Understanding and Overcoming Common Problems in Data Modeling

Janani Ravi

CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Identifying and mitigating common biases

Overfitted models

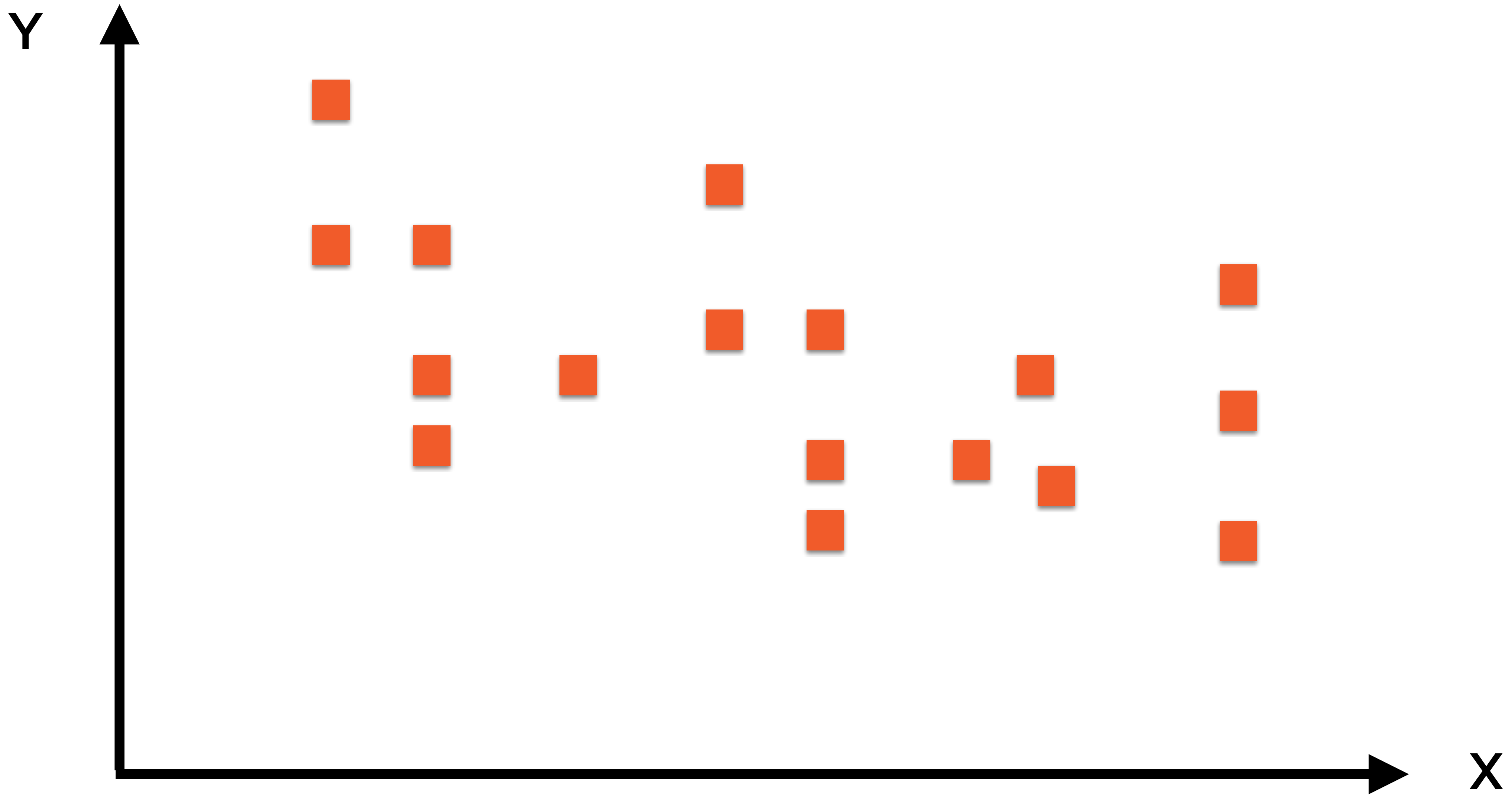
Bias/variance trade-off

Evaluating models using accuracy, precision, and recall

Understanding the ROC curve

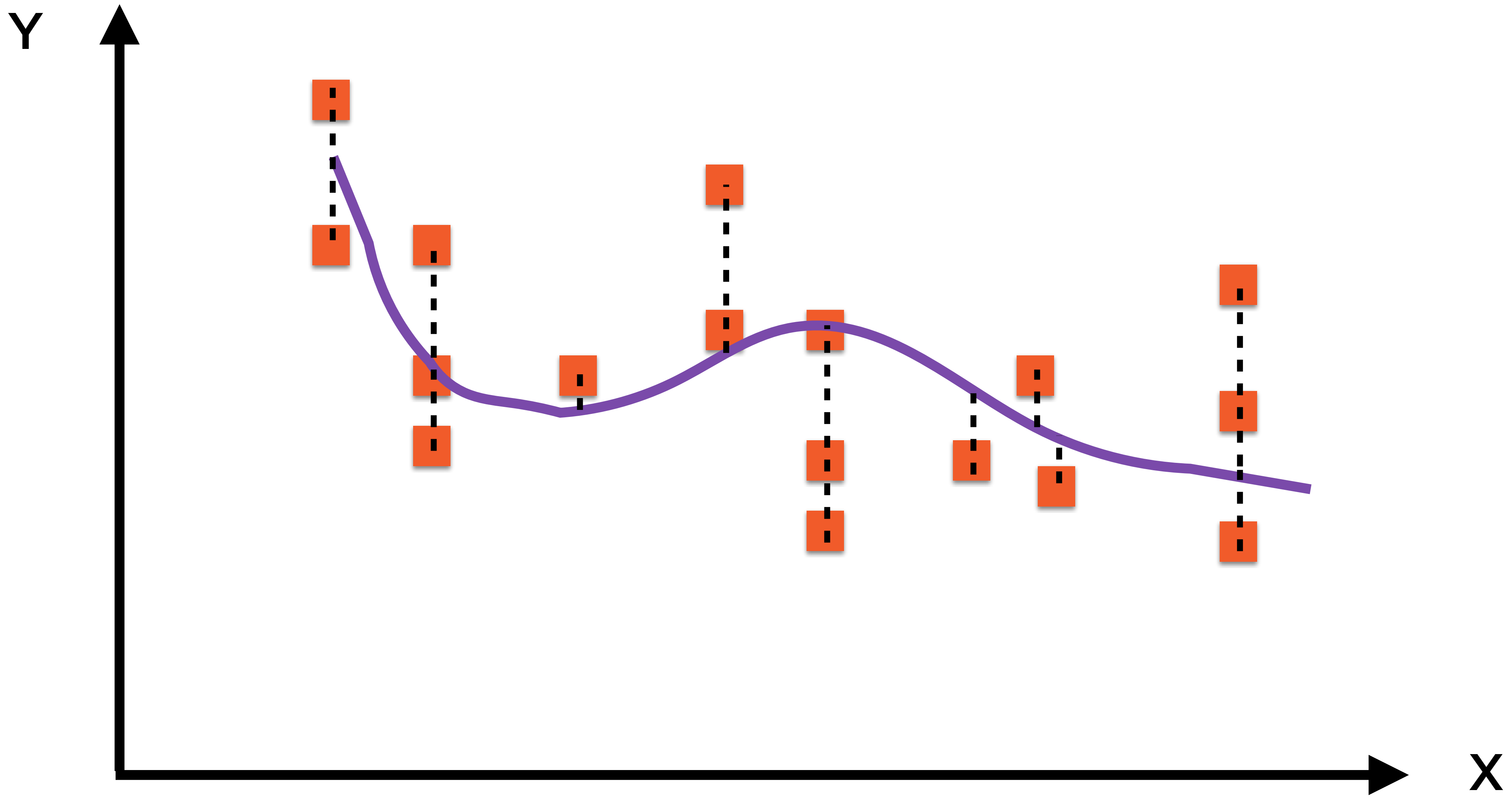
Overfitting and Preventing Overfitting

Connecting the Dots



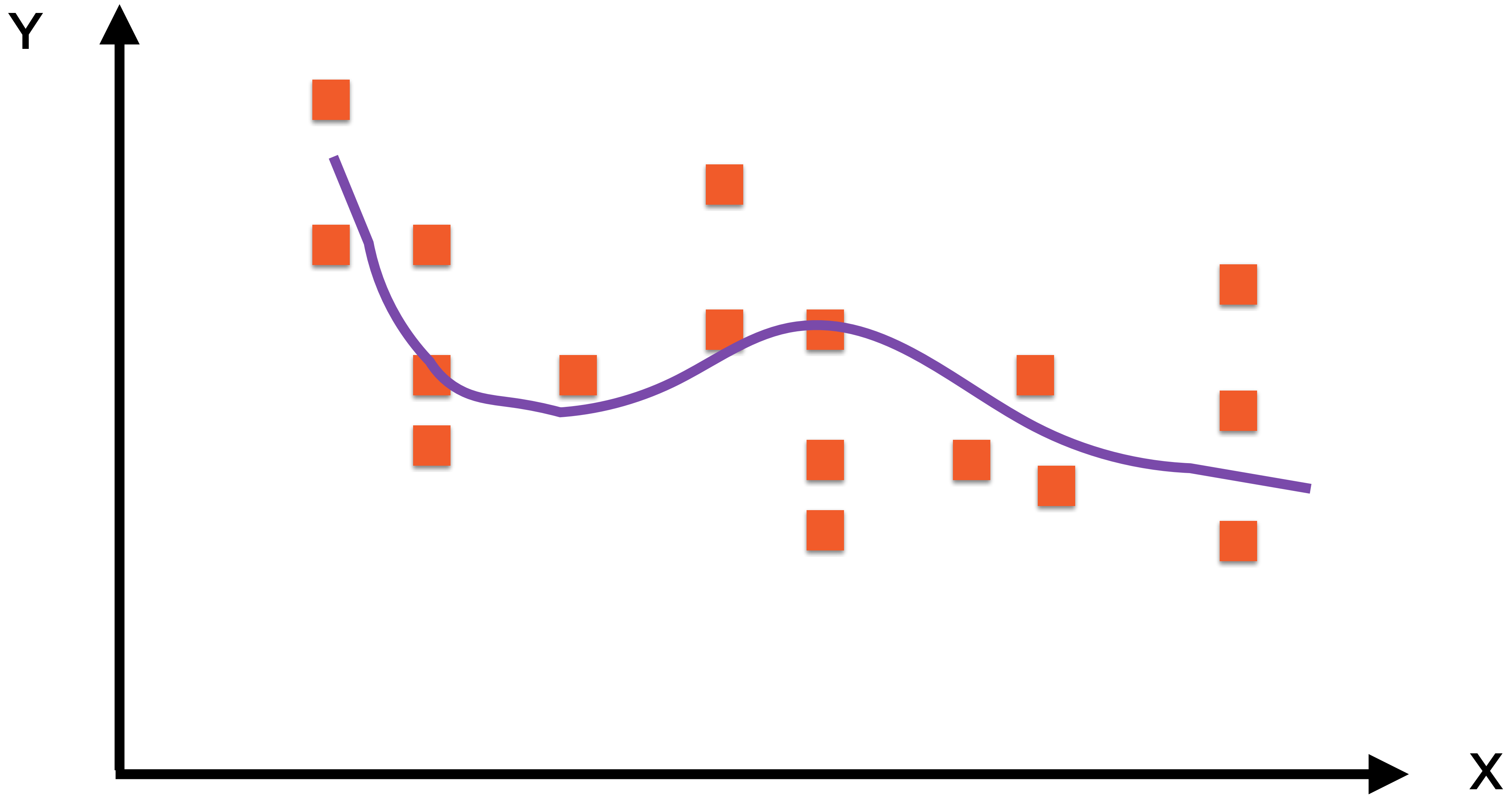
Challenge: Find the "best" curve through these points

Good Fit?



