Classifier evaluation

IIT CS481 Spring

Text classification

- ❑ Pre-processing
  - Lower case
  - tokenization
  - punctuation removal
  - stop words removal
  - Stemming
  - lemmatization
  - POS tagging
  - Generating word cloud
- ❑ Feature engineering
  - CountVectorizer
  - TfidfVectorizer
- ❑ LogisticRegression classifier
- ❑ Train test split
- ❑ Interpretation
- ❑ **Evaluate classifier performance**
- ❑ **Parameter tuning**

Evaluation

❑**Evaluation matrix:**

- Confusion matrix
- Precision
- Recall
- F1
- Accuracy
- ROC AUC
- Precision recall curve

```
y_true = [0, 0, 0, 1, 1, 1, 1, 1]
y_pred = [0, 1, 0, 1, 0, 1, 0, 1]
```

❑**Parameter tuning**

- Kfold cross validation
- Grid search

Confusion matrix

	Actual pos	Actual neg
Predicted pos	tp (correctly predicted as pos)	fp (falsely predicted as pos)
Predicted neg	fn (falsely predicted as neg)	tn (correctly predicted as neg)

- FN: a person that actually got COVID but tested negative;
- FP: a person that don't get COVID but tested positive;

Confusion matrix

	Actual pos	Actual neg
Predicted pos	tp (correctly predicted as pos)	fp (falsely predicted as pos)
Predicted neg	fn (falsely predicted as neg)	tn (correctly predicted as neg)

```
y_true = [0, 0, 0, 1, 1, 1, 1, 1]
y_pred = [0, 1, 0, 1, 0, 1, 0, 1]
```

	Actual pos	Actual neg
Predicted pos	tp = 3	fp = 1
Predicted neg	fn = 2	tn = 2

$$\text{precision} = \frac{tp}{tp + fp}$$

	Actual pos	Actual neg
Predicted pos	tp (correctly predicted as pos)	fp (falsely predicted as pos)
Predicted neg	fn (falsely predicted as neg)	tn (correctly predicted as neg)

- Among all the samples predicted as pos, what's the fraction of tp samples?
- The ability of the classifier to accurately predict pos samples
- **Minimize false positives** increases precision

	Actual pos	Actual neg
Predicted pos	tp = 3	fp = 1
Predicted neg	fn = 2	tn = 2

3/(3+1)

$$\text{recall} = \frac{tp}{tp + fn}$$

	Actual pos	Actual neg
Predicted pos	tp (correctly predicted as pos)	fp (falsely predicted as pos)
Predicted neg	fn (falsely predicted as neg)	tn (correctly predicted as neg)

- Among all the actual pos samples, what's the fraction of tp samples?
- The ability of the classifier to find all positive samples
- **minimize false negatives** increases recall

	Actual pos	Actual neg
Predicted pos	tp = 3	fp = 1
Predicted neg	fn = 2	tn = 2

3/(3+2)

Precision VS recall

$$\text{precision} = \frac{tp}{tp + fp}$$

$$\text{recall} = \frac{tp}{tp + fn}$$

Pos

- How accurately could we predict for correct positive samples?
- Minimize FP increases precision

- How many correct positive samples do we find among all the positive samples?
- Minimize FN increases recall

F-measure

$$F_\beta = (1+\beta^2)\frac{\text{precision} \times \text{recall}}{\beta^2\text{precision} + \text{recall}}$$

F-beta score: the weighted mean of precision and recall (0~1)

F1: beta = 1

$$\text{precision} = \frac{tp}{tp + fp}$$

$$\text{recall} = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

	Actual pos	Actual neg
Predicted pos	tp = 3	fp = 1
Predicted neg	fn = 2	tn = 2

- Precision = 3 / (3+1)
- Recall = 3/(3+2)
- F1 = 2*(3/4*3/5)/(3/4+3/5)

accuracy

	Actual pos	Actual neg
Predicted pos	**tp** (correctly predicted as pos)	fp (falsely predicted as pos)
Predicted neg	fn (falsely predicted as neg)	**tn** (correctly predicted as neg)

