

Tuning your model

PREDICTING CUSTOMER CHURN IN PYTHON



Mark Peterson

Senior Data Scientist, Alliance Data

Refresher

```
from sklearn.svm import SVC
```

```
svc = SVC()
```

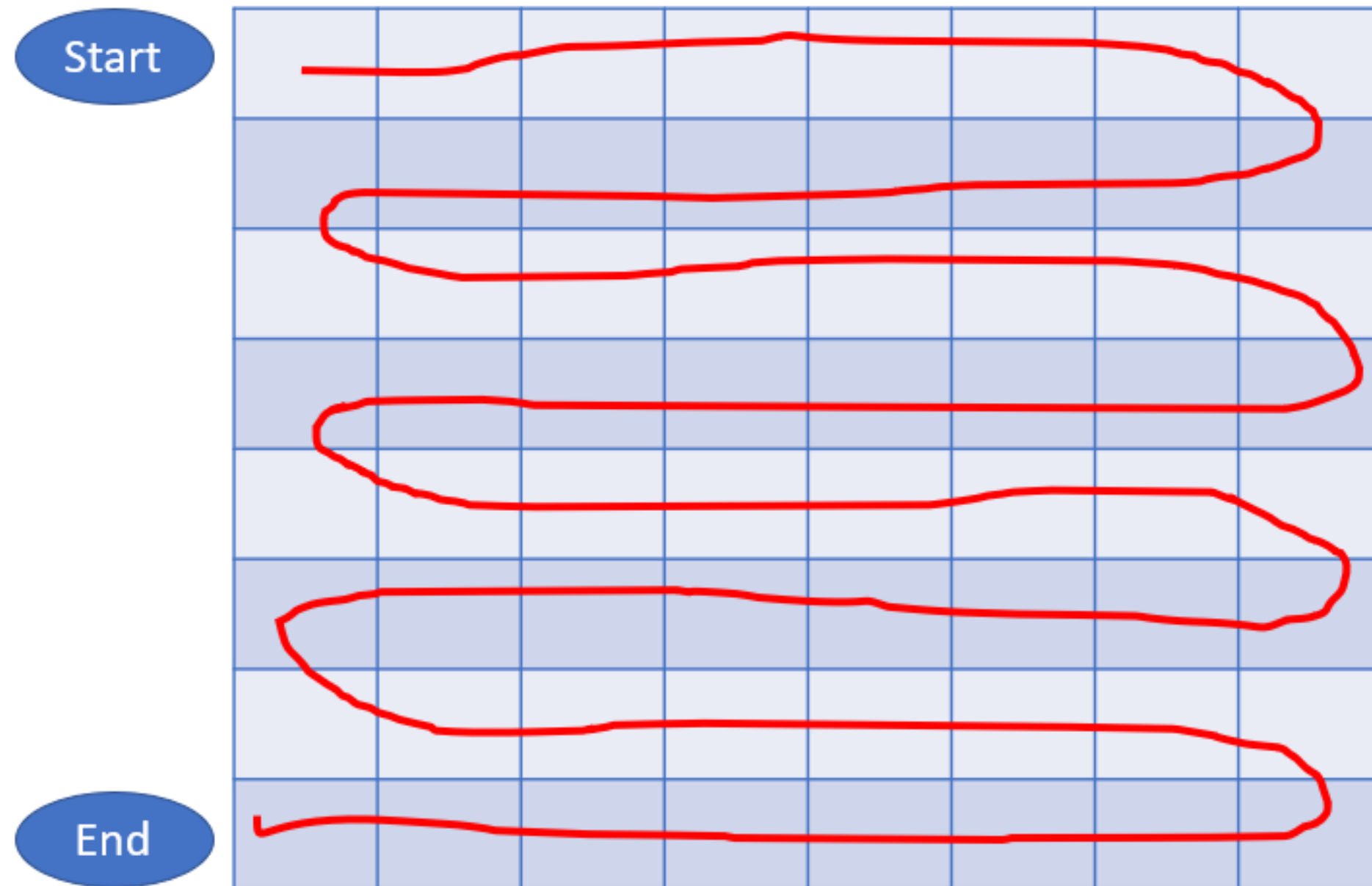
```
svc.fit(telco['data'], telco['target'])
```

```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,  
    decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',  
    max_iter=-1, probability=False, random_state=None, shrinking=True,  
    tol=0.001, verbose=False)
```