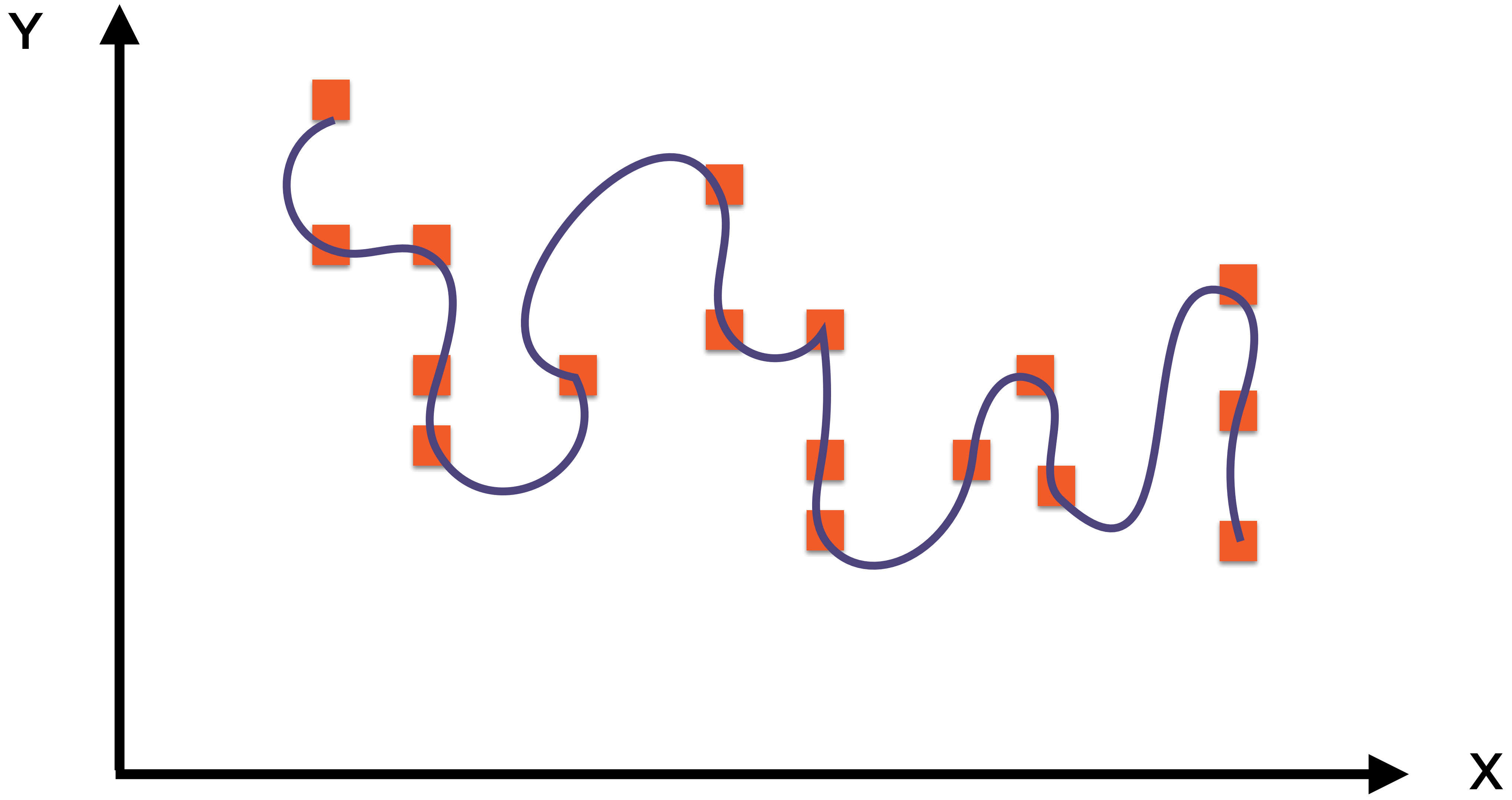
A curve has a "good fit" if the distances of points from the curve are small

Connecting the Dots



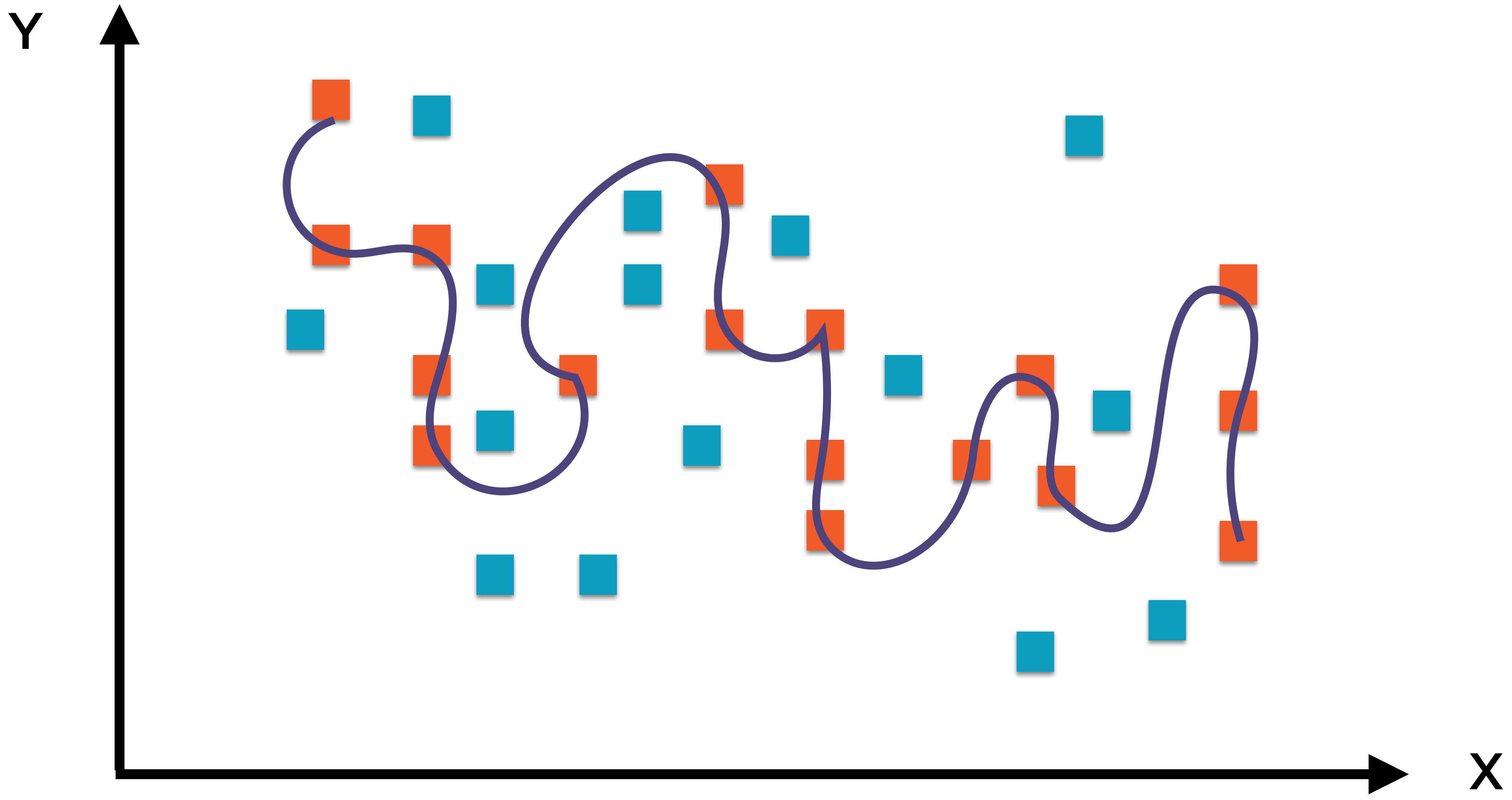
We could draw a pretty complex curve

Connecting the Dots



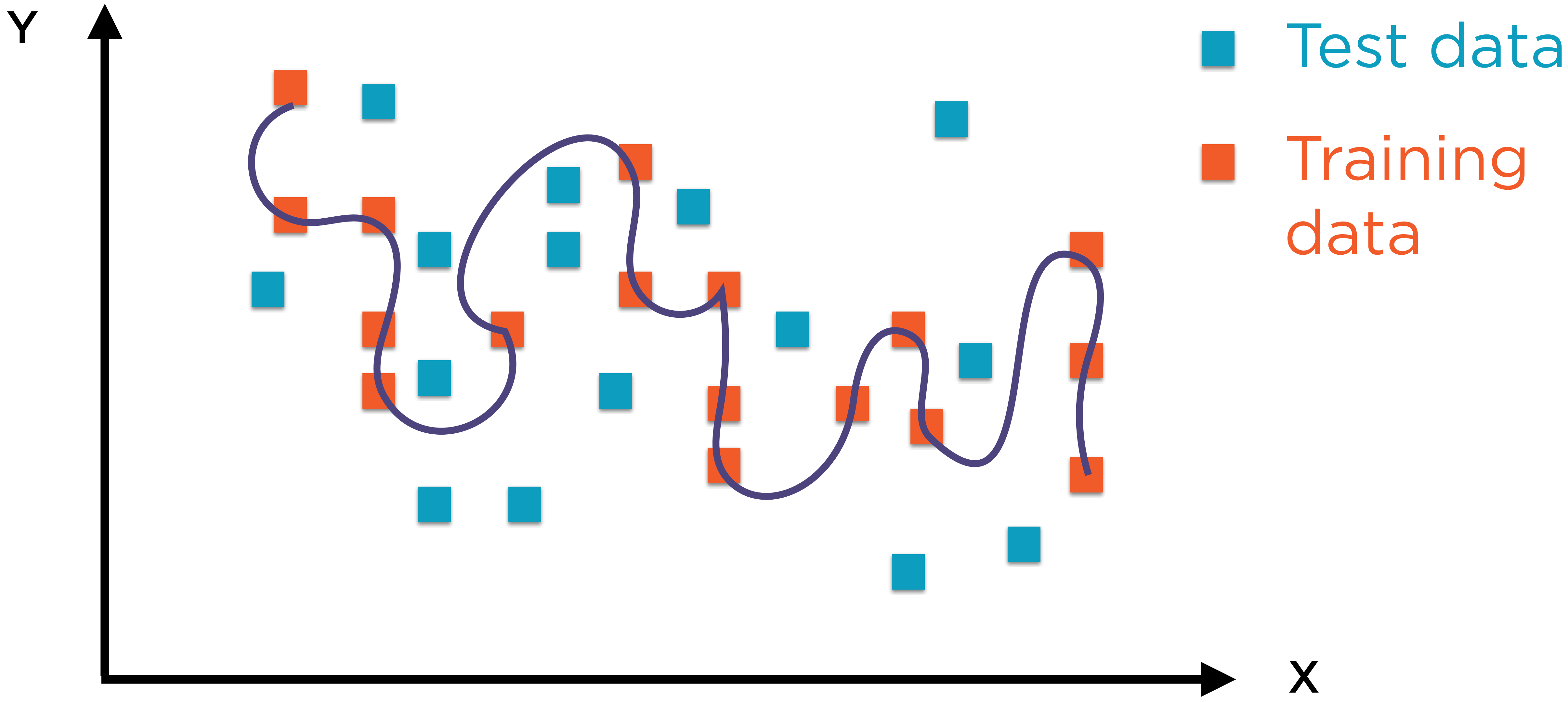
We can even make it pass through every single point

Connecting the Dots



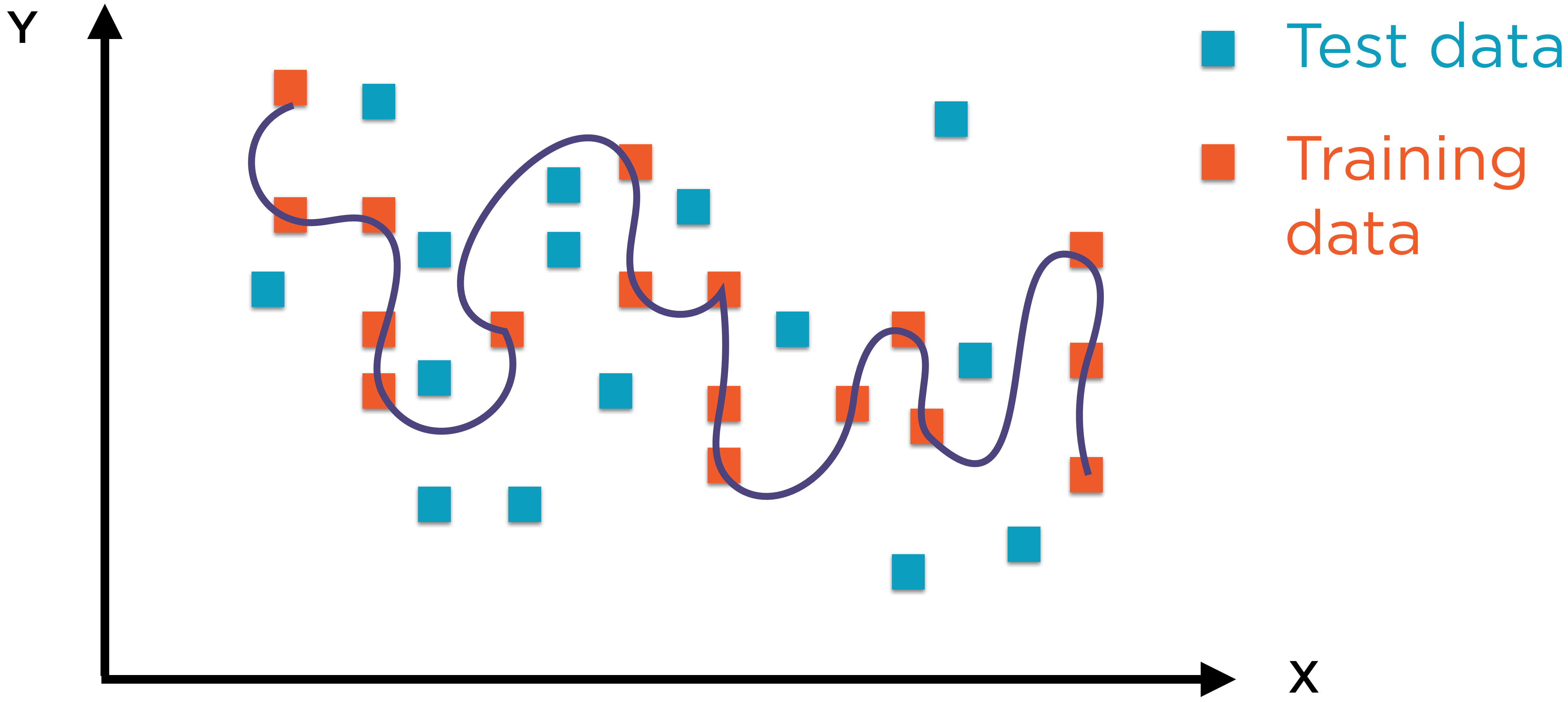
But given a new set of points, this curve might perform quite poorly