The fraction of the correct predictions:

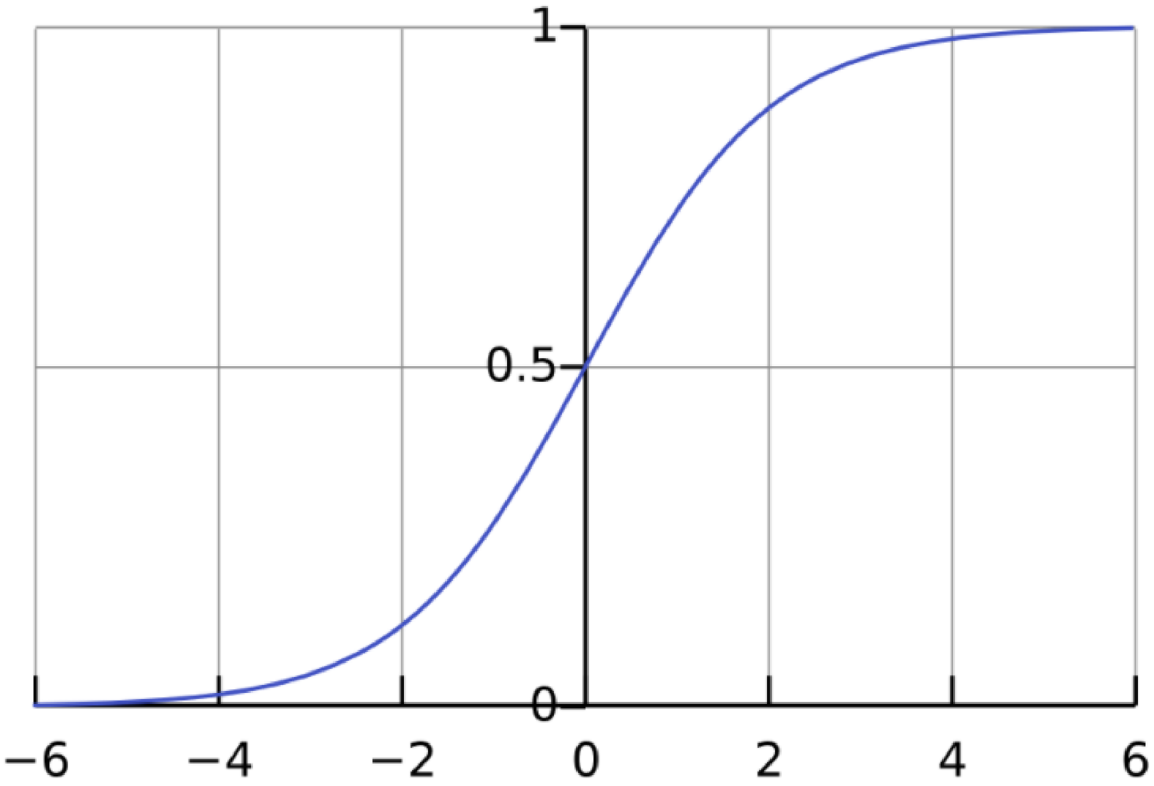
$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

$$\texttt{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

Summary

- **Precision and recall** provide two ways to summarize the errors made for the **positive class** (FP, FN).
- **F-measure** provides a single score that summarizes the precision and recall.
- **Accuracy** summarizes the correct predictions for both positive and negative classes.

