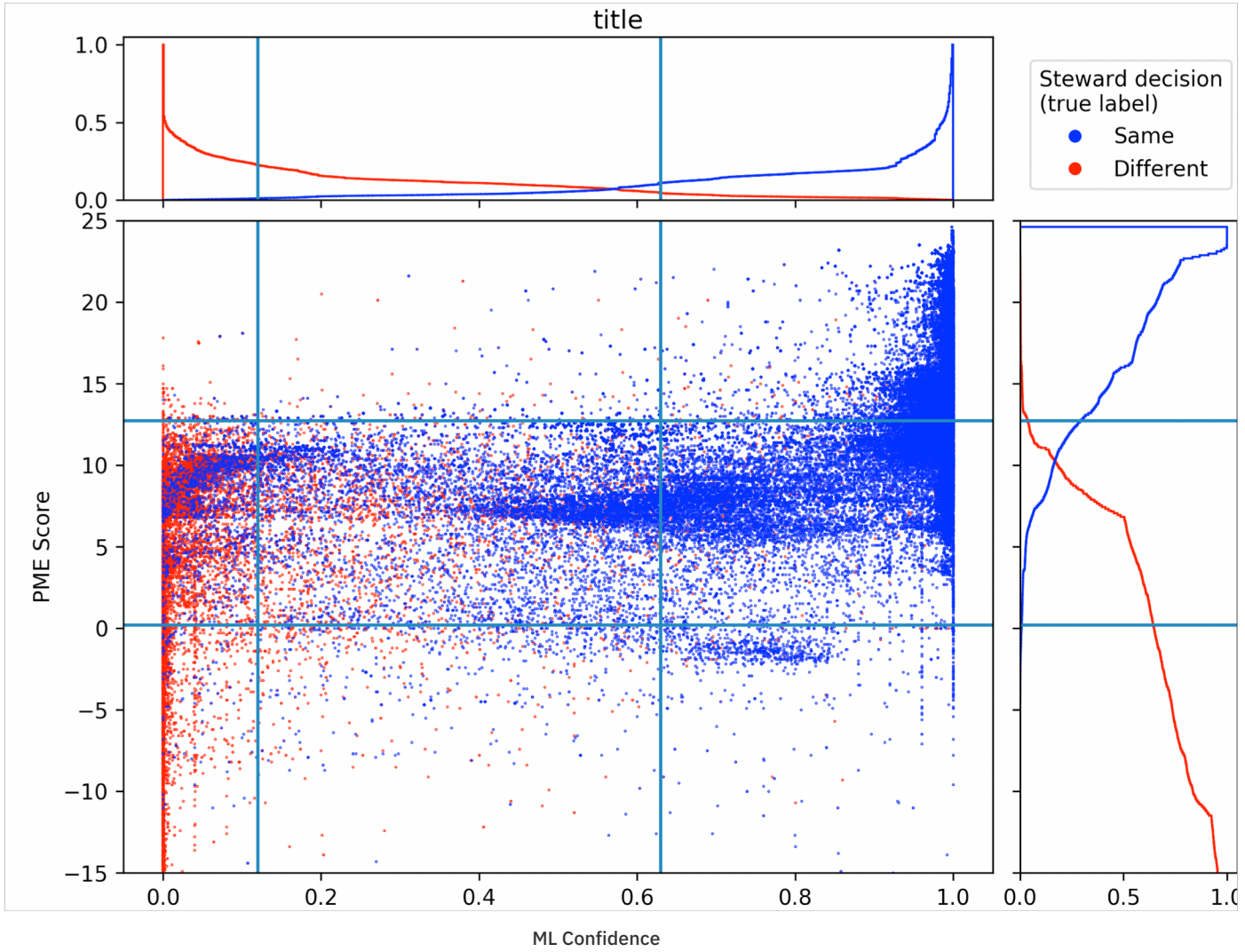
ROC Curve Comparison





Comparing PME results with ML Recommendations

- Comparing matching score with ML confidence
- Matching Engine
  - Clerical Tasks: 106,218
  - False Positive Rate: 1.02%
  - False Negative Rate: 0.98%
- Machine Learning
  - Clerical Tasks: 34,792
  - False Positive Rate: 0.93%
  - False Negative Rate: 0.96%