Random forest hyperparameters

Parameter	Purpose
n_estimators	Number of trees
criterion	Quality of Split
max_features	Number of features for best split
max_depth	Max depth of tree
min_sample_splits	Minimum samples to split node
bootstrap	Whether Bootstrap samples are used

Grid search



Grid search in sklearn

```
from sklearn.model_selection import GridSearchCV
param_grid = {'n_estimators': np.arange(10, 51)}
clf_cv = GridSearchCV(RandomForestClassifier(), param_grid)
clf_cv.fit(X, y)
clf_cv.best_params_
```

```
{'n_estimators': 43}
```

```
clf_cv.best_score_
```

```
0.9237923792379238
```

Happy tuning!

PREDICTING CUSTOMER CHURN IN PYTHON

Feature importances

PREDICTING CUSTOMER CHURN IN PYTHON



Mark Peterson

Senior Data Scientist, Alliance Data

Feature importances

- Scores representing how much each feature contributes to a prediction
 - Effective way to communicate results to stakeholders
-
- Which features are important drivers of churn?
 - Which features can be removed from the model?

Interpretability vs accuracy

- Different models have different strengths
- Need to balance prediction accuracy vs. interpretability

Random forest feature importances

```
random_forest = RandomForestClassifier()
```

```
random_forest.fit(X_train, y_train)
```

```
random_forest.feature_importances_
```

```
array([0.05069206, 0.04501006, 0.14427055, 0.08018487, 0.05222886,  
       0.04418832, 0.11497537, 0.04463341, 0.12179754, 0.04756014,  
       0.06818244, 0.05074536, 0.04616382, 0.03110577, 0.05826142])
```

Let's practice!