Interpret probability with different thresholds



Probability of a sample belonging to each class:

- Prob>=0.5 → pos
- Prob<0.5 → neg

Interpret probability using other thresholds:

- Prob>=0.4 → pos
- Prob<0.4 → neg
- Want less false neg errors (COVID)

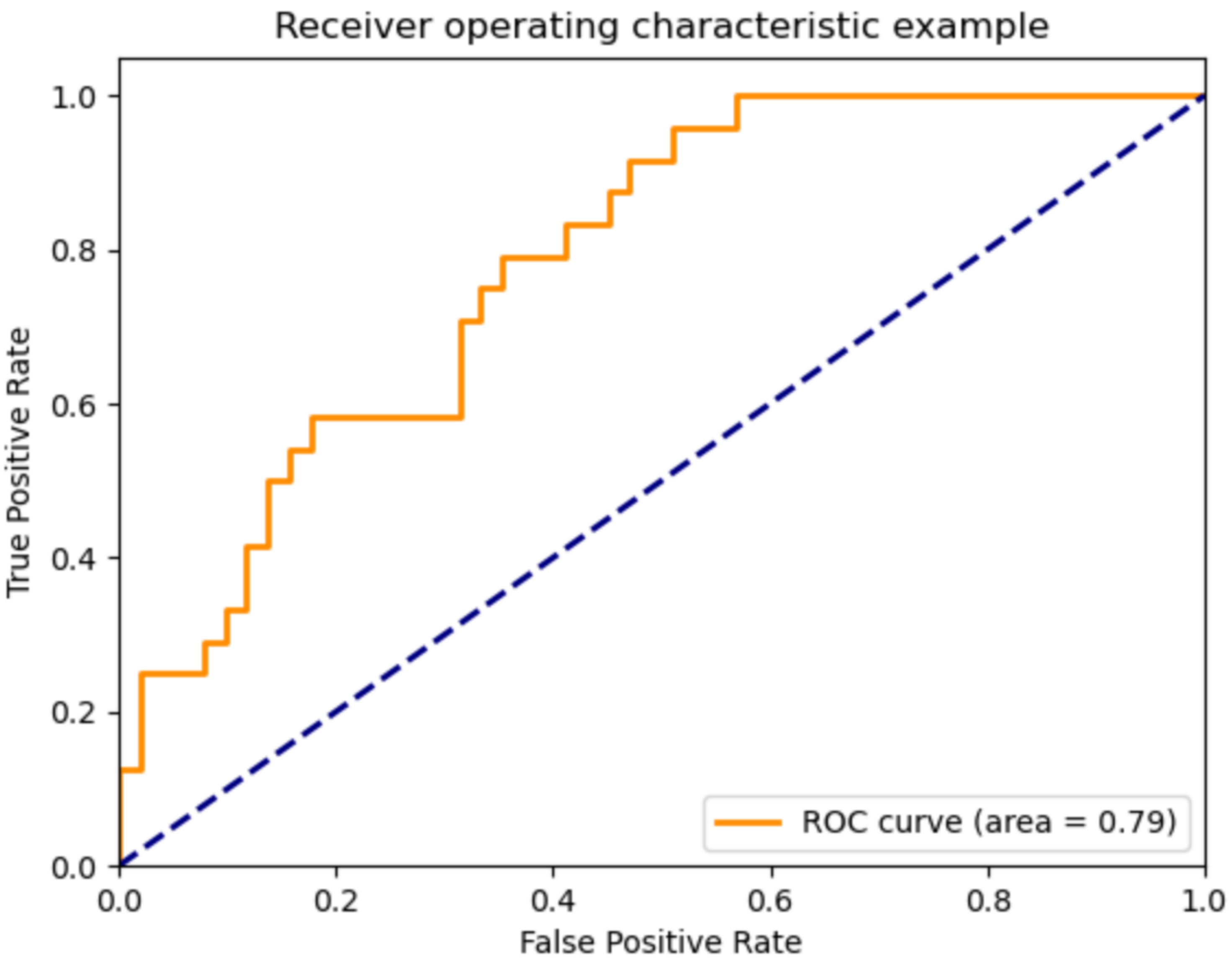
- Prob>=0.6 → pos
- Prob<0.6 → neg
- Want less false pos errors (crime conviction)

ROC curve

Precision-recall curve

ROC curve

Correctly predicted as pos



Falsely predicted as pos

ROC curve

	Actual pos	Actual neg
Predicted pos	tp	fp
Predicted neg	fn	tn

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{N}} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} = 1 - \mathrm{TNR}$$