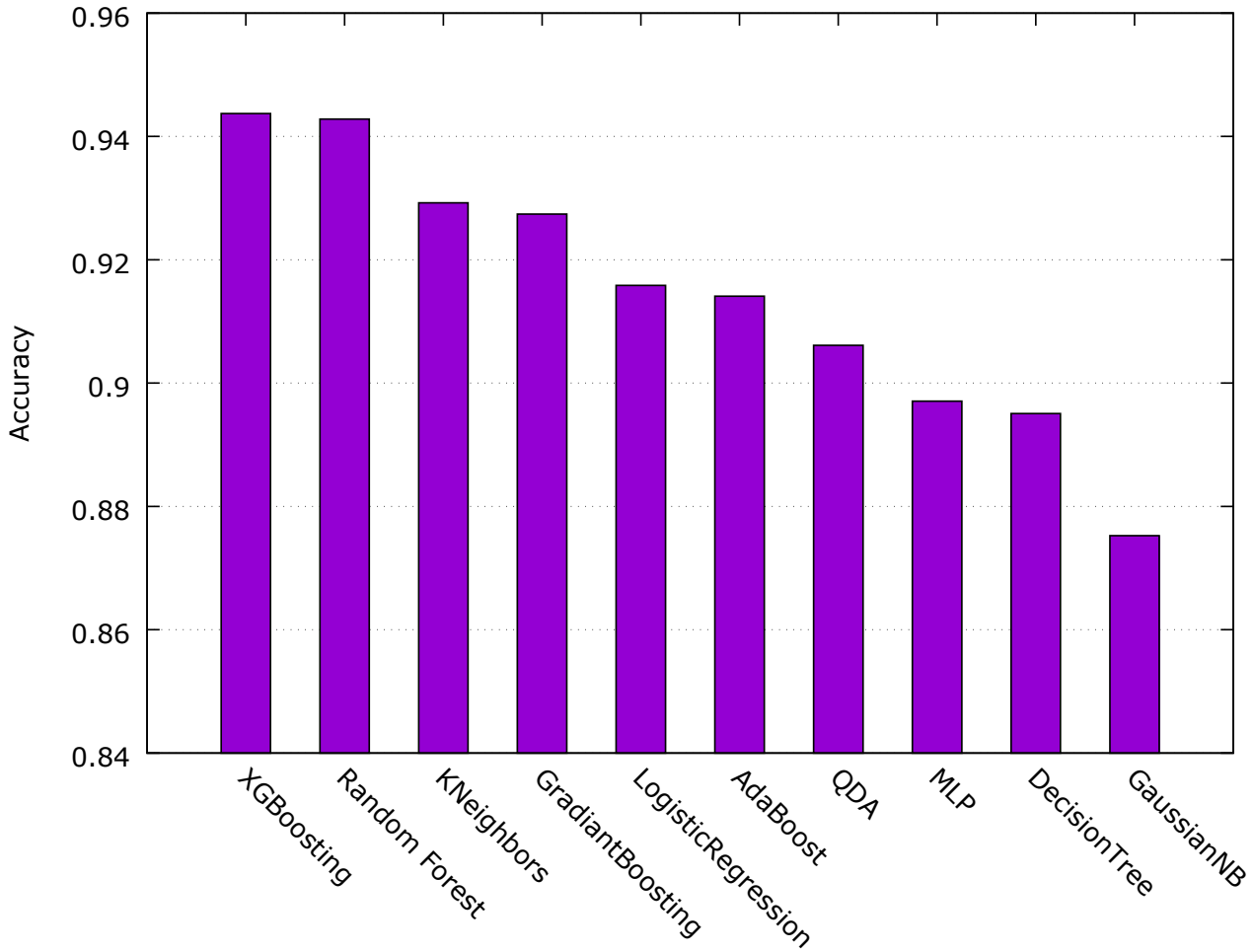


Final Results

- Using Random Oversampling
- Using Artificial Features
- Results
  - Accuracy = 0.94
  - Precision = 0.98 (same), 0.80 (different)
  - Recall = 0.94 (same), 0.91 (different)

We showed that the ML approach works better than a highly tuned Matching Engine.