PREDICTING CUSTOMER CHURN IN PYTHON

Adding New Features

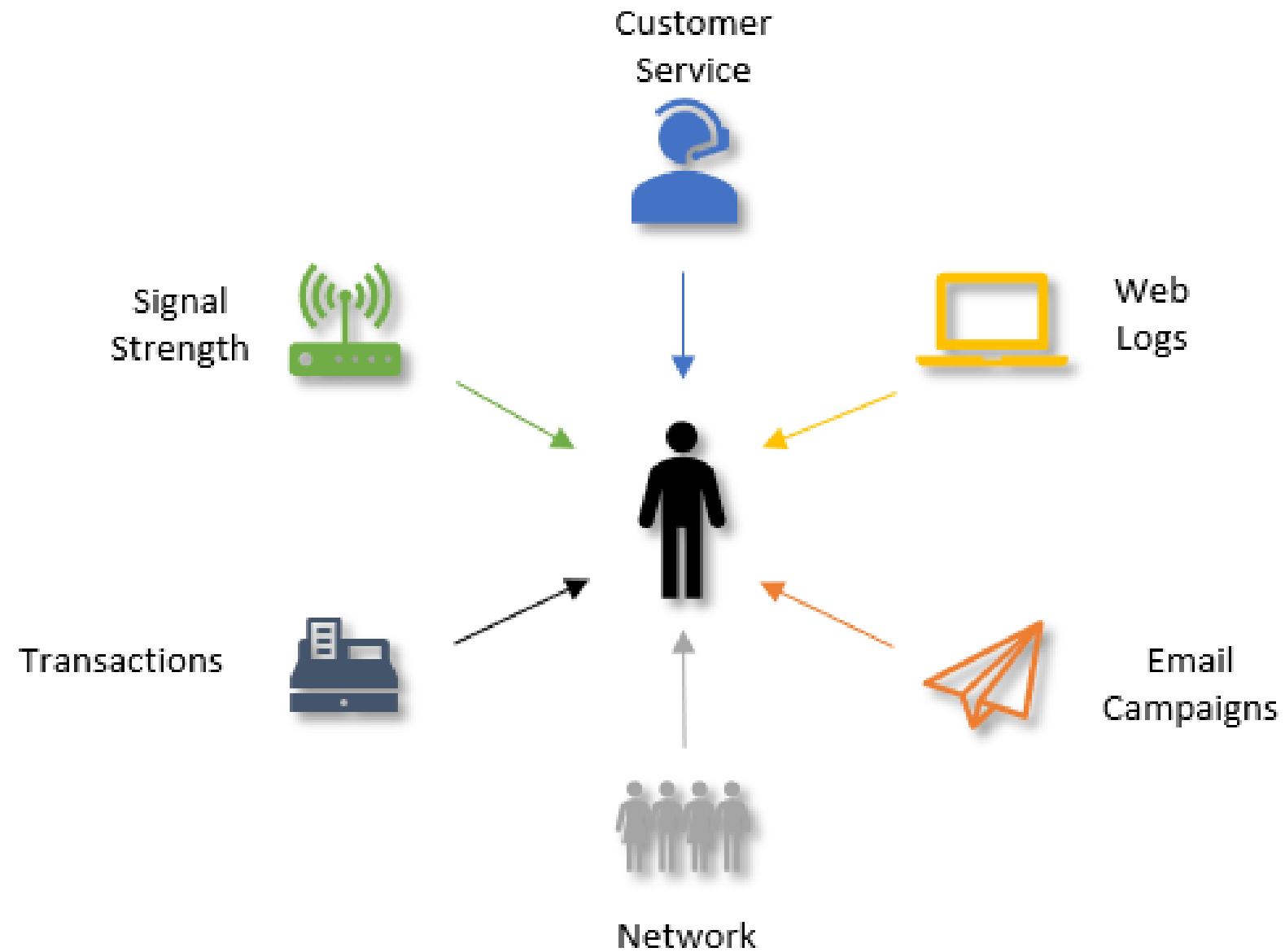
PREDICTING CUSTOMER CHURN IN PYTHON



Mark Peterson

Senior Data Scientist, Alliance Data

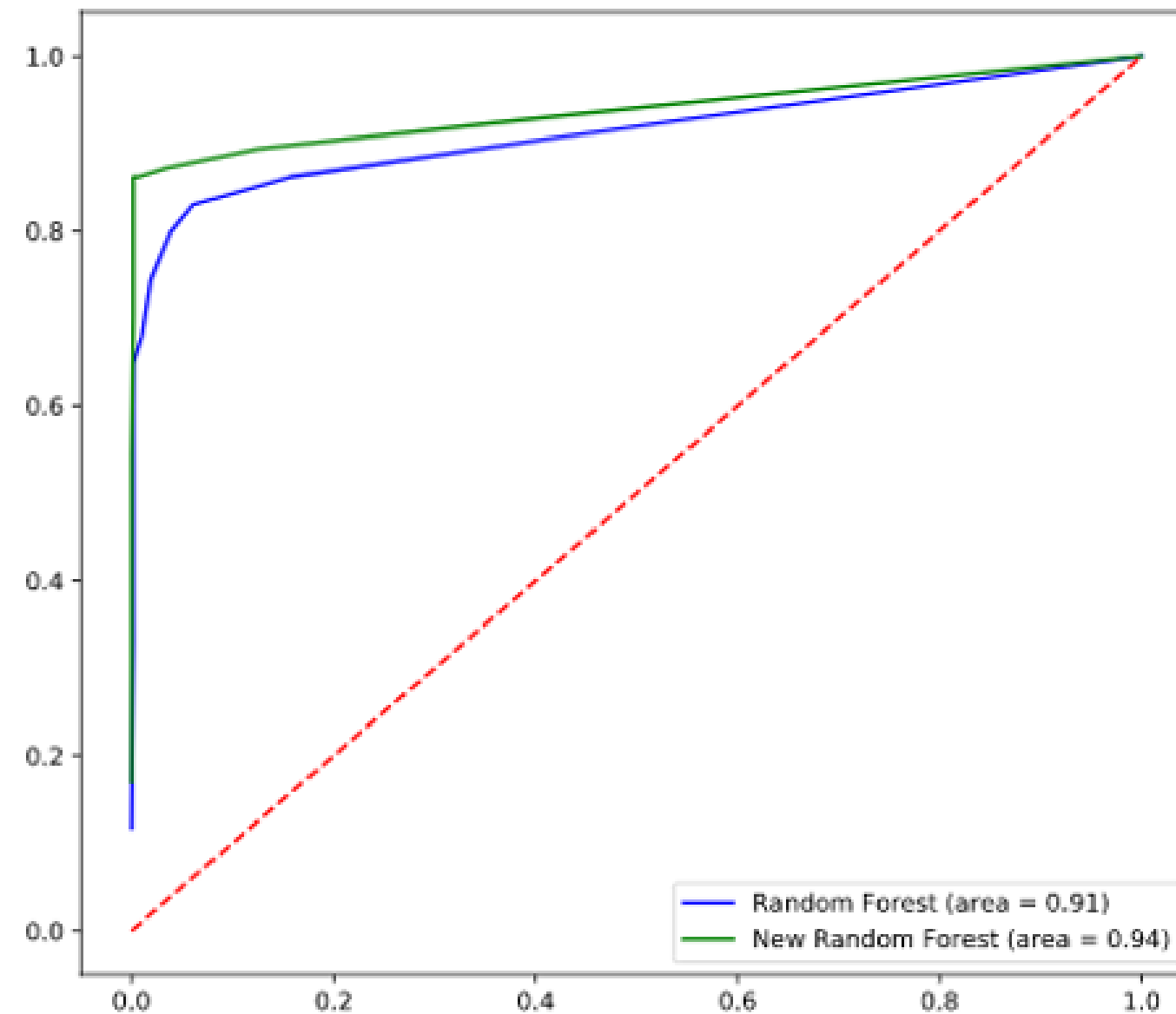
Additional Data Sources



Churn Features

- Region Code
- Total Charges
- Total Minutes
- Minutes Per Call
- Cost Per Call
- Total Calls

Model Improvement



Benefits | Costs

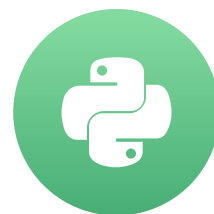
- Benefits
 - Improved Return on Investment
 - Decreased Costs
 - Increased Performance
- Costs
 - Increased Complexity
 - Increased Resources
 - Increased Time to Operationalizing

Let's practice!