ROC curve

```
y_true = np.array([0, 0, 1, 1])
y_scores = np.array([0.1, 0.4, 0.35, 0.8])
```

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}}$$

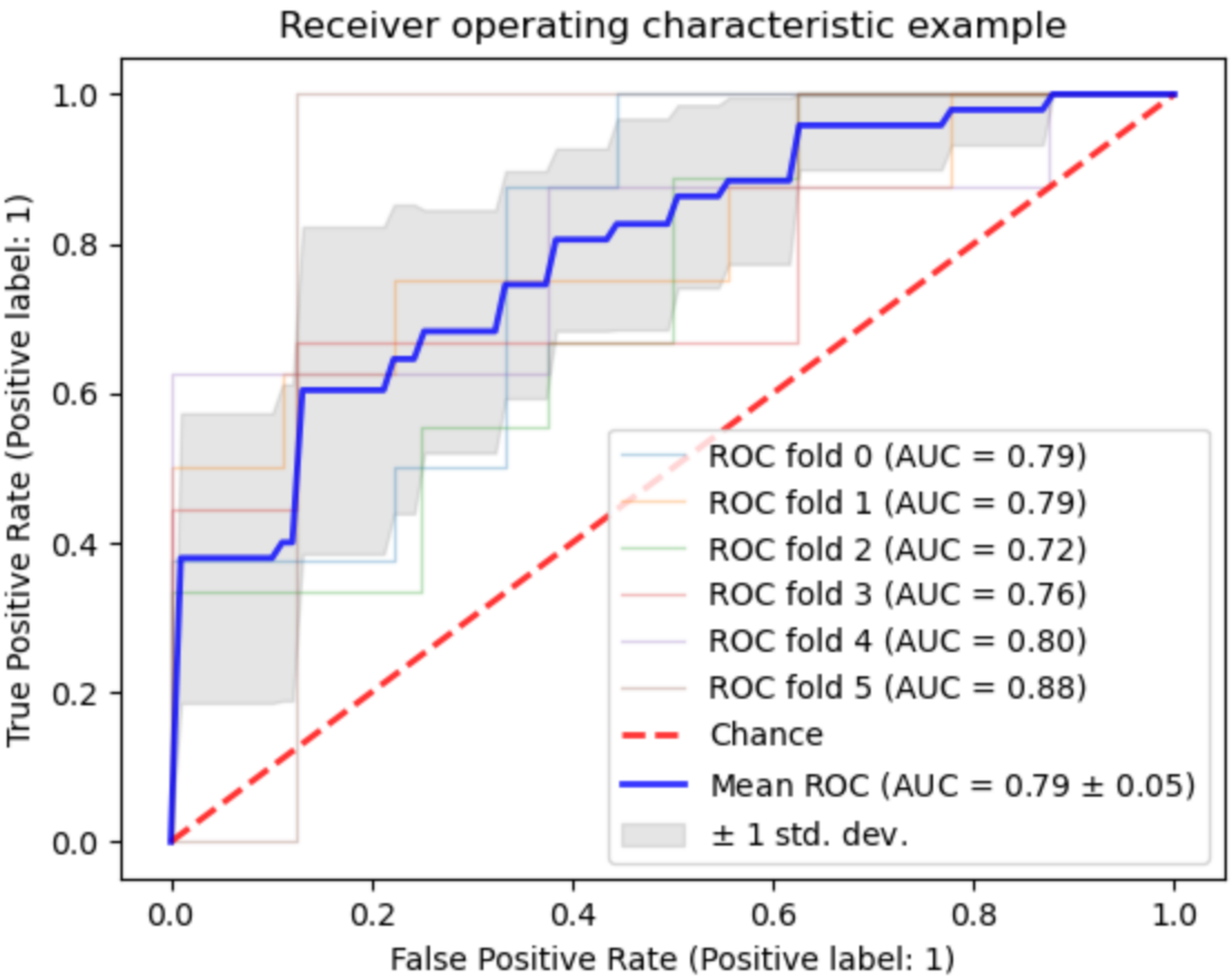
$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{N}}$$