Holding the false negative and positive rates at around 1% we can reduce the number of clerical task by two thirds.



Outlook

Impact of Data Volume

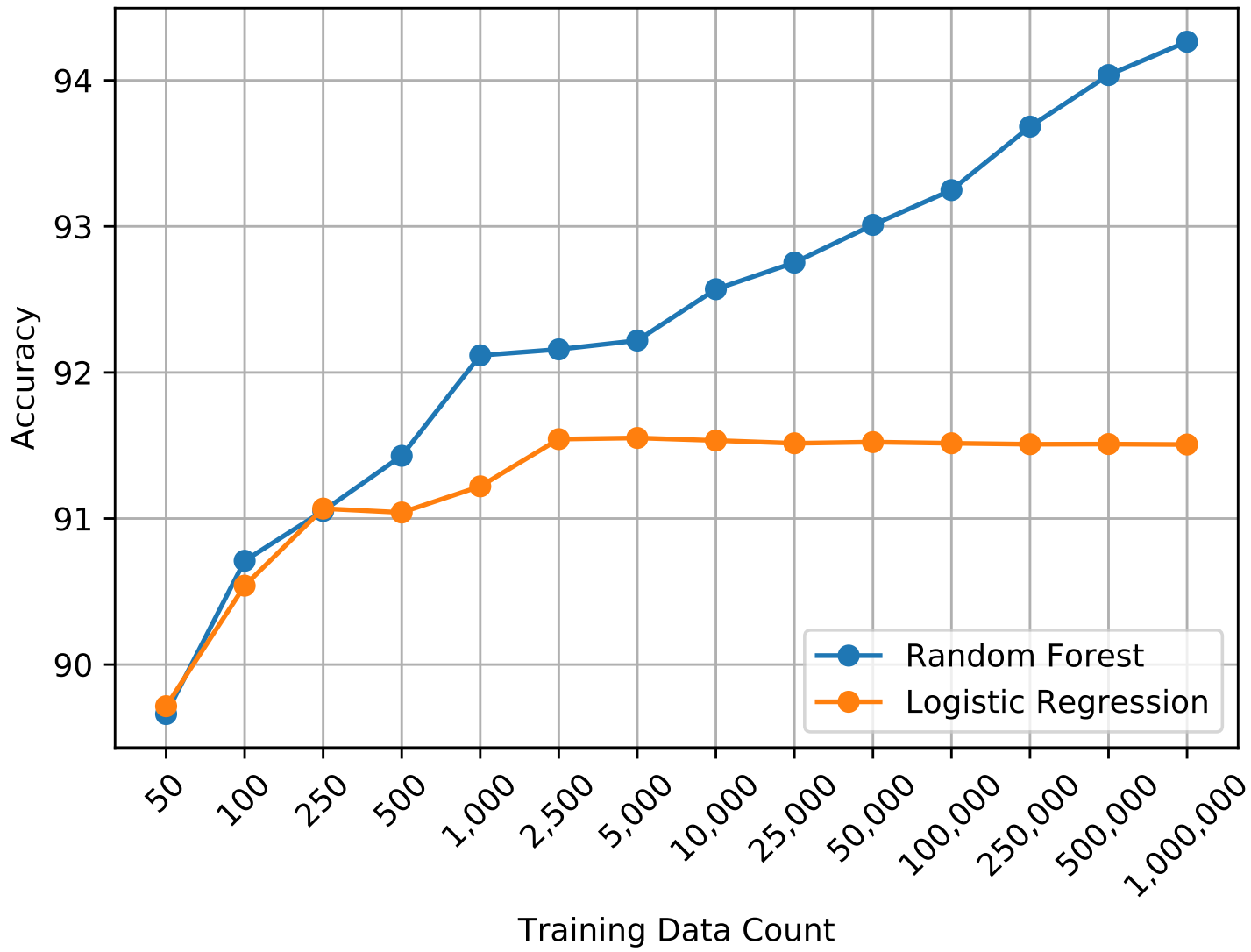
Plot created by training with different data volumes on the same test data. We executed multiple runs and choose the median for plotting.