PREDICTING CUSTOMER CHURN IN PYTHON

Final thoughts

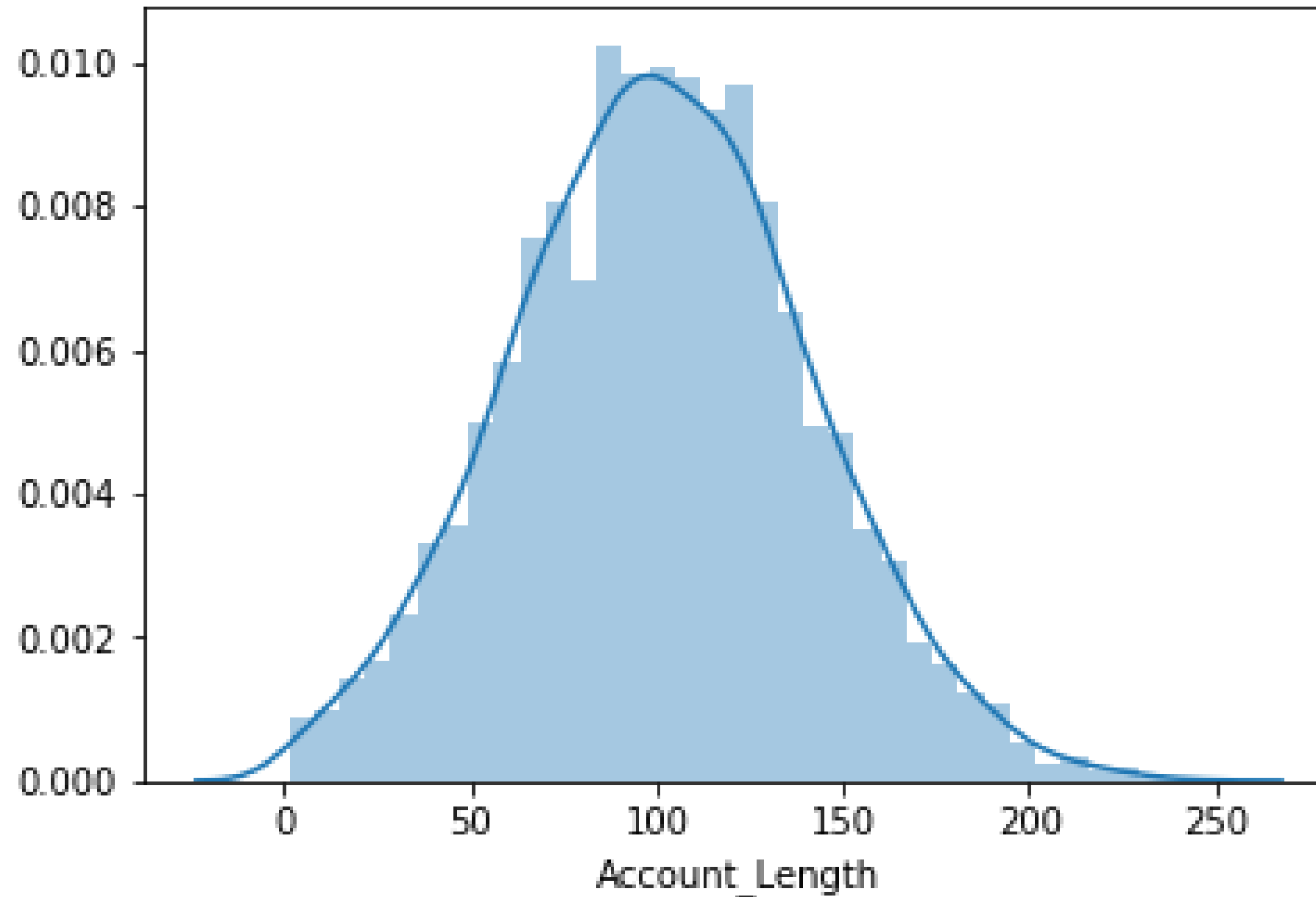
PREDICTING CUSTOMER CHURN IN PYTHON



Mark Peterson