truth	threshold	TPR	FPR
	>=1.8	0/2	0/2
1	>=0.8	1/2	0/2
0	>=0.4	1/2	1/2
1	>=0.35	2/2	2/2
0	>=0.1	2/2	2/2

```
FP rate: [0.   0.   0.5 0.5 1. ]
TP rate: [0.   0.5 0.5 1.   1. ]
thresholds: [1.8  0.8  0.4  0.35 0.1 ]
```

ROC AUC(Area Under the Curve)

Correctly predicted as pos



an aggregate measure of model performance across all possible classification thresholds

Falsely predicted as pos

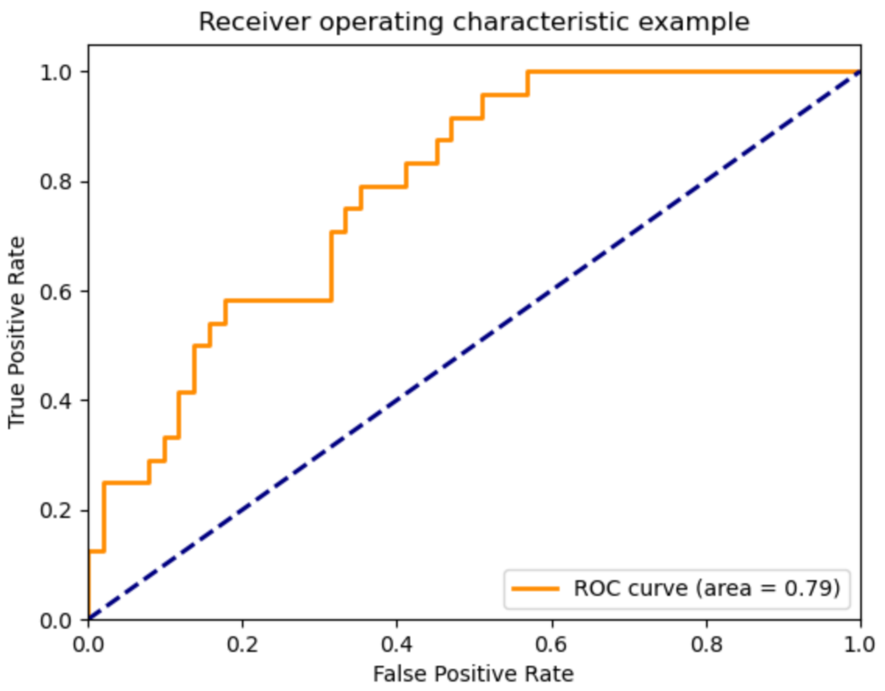