Connecting the Dots



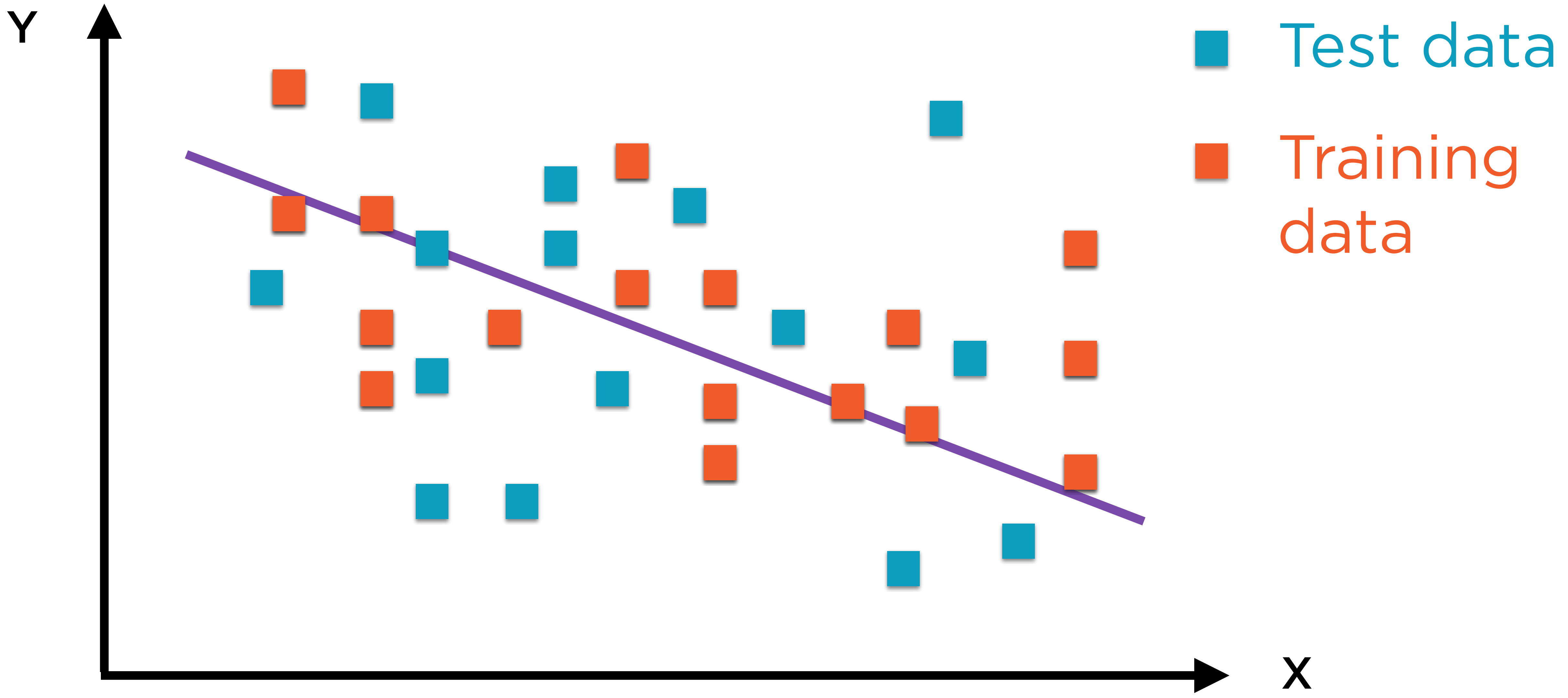
The original points were “training data”, the new points are “test data”

Overfitting



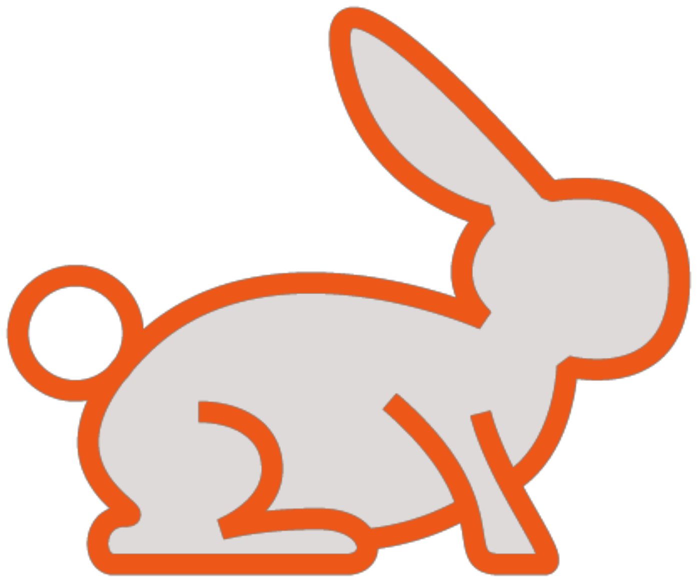
Great performance in training, poor performance in real usage

Connecting the Dots



A simple straight line performs worse in training, but better with test data

Overfitting



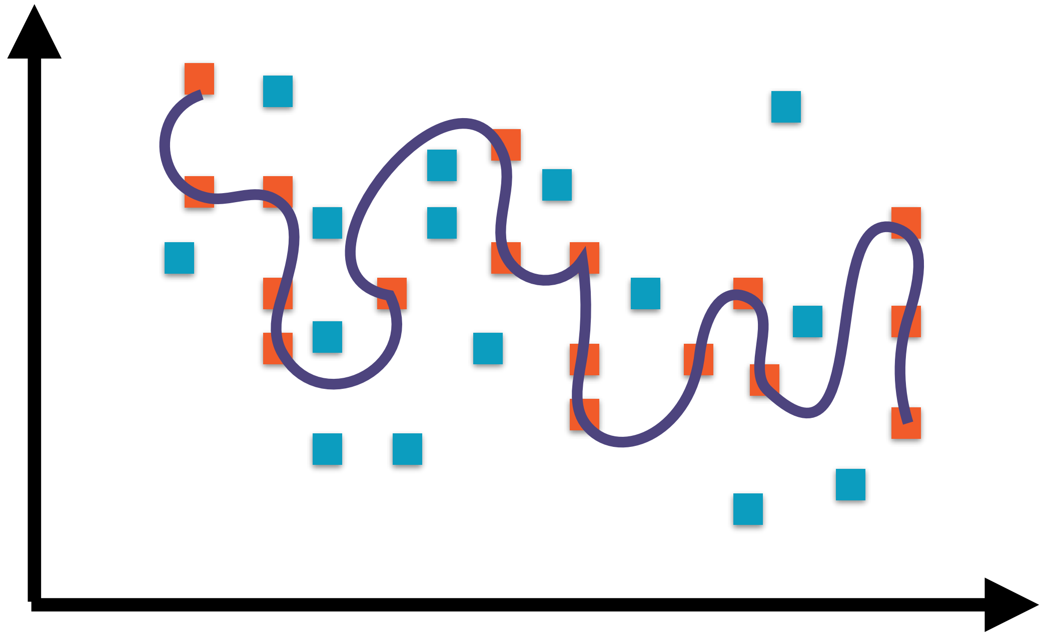
Low Training Error

Model does very well in training…

High Test Error

…but poorly with real data

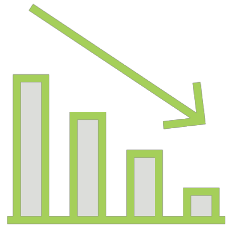
Cause of Overfitting



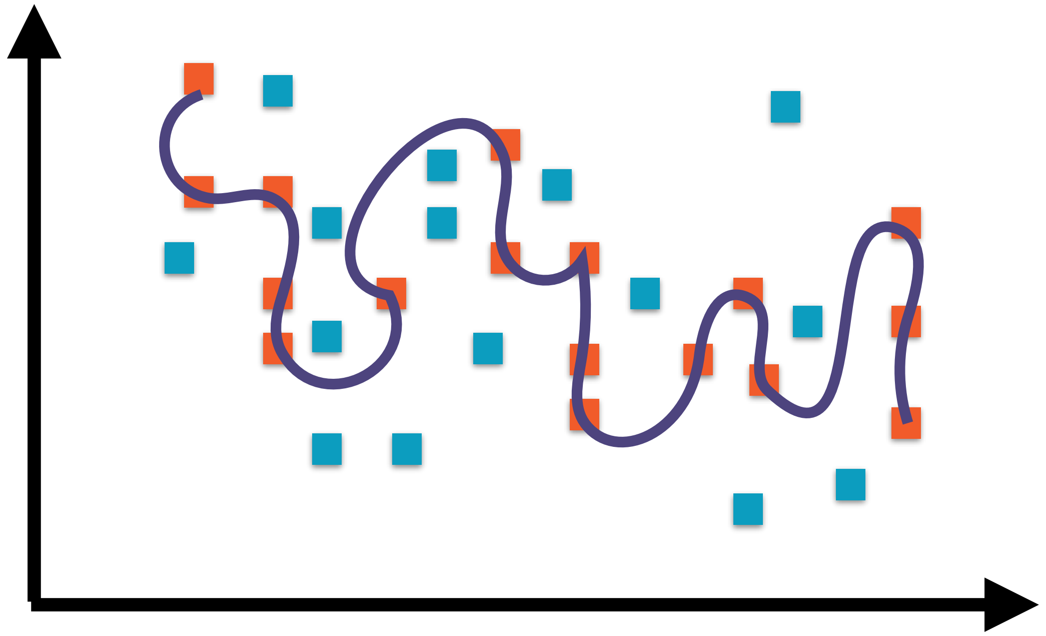
Sub-optimal choice in the bias-variance trade-off

An overfitted model has:

- high variance error
- low bias error

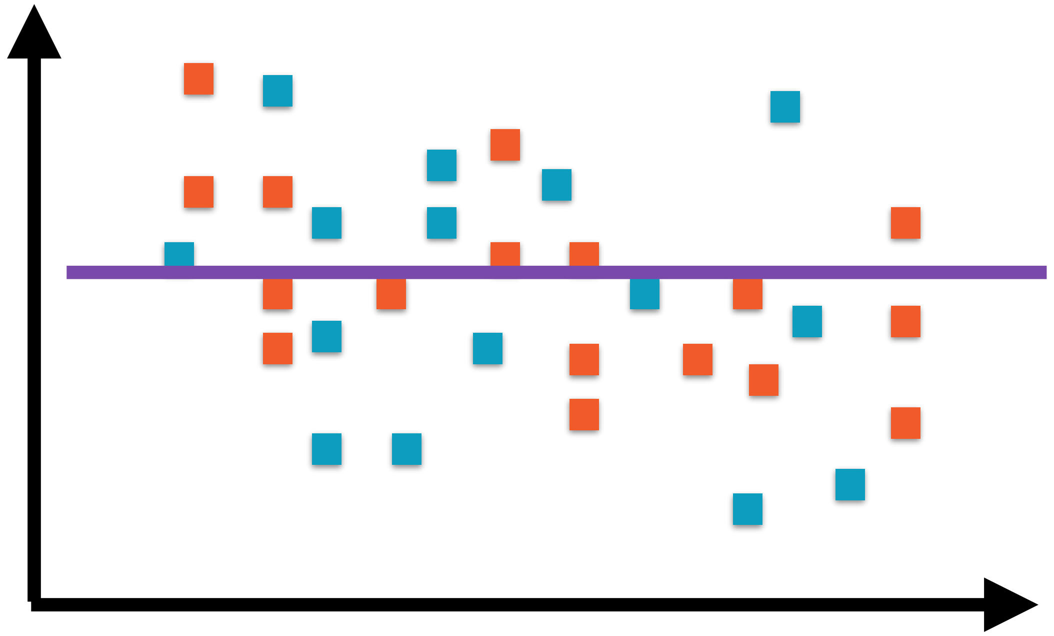


Bias



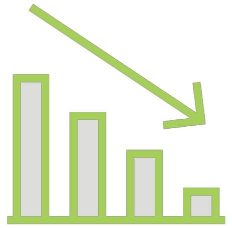
Low bias

Few assumptions about the underlying data

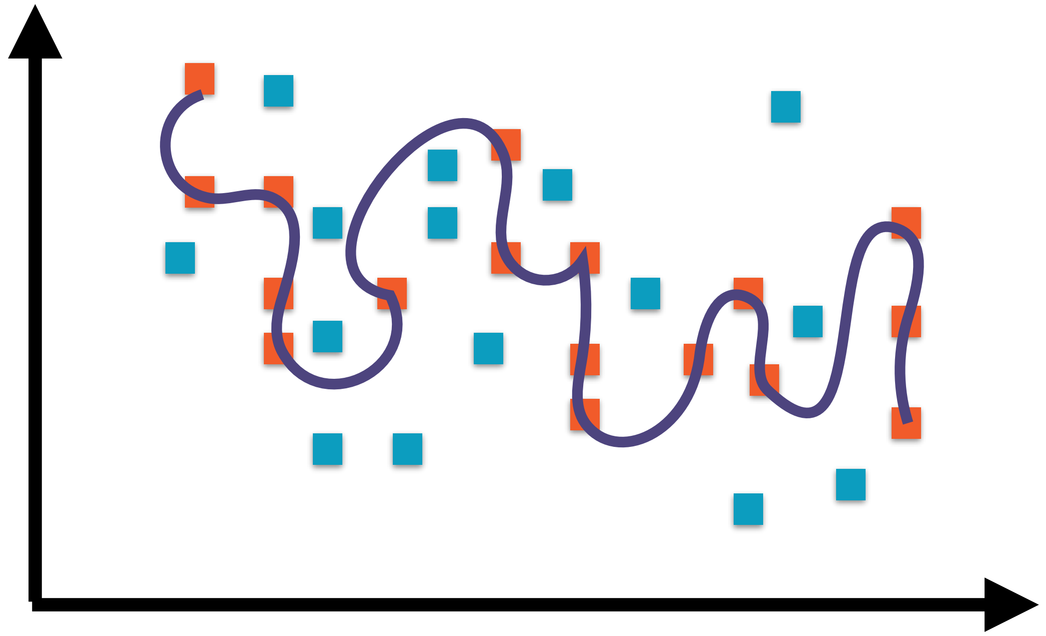


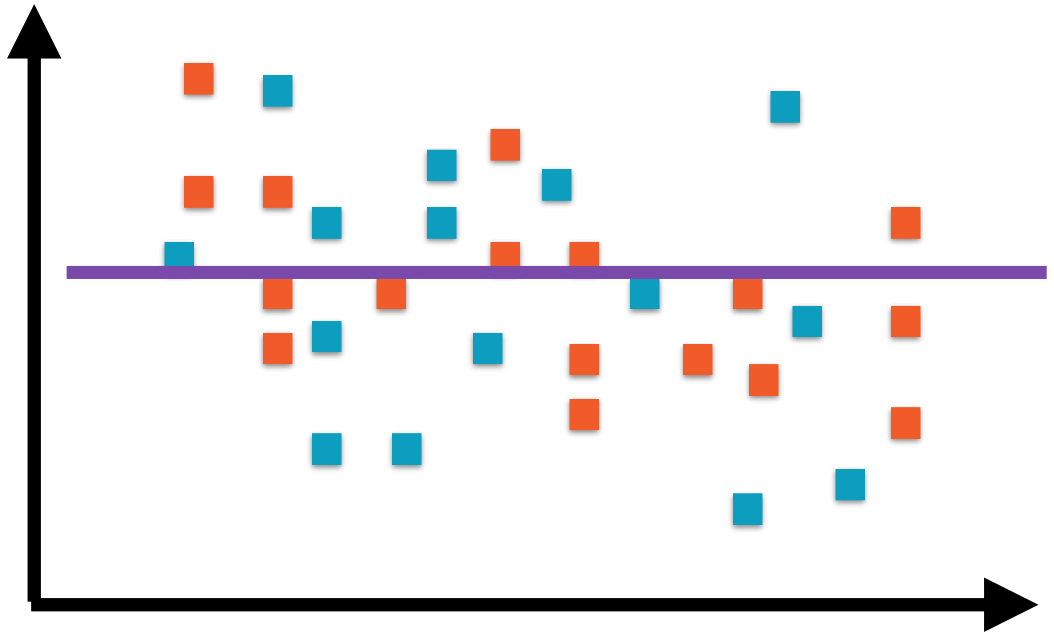
High bias

More assumptions about the underlying data



Bias





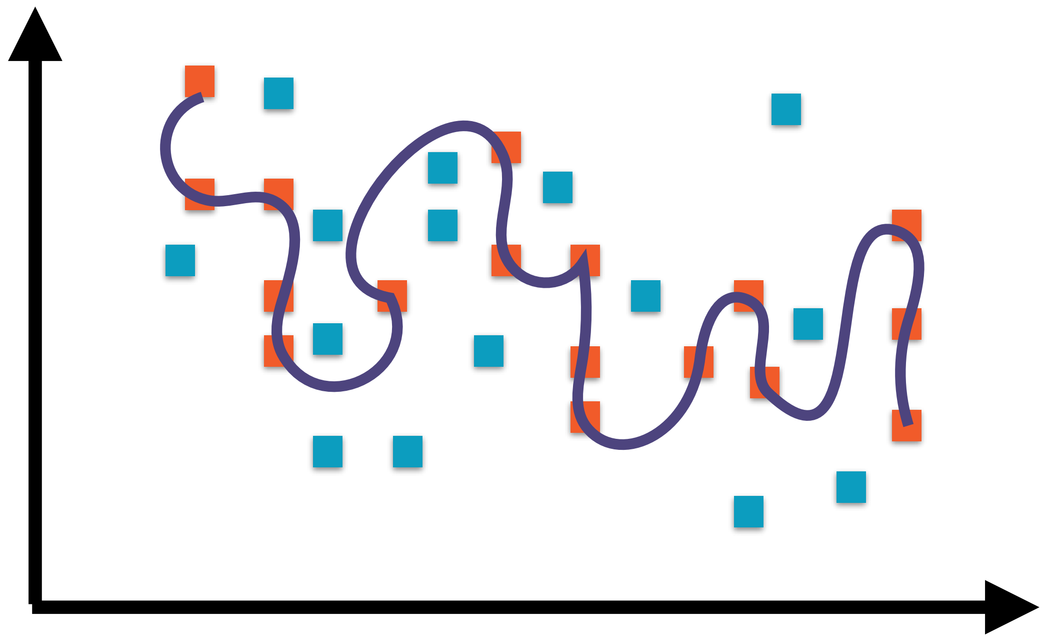
Model too complex

Training data all-important, model parameter counts for little

Model too simple

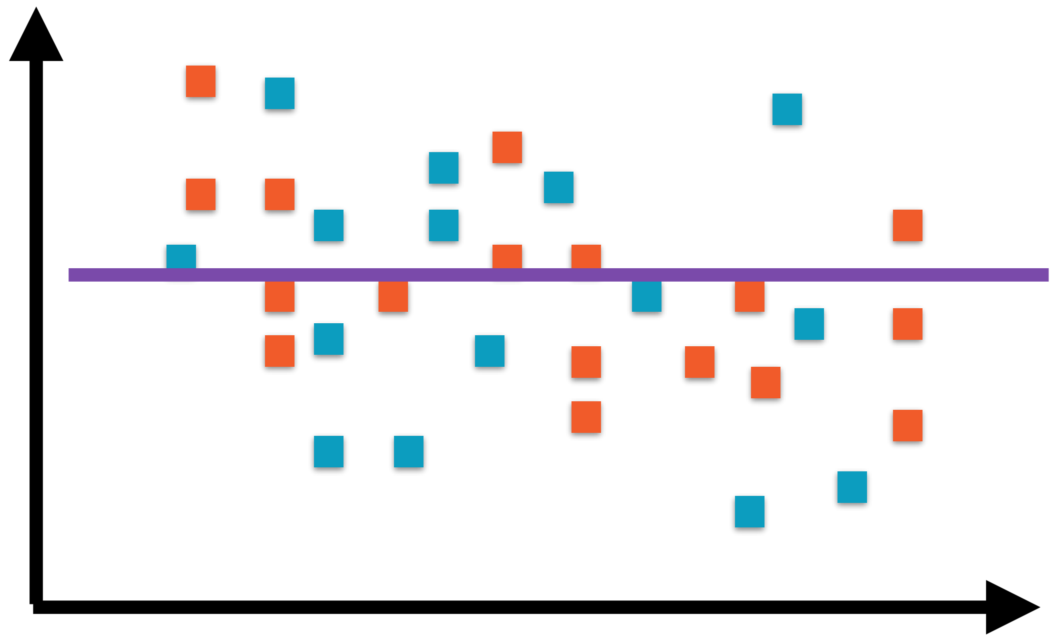
Model parameter all-important, training data counts for little

Variance



High variance

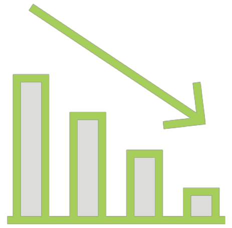
The model changes significantly when training data changes

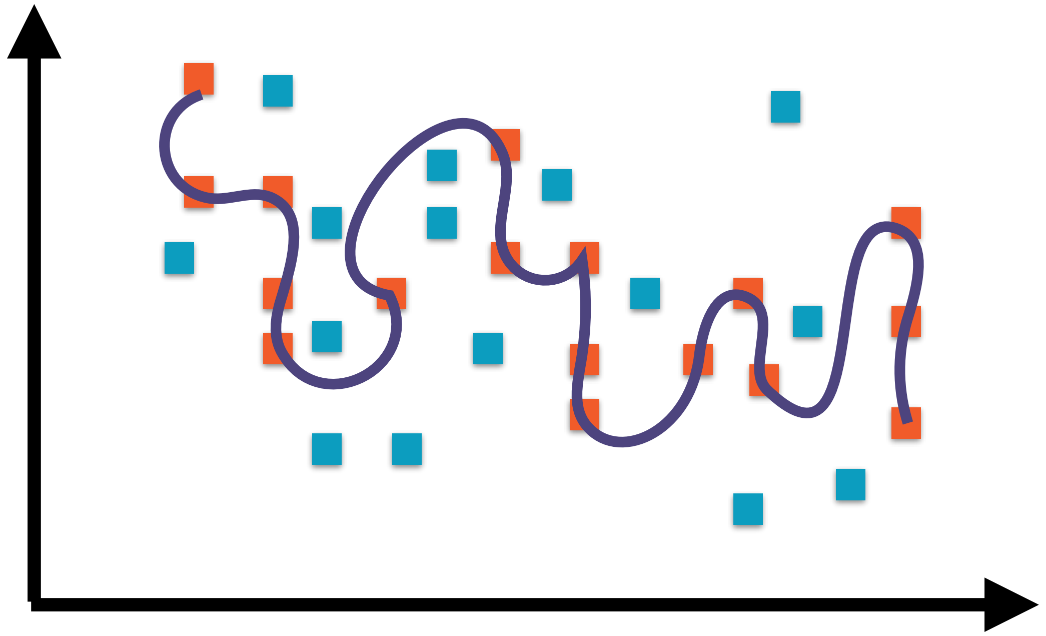


Low variance

The model doesn't change much when the training data changes

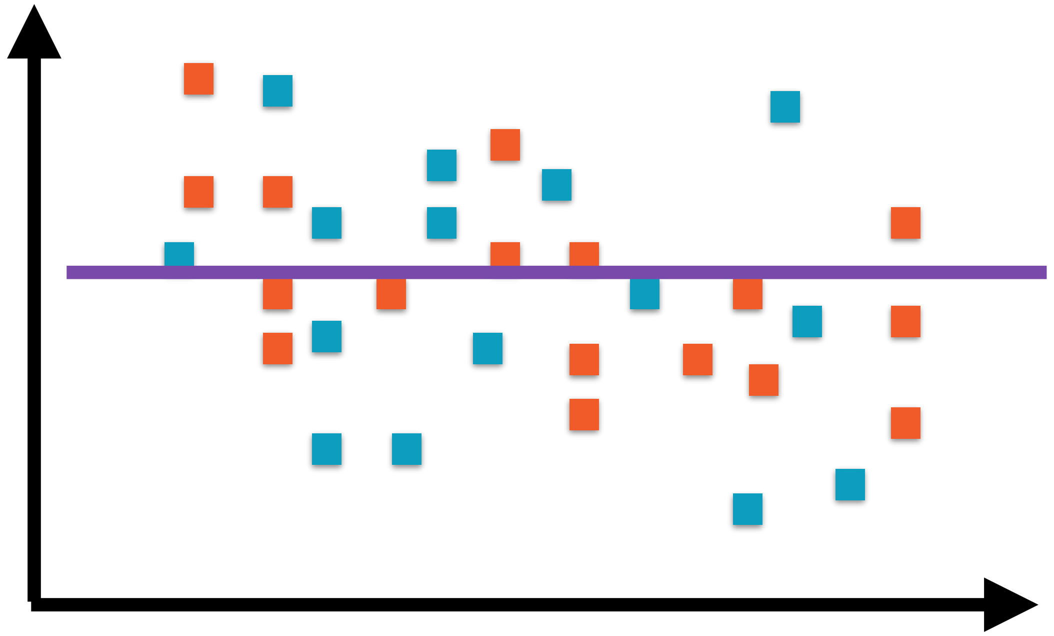
Variance





Model too complex

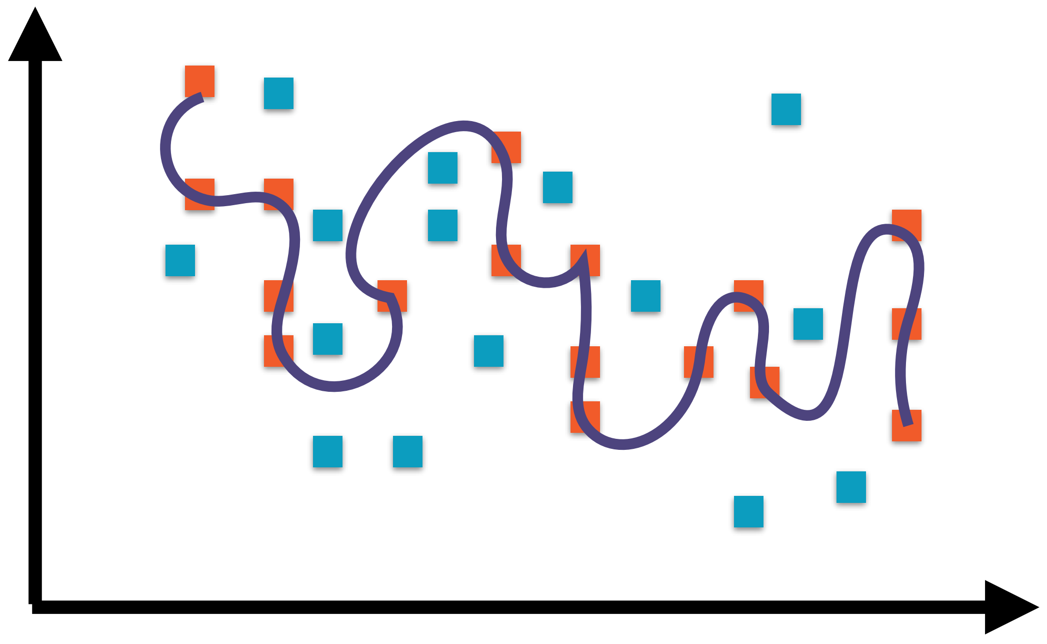
Model varies too much with changing training data