Senior Data Scientist, Alliance Data

- Defined customer churn
- Exploratory data analysis

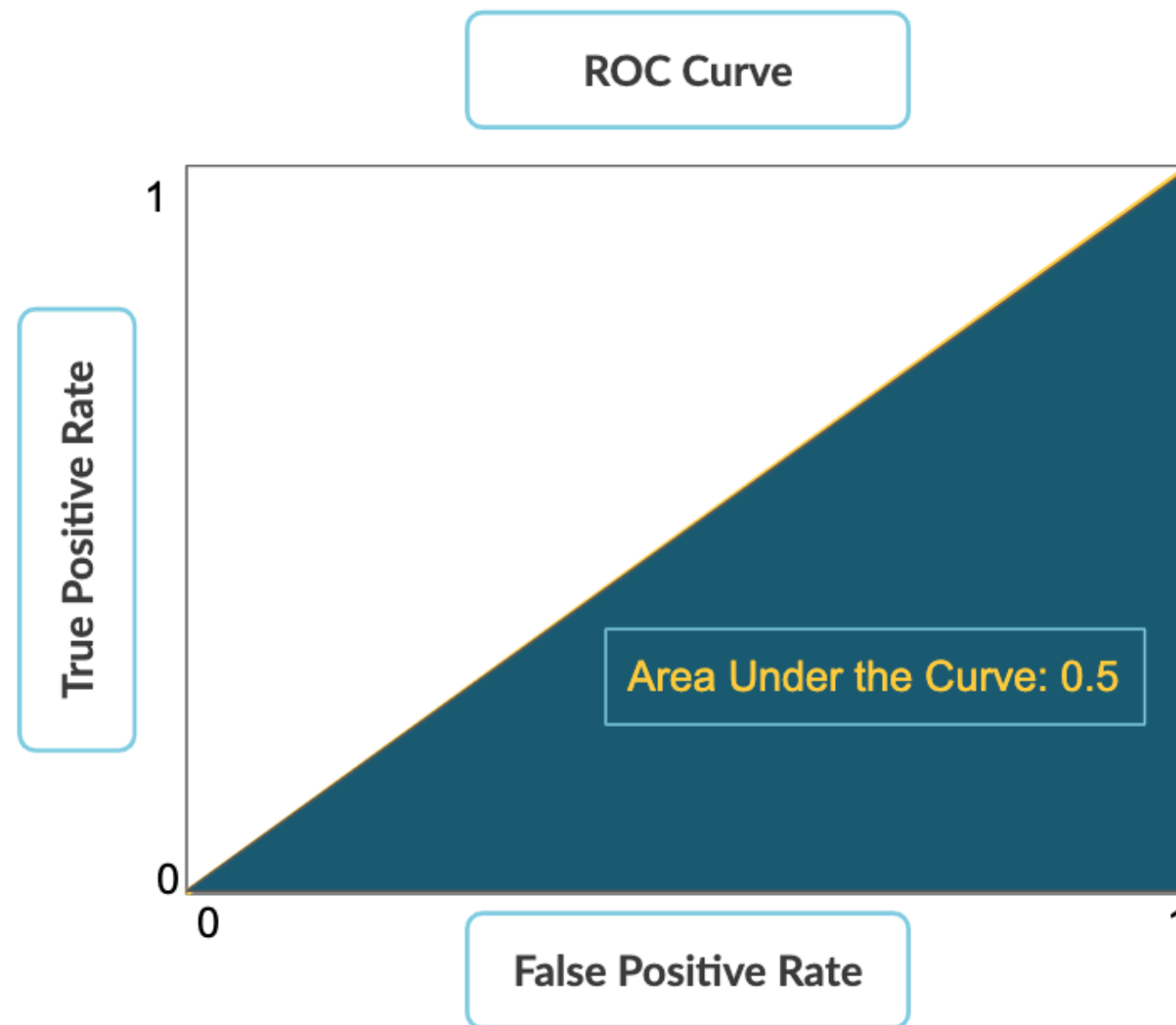