Model Evaluation

	Actual pos	Actual neg
Predicted pos	tp	fp
Predicted neg	fn	tn

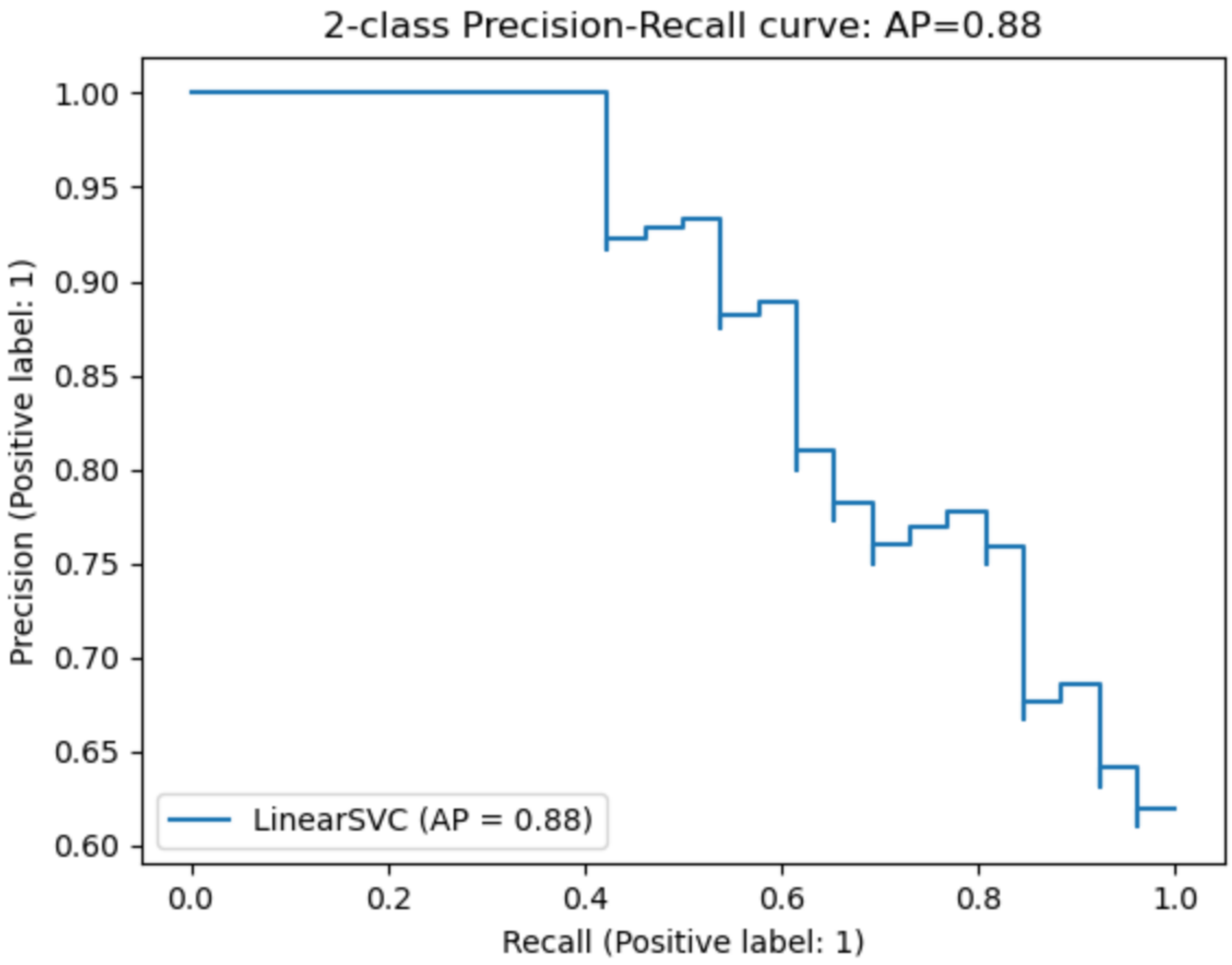
$$\text{precision} = \frac{tp}{tp + fp}$$

$$\text{recall} = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$



Precision_recall_curve



Perfect model towards (1,1)

Precision_recall_curve

```
y_true = np.array([0, 0, 1, 1])
y_scores = np.array([0.1, 0.4, 0.35, 0.8])
```