Model too simple

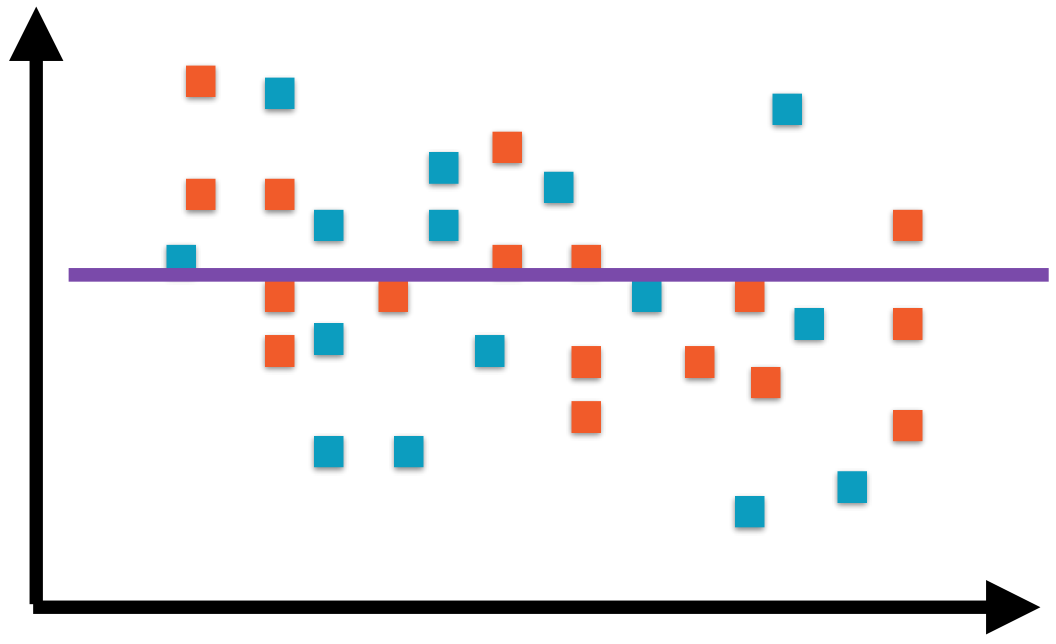
Model not very sensitive to training data

Bias-Variance Trade-off



Model too complex

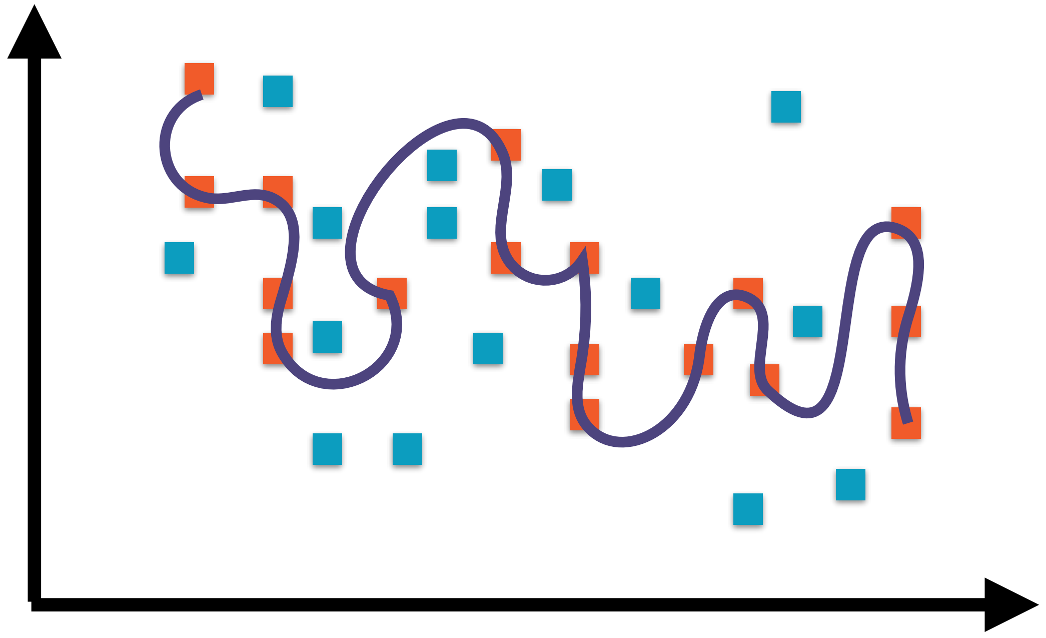
High variance error



Model too simple

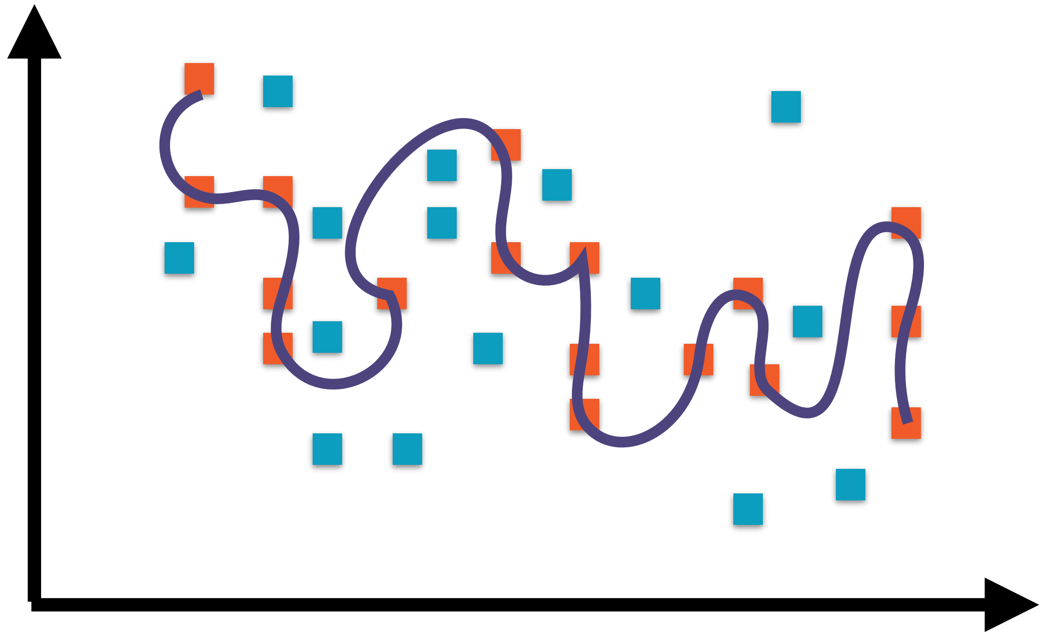
High bias error

Bias-Variance Trade-off



- **High-bias algorithms: simple parameters**
    - Regression
- **High-variance algorithms: complex parameters**
    - Decision trees
    - Dense neural networks

Preventing Overfitting



Regularization

Cross-validation

Ensemble learning

Dropout

Regularization

Penalize complex models

Add penalty to objective function

Forces optimizer to keep it simple

Cross-Validation

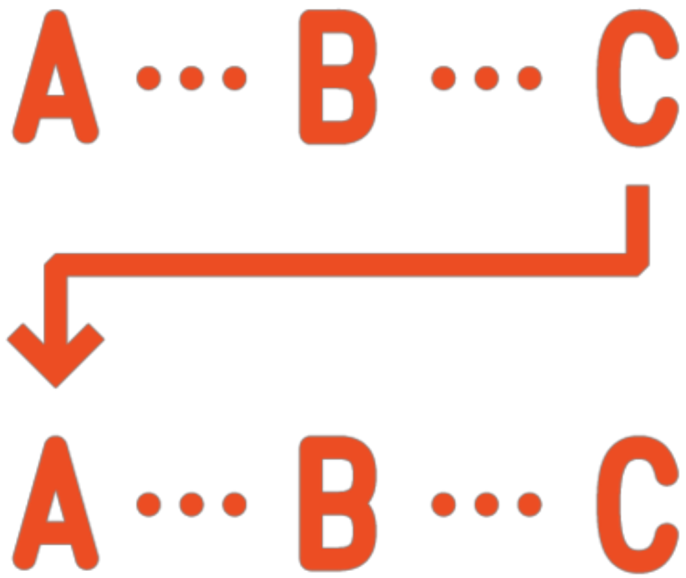
Distinct training and validation phases

Train different models (with training data only)

Select model that does best on validation data

“Hyperparameter tuning”

Ensemble Learning



Construct several models and then combine their outputs

Each individual model could be a relatively weak learner

Combining many weak learners can yield a strong learner

Dropout

Specialized technique used in training deep learning

Deep learning models consist of layers of interconnected neurons

Dropout involves intentionally turning off some neurons at random

Each iteration during training thus has subtly different architecture

Accuracy, Precision, Recall

The most ground-breaking applications of ML in recent years have been to classification problems

Accuracy

Compare predicted and actual labels

More matches = higher accuracy

High accuracy is good, but...

An algorithm might have high accuracy but still be a poor machine learning model

Its predictions are **useless**

All-is-well Binary Classifier

Here, accuracy for rare cancer may be 99.9999%, but…

Accuracy

Some labels maybe much more common/rare than others

Such a dataset is said to be skewed

Accuracy is a poor evaluation metric here

Confusion Matrix

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	**10 instances**	**4 instances**
No Cancer	**5 instances**	**1000 instances**

Confusion Matrix

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10	4
No Cancer	5	1000

True Positive

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4
No Cancer	5	1000

Actual Label = Predicted Label

False Positive

Actual Label \ Predicted Labels	Cancer	No Cancer
Cancer	10	4
No Cancer	5 FP	1000

Actual Label ≠ Predicted Label

True Negative

Predicted Labels

Actual Label	Cancer	No Cancer
Cancer	10	4
No Cancer	5	1000 TN

Actual Label = Predicted Label

False Negative

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10	4 **FN**
No Cancer	5	1000

Actual Label ≠ Predicted Label

Confusion Matrix

Actual Label \ Predicted Labels	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Accuracy

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Accuracy

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Actual Label = Predicted Label

Accuracy

Predicted Labels

Actual Label	Cancer	No Cancer
Cancer	TP 10	FN 4
No Cancer	FP 5	TN 1000

$$\text{Accuracy} = \frac{TP + TN}{\text{Num Instances}} = \frac{1010}{1019} = 99.12\%$$

Accuracy

Accuracy = 99.12%

Classifier gets it right 99.12% of the time

But...

Accuracy

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

People on chemotherapy, radiation when not required

Accuracy

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Cancer not detected, no treatment prescribed

Accuracy is not a good metric to evaluate whether this model performs well

Precision

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Precision

Predicted Labels

Actual Label	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Precision = Accuracy when classifier flags cancer

Precision

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{10}{15} = 66.67\%$$

Precision

Precision = 66.67%

1 in 3 cancer diagnoses is incorrect

Recall

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Recall

Actual Label	Predicted Labels: Cancer	Predicted Labels: No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

Recall = Accuracy when cancer actually present

Recall

Predicted Labels

Actual Label

	Cancer	No Cancer
Cancer	10 TP	4 FN
No Cancer	5 FP	1000 TN

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{10}{14} = 71.42\%$$

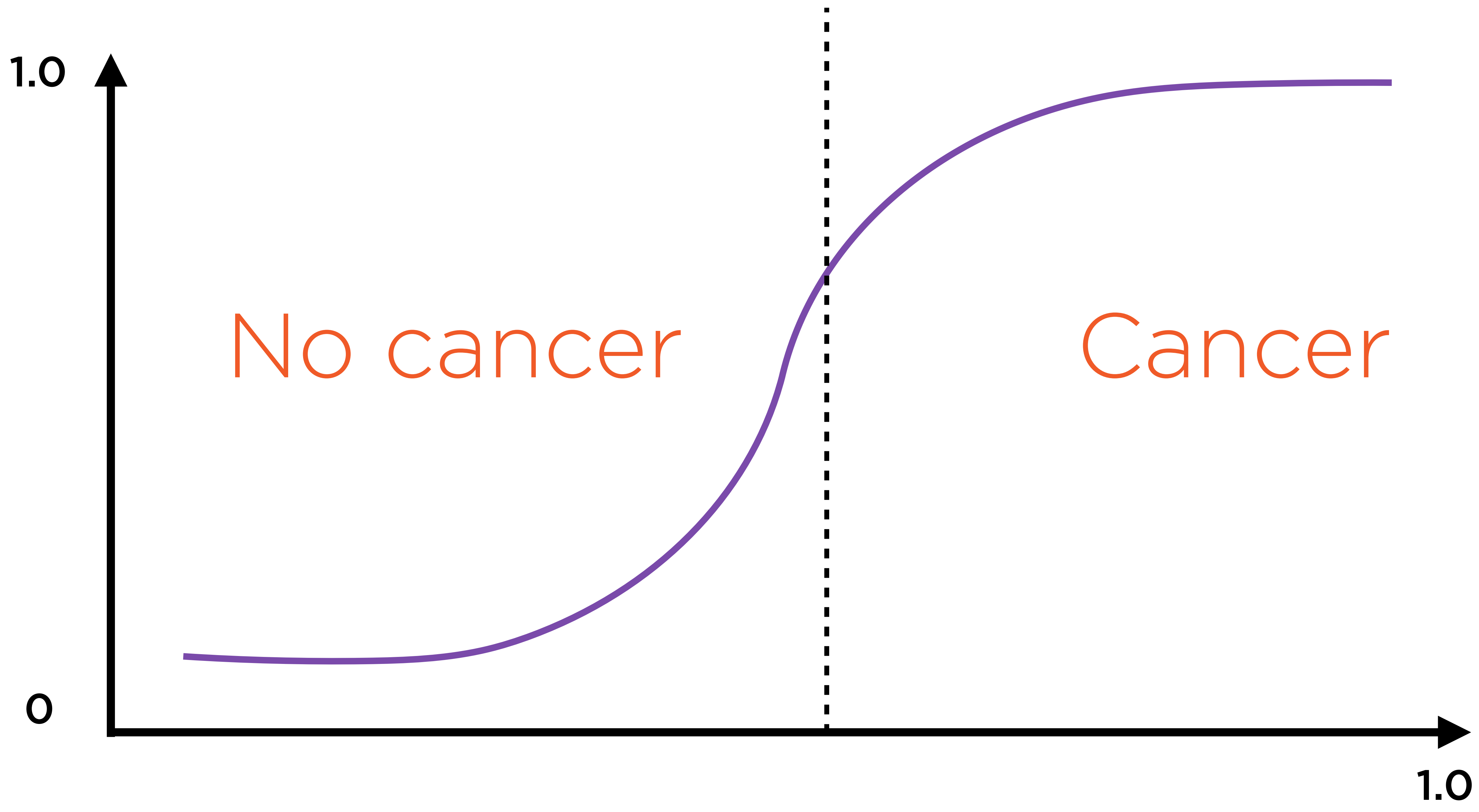
Recall

Recall = 71.42%

2 in 7 cancer cases missed

The ROC Curve

The Logistic Regression S-curve



“Always Negative”