- Drop unnecessary features

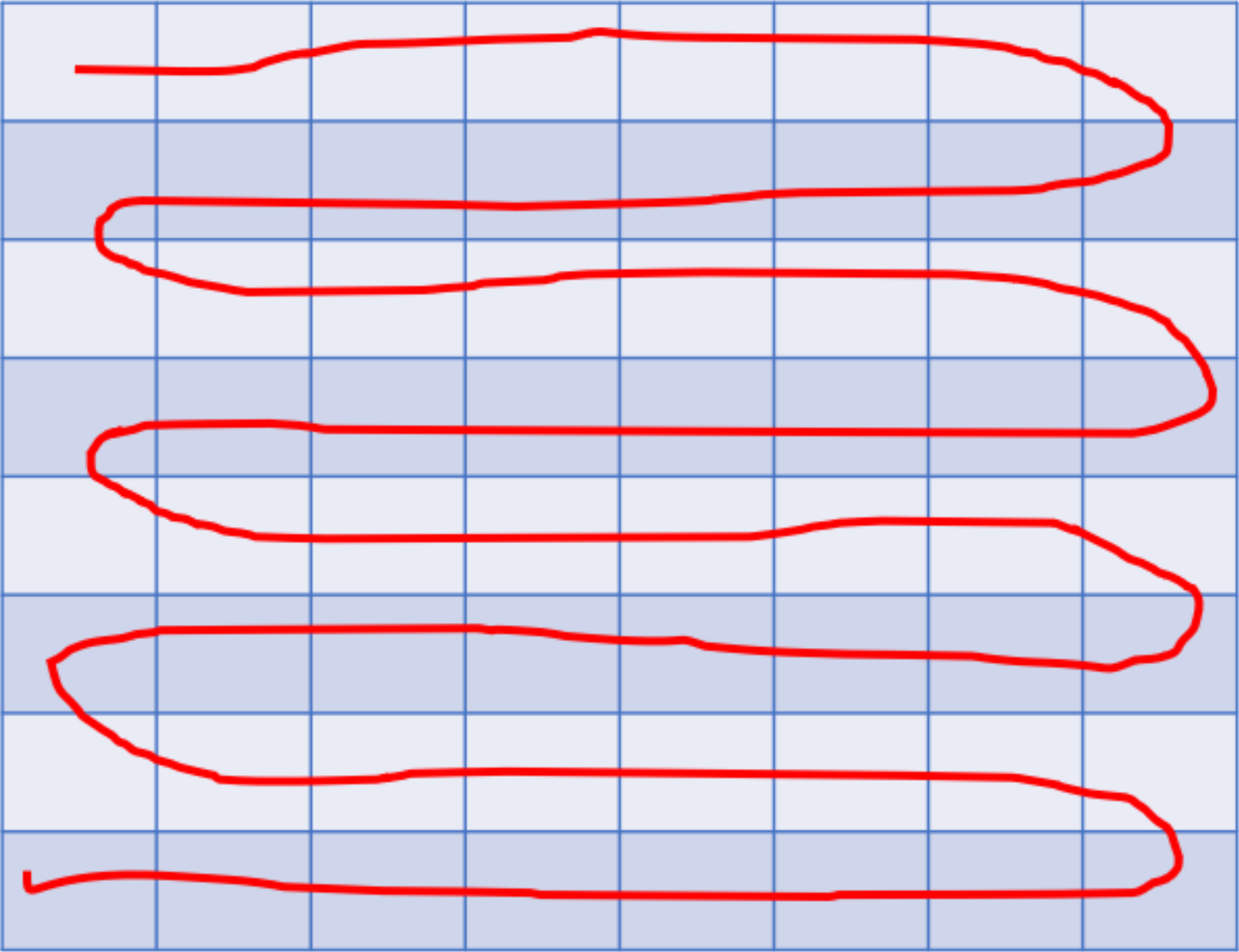
State		State_KS	State_OH	State_NJ
KS		1	0	0
OH	➡	0	1	0
NJ		0	0	1
OH		0	1	0