$$\text{precision} = \frac{tp}{tp + fp}$$

$$\text{recall} = \frac{tp}{tp + fn}$$

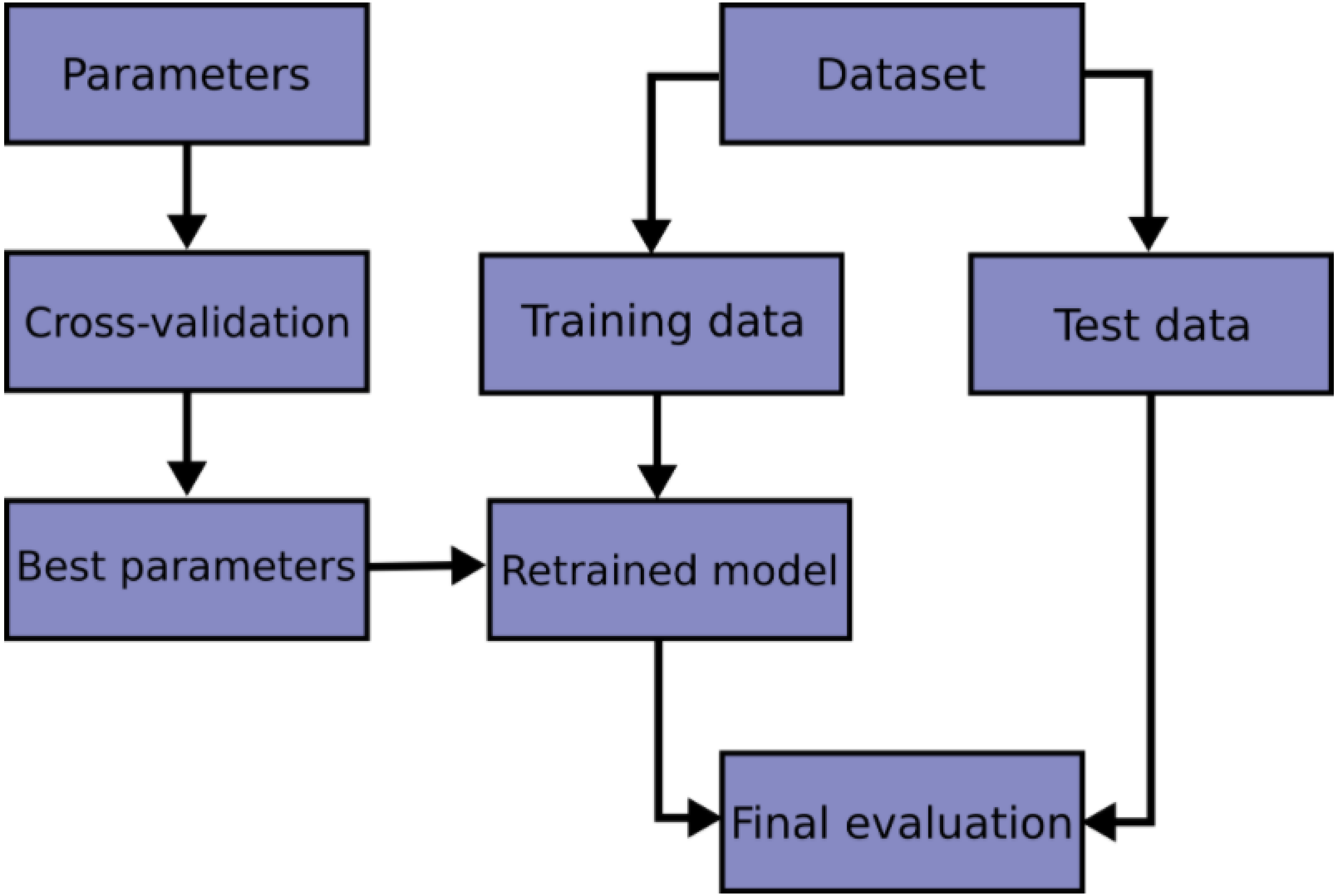
		Precision	Recall
1	>=0.8	1/1	1/2
0	>=0.4	1/2	1/2
1	>=0.35	2/3	2/2
0	0.1	2/4	2/2

```
>>> precision
array([0.66666667, 0.5       , 1.        , 1.        ])
>>> recall
array([1. , 0.5, 0.5, 0. ])
>>> thresholds
array([0.35, 0.4 , 0.8 ])
```

ROC vs Precision-Recall Curves

- Summarizes model performance using **different probability thresholds**
- ROC curves should be used when there are roughly **equal numbers of observations** for each class
- Precision-Recall curves should be used when there is a moderate to **large class imbalance** (when we are interested in the pos class and there's only a few pos samples)

Parameter tuning



K-fold cross validation