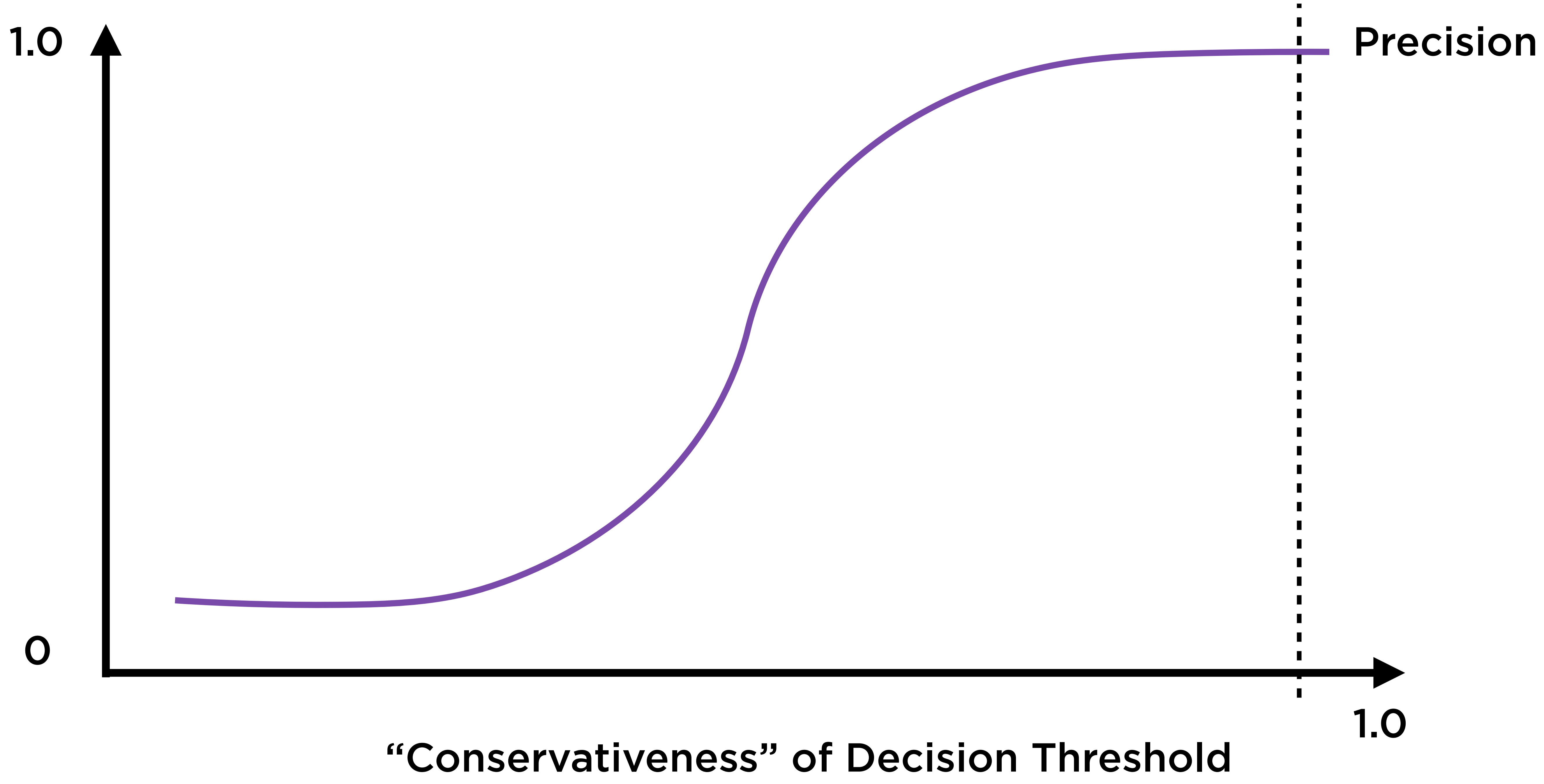
$P_{threshold} = 1$

Predicted

Actual	Cancer	No Cancer
Cancer	TP 0	FN 14
No Cancer	FP 0	TN 1005

- Recall = 0%
- Precision = Infinite
- Classifier too conservative

Precision vs. “Conservativeness”



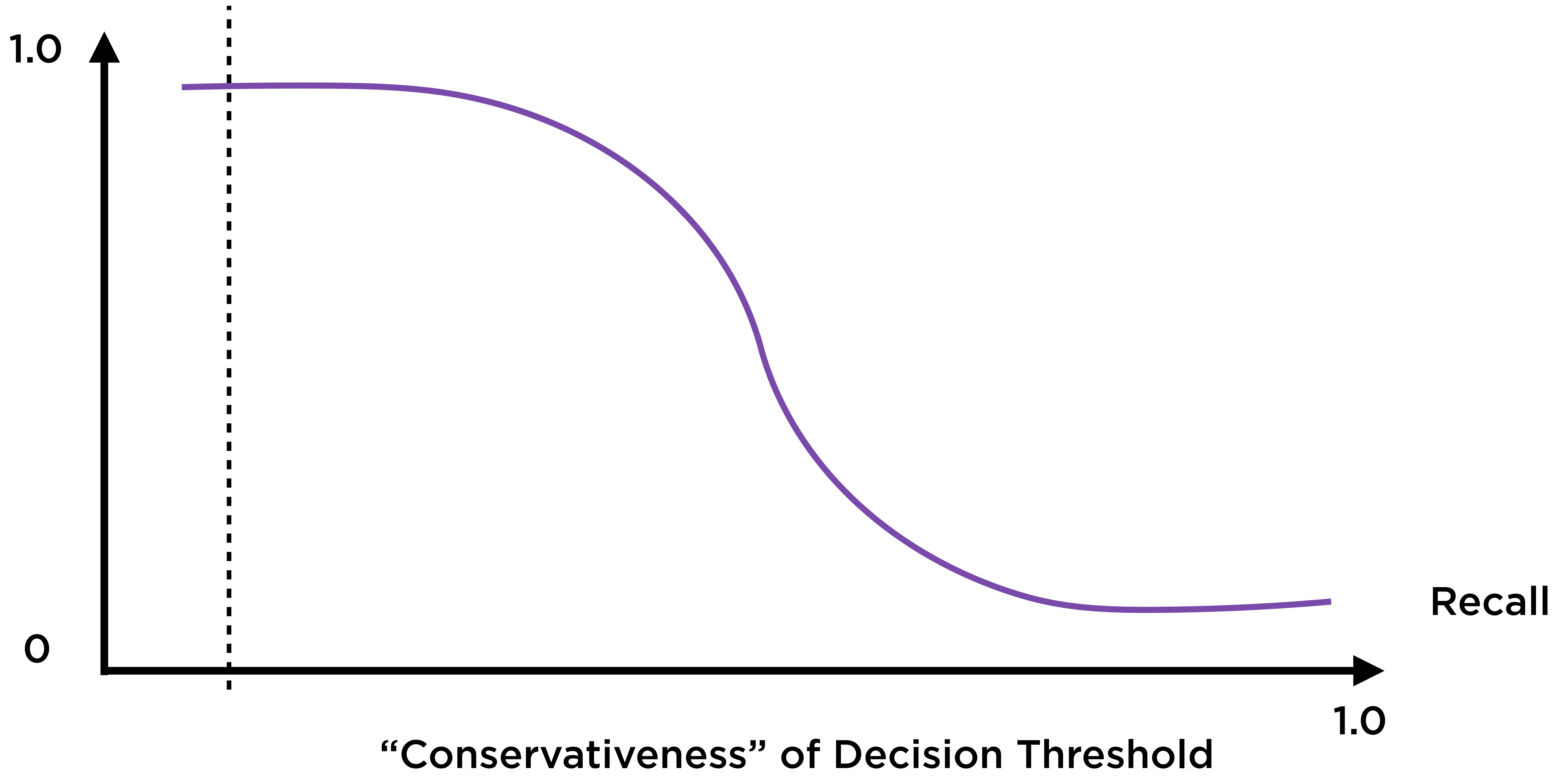
“Always Positive”

$P_{threshold} = 0$

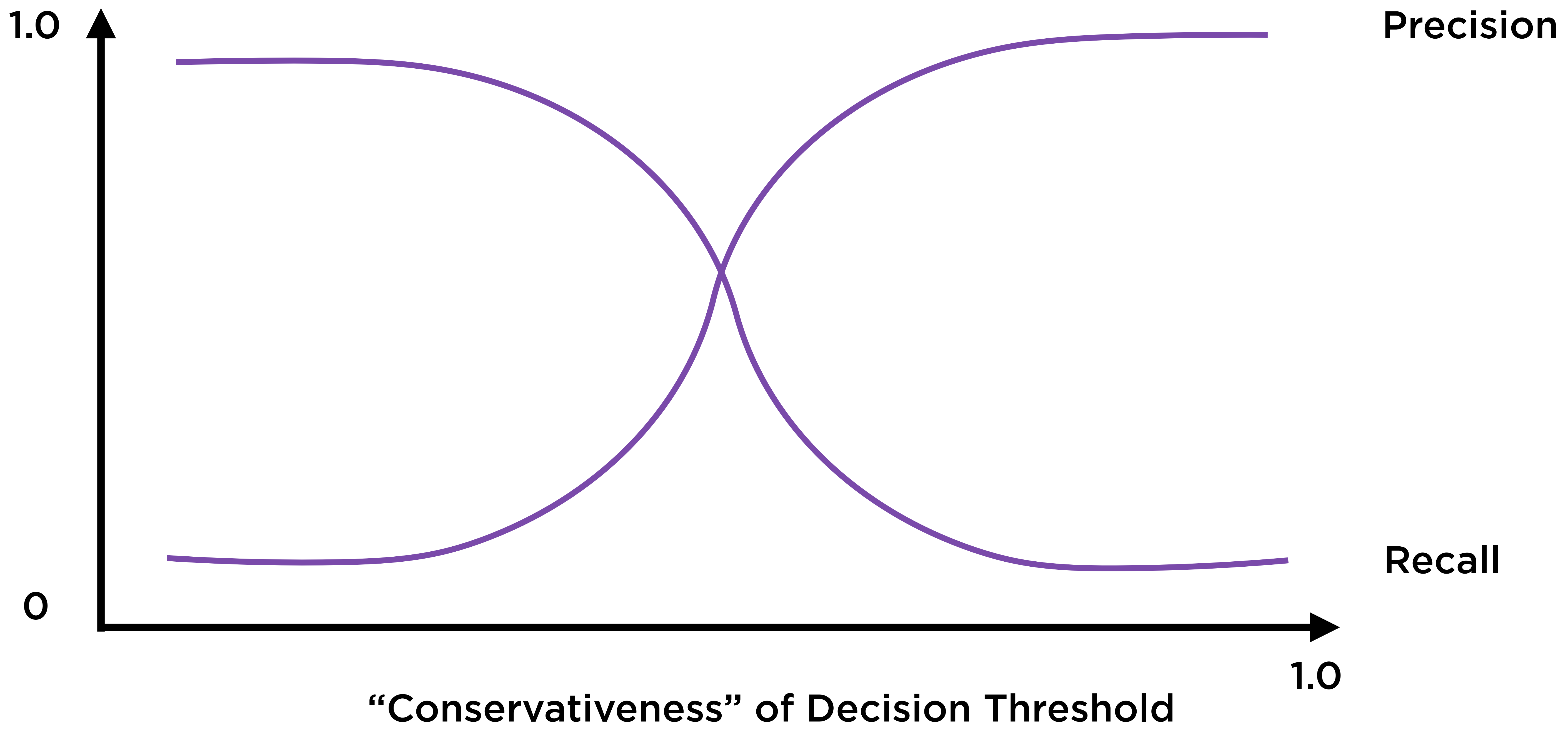
Actual \ Predicted	Cancer	No Cancer
Cancer	TP 14	FN 0
No Cancer	FP 1005	TN 0

- Recall = 100%
- Precision = 14/1019 = 13.7%
- Classifier not conservative enough

Recall vs. "Conservativeness"



Precision-Recall Tradeoff



Precision-Recall Tradeoff

Heuristics to Choose a Model

F1 Score

Harmonic mean of precision and recall

ROC Curve

Plot a curve to maximize true positives, minimize false positives

Heuristics to Choose a Model

F1 Score

Harmonic mean of precision and recall

ROC Curve

Plot a curve to maximize true positives, minimize false positives

F_1 Score

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- Harmonic mean of precision, recall
- Closer to lower of two
- Favors even tradeoff