Logistic Regression

Flattens out at about 91.5%

Random Forest

Prediction quality keeps on improving. With 1,000 resolved tasks accuracy is over 92%



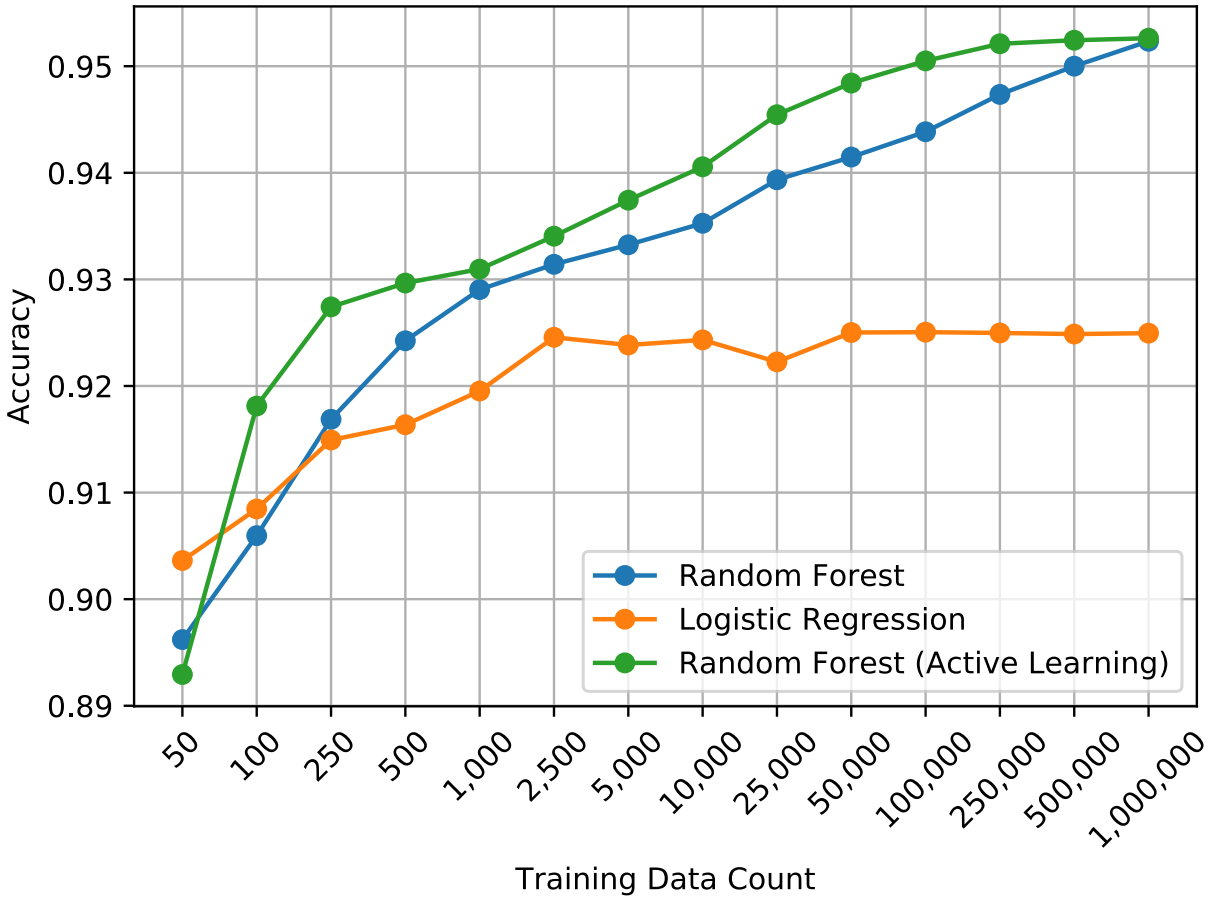
Active Learning Comparison

Clustering

Using k-means to identify first 10 tasks to process by data stewards.

Active Learning

Actively suggesting the next 10 tasks with most information gain to process by data stewards.



Thank You

Backup