- Feature scaling

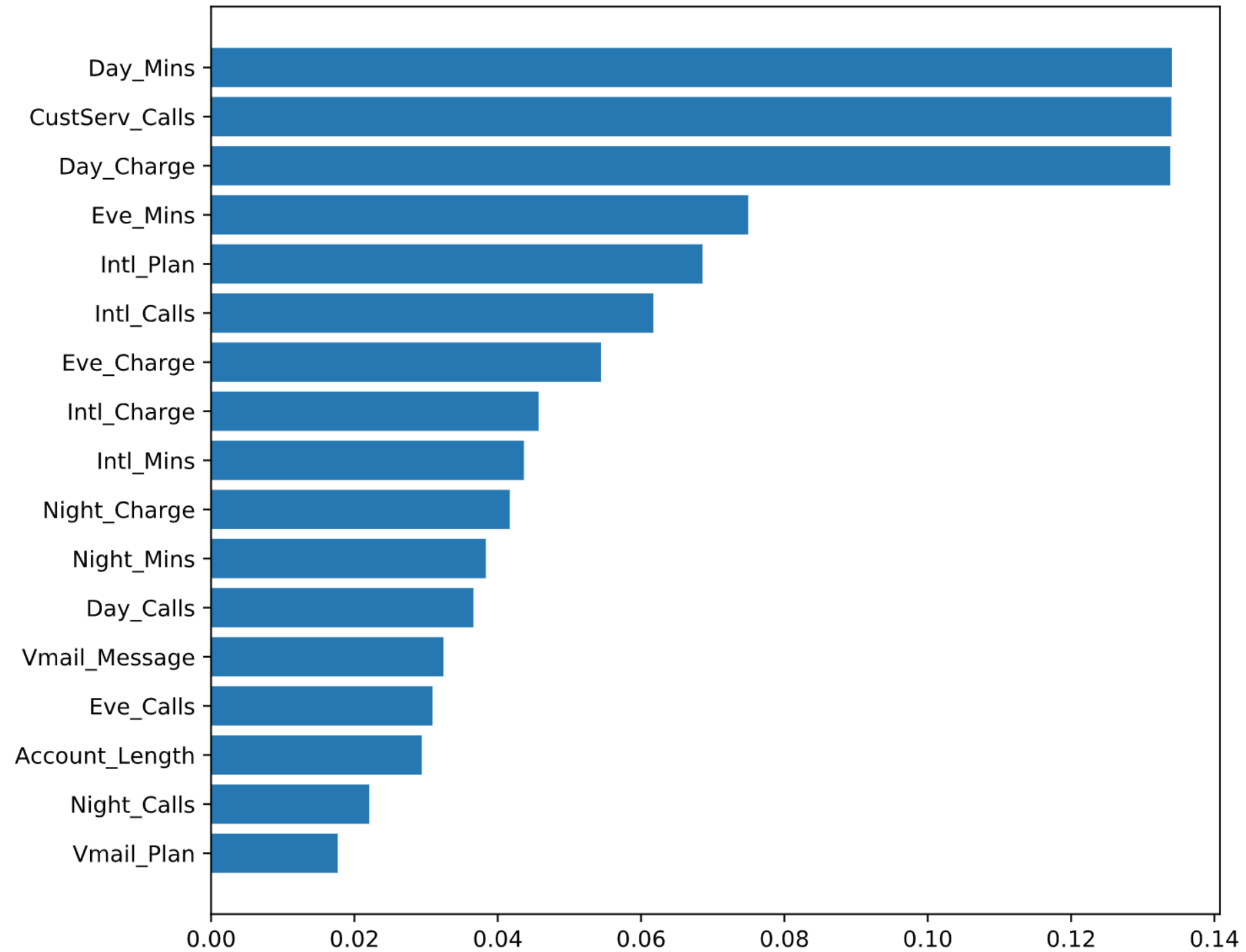
- Making predictions
- Training and testing sets



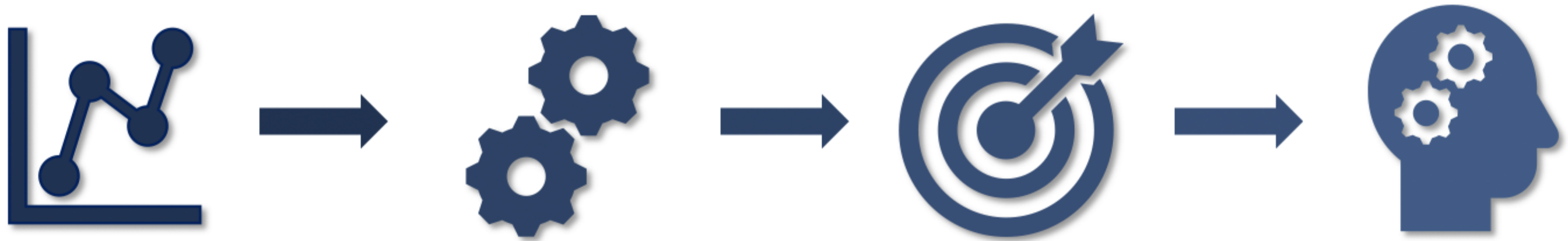
Start



End



Churn Workflow



Recommended next steps

- [Exploratory Data Analysis in Python](#)
- [Designing Machine Learning Workflows in Python](#)

Additional resources

- Kaggle competitions
- Coursera advanced business analytics specialization

Great Work!

PREDICTING CUSTOMER CHURN IN PYTHON