Heuristics to Choose a Model

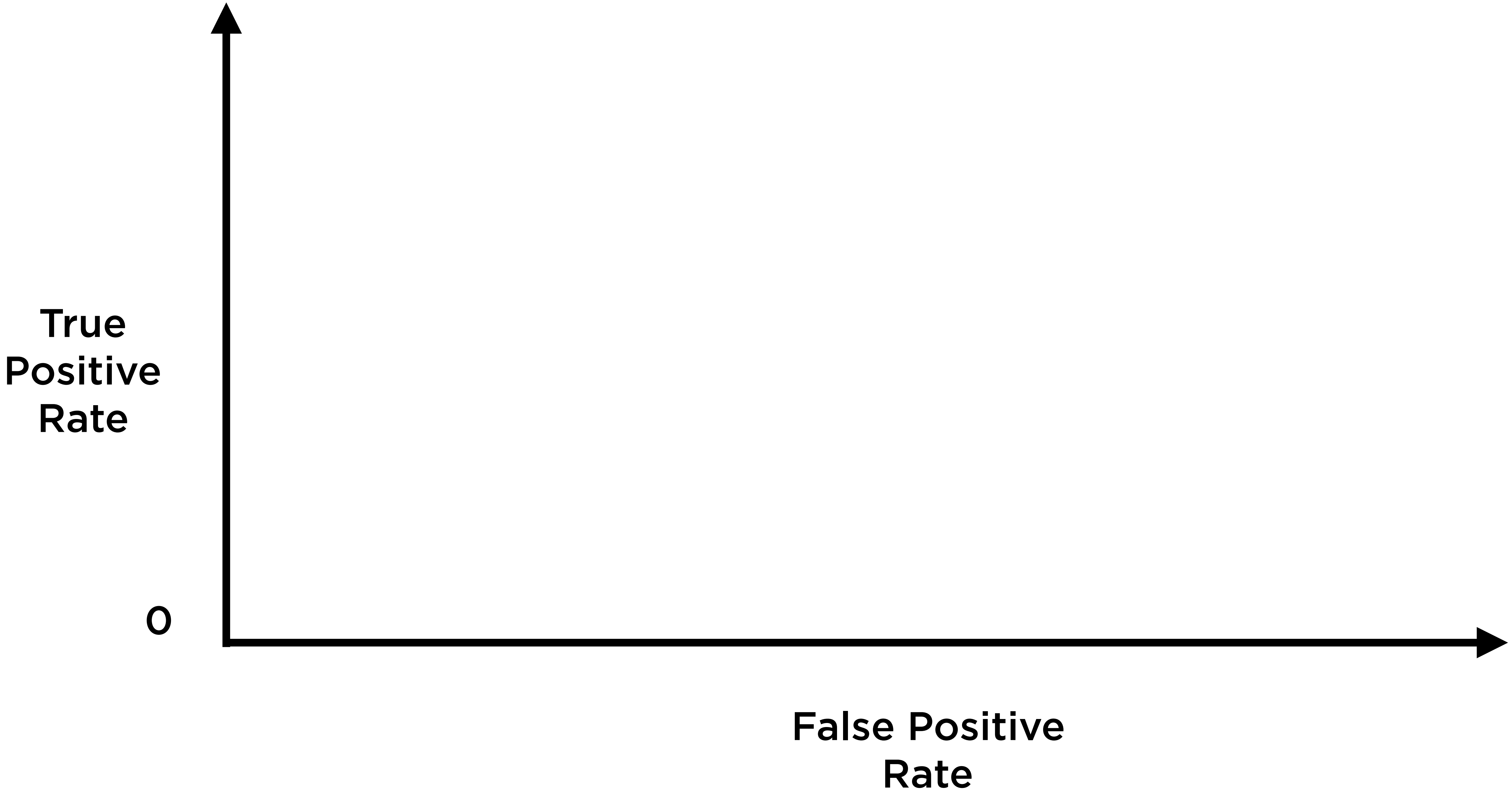
F1 Score

Harmonic mean of precision and recall

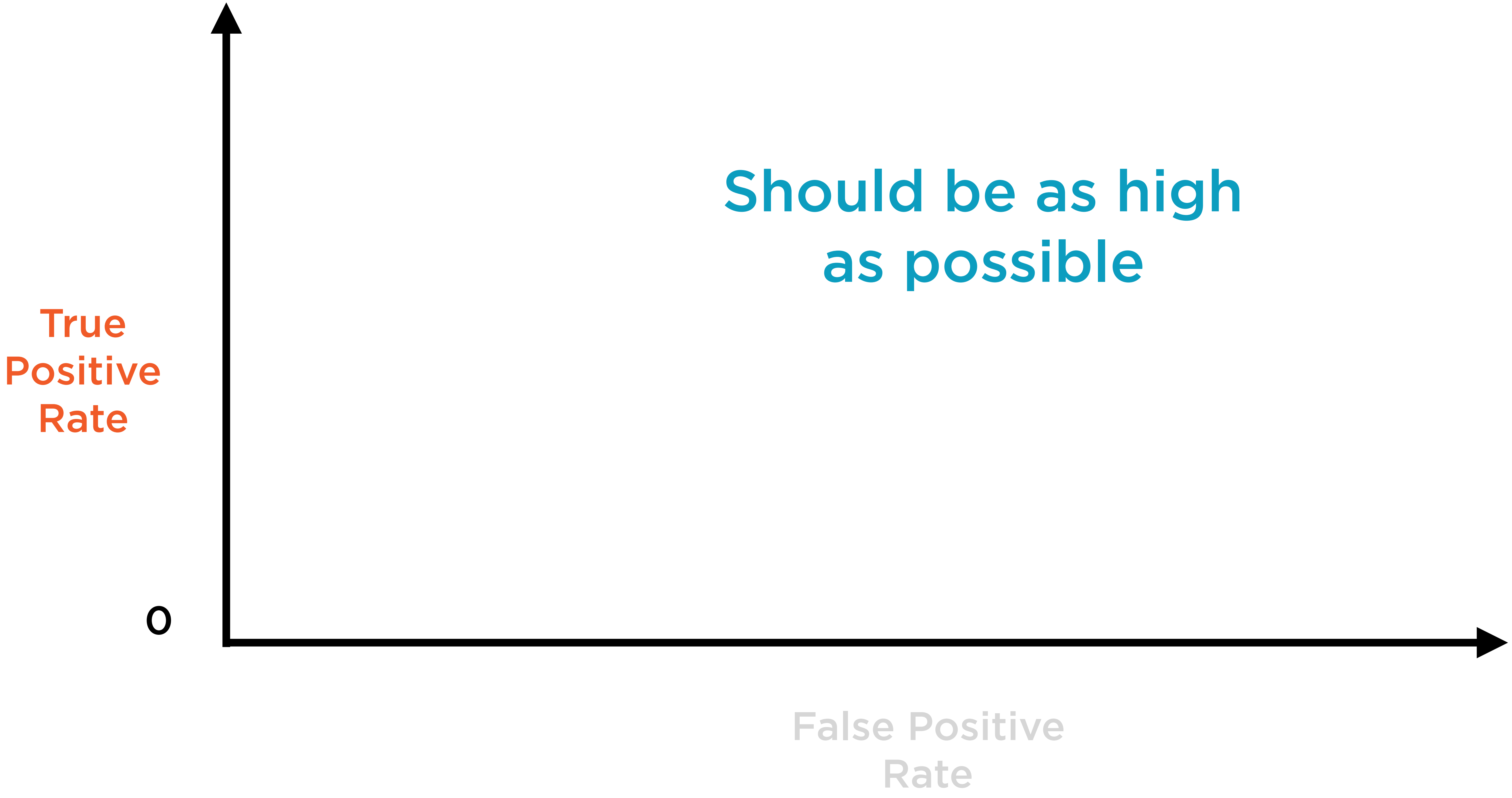
ROC Curve

Plot a curve to maximize true positives, minimize false positives

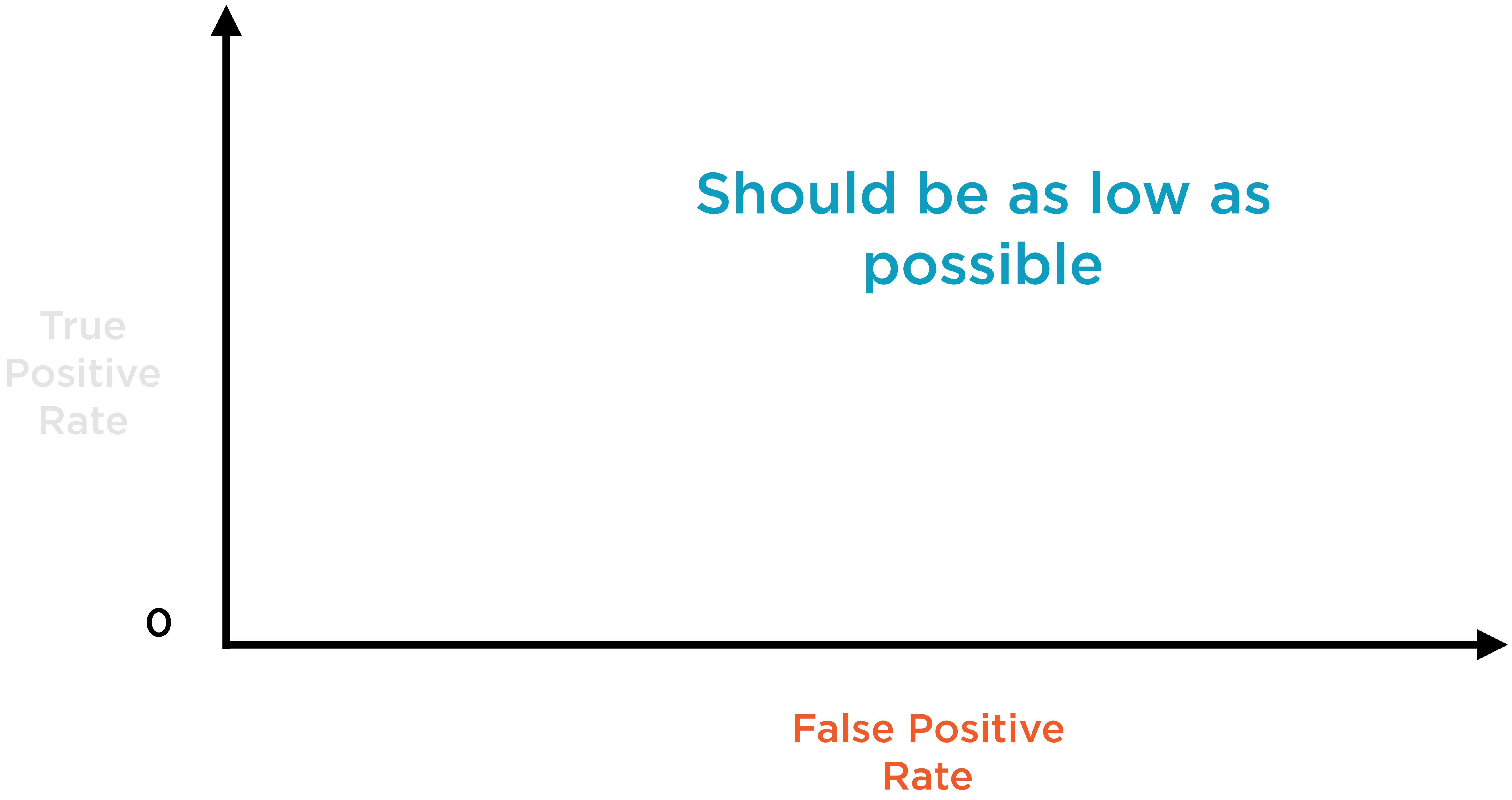
Choosing $P_{threshold}$



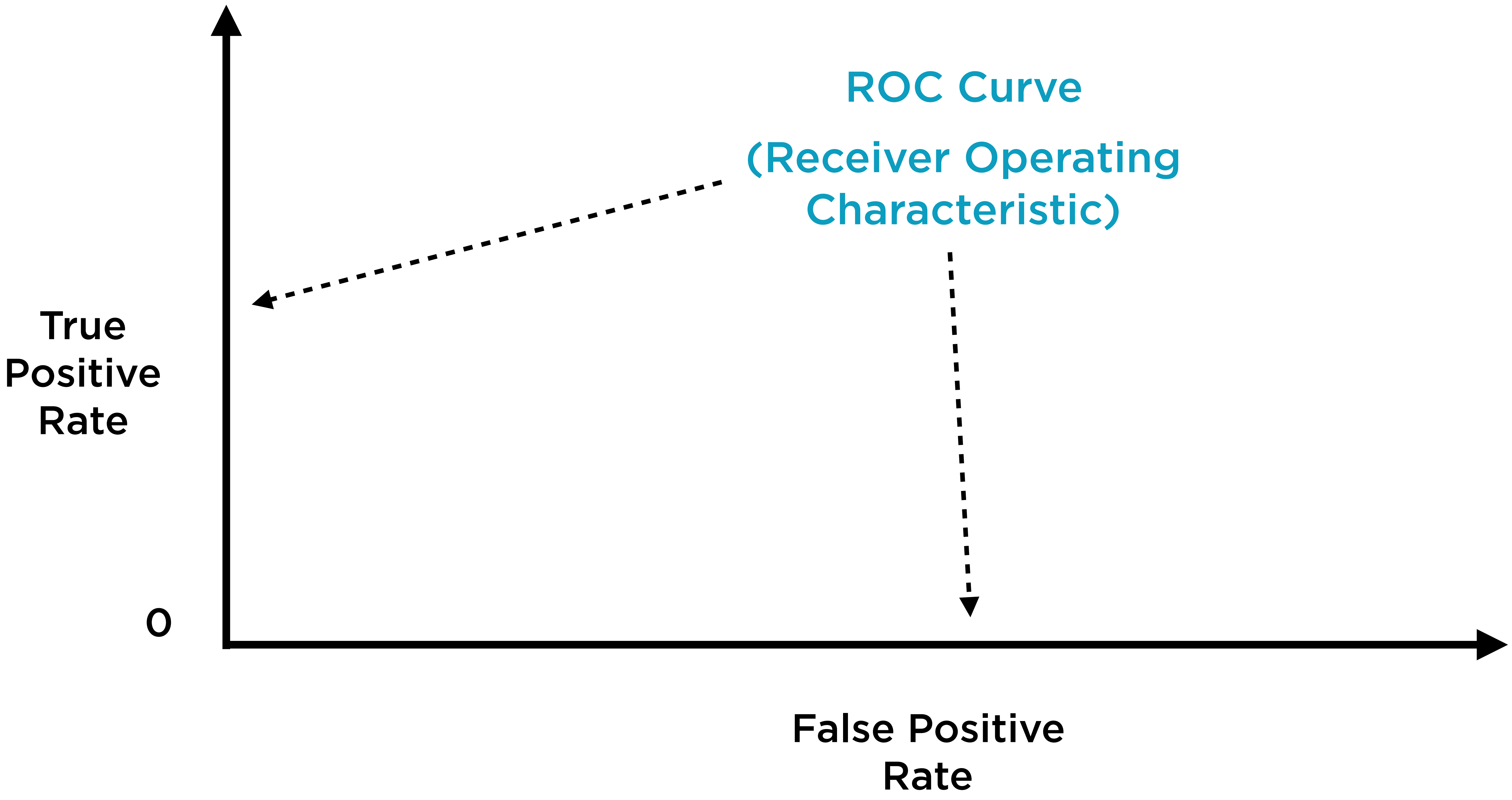
Choosing $P_{threshold}$



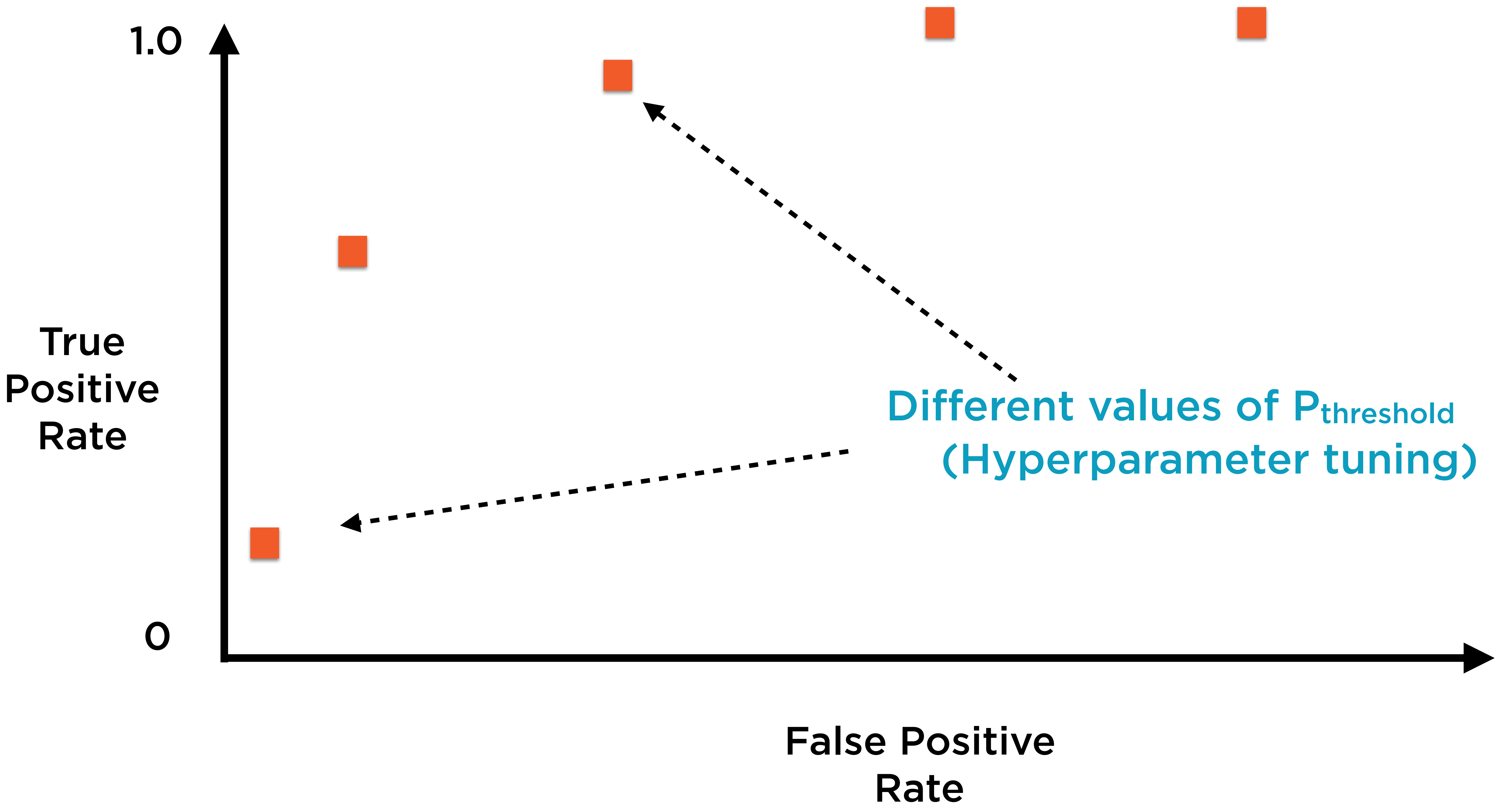
Choosing $P_{threshold}$



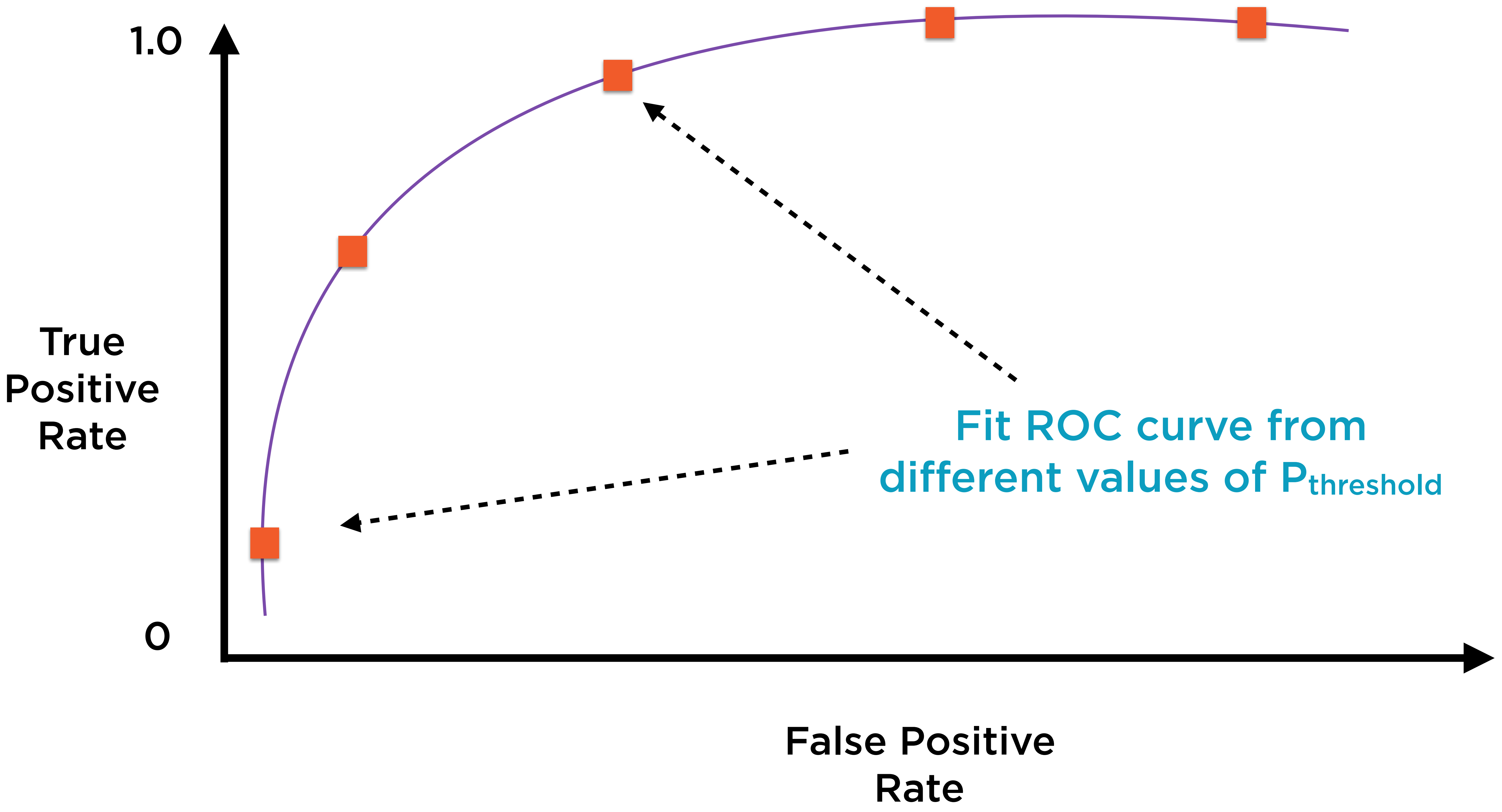
Choosing $P_{threshold}$



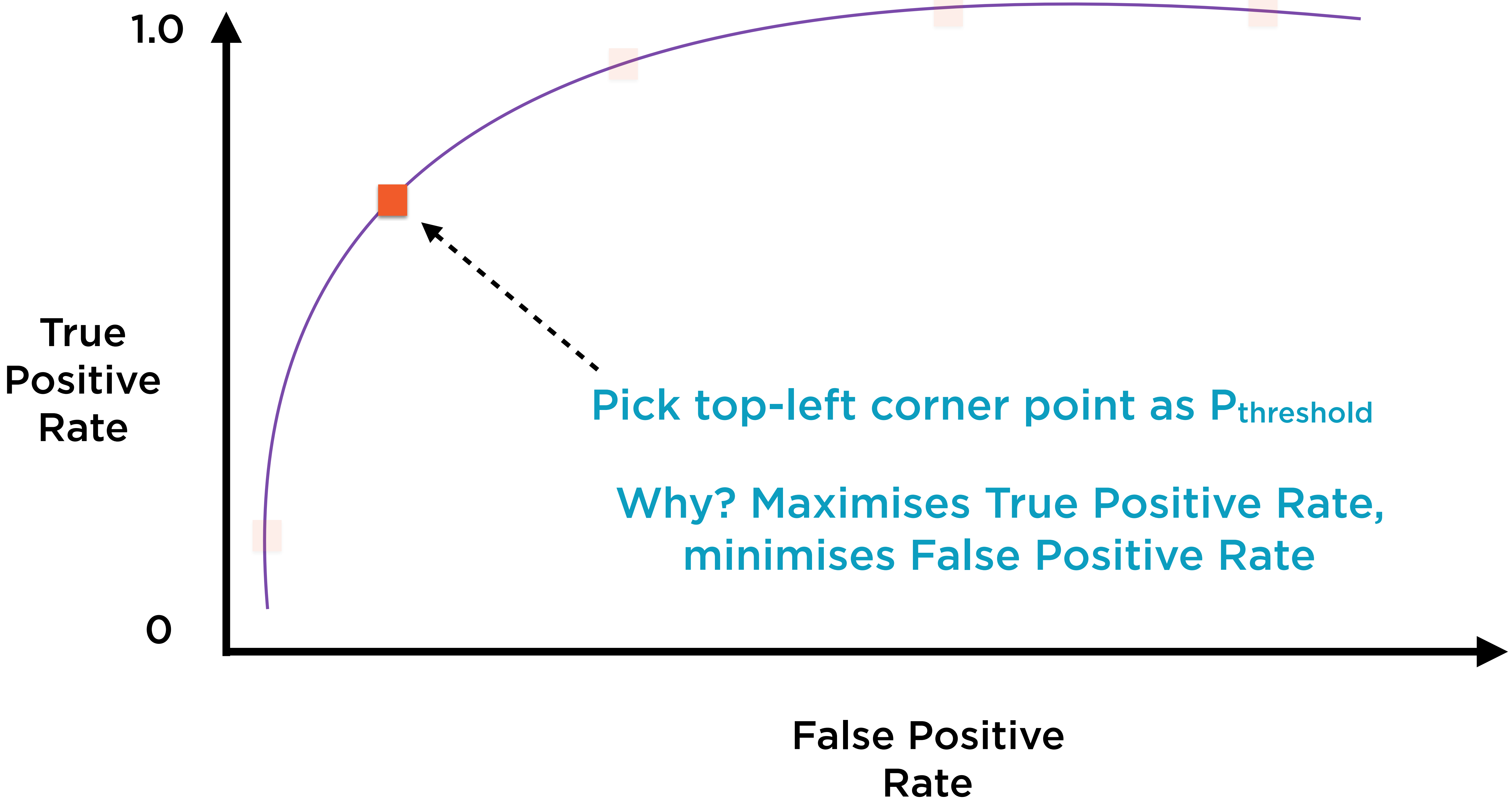
Choosing $P_{threshold}$



Choosing $P_{threshold}$



ROC Curve


1.0
0
True Positive Rate
False Positive Rate
Pick top-left corner point as P_{threshold}
Why? Maximises True Positive Rate, minimises False Positive Rate

Demo

Build and train a classification model for cancer detection

Summary

Identifying and mitigating common biases

Overfitted models

Bias/variance trade-off

Evaluating models using accuracy, precision, and recall

Understanding the ROC curve