

AI Bootcamp

Model Validation and Imbalanced Data

Module 14 Day 1



Class Objectives

By the end of class, you will be able to:

- 1 Select a target
- 2 Choose a metric
- 3 Defend a metric choice
- 4 Describe the limitations of targets and metrics
- 5 Describe overfitting
- 6 Detect overfitting
- 7 Adjust a model to optimize between over and underfitting



Why are we doing this?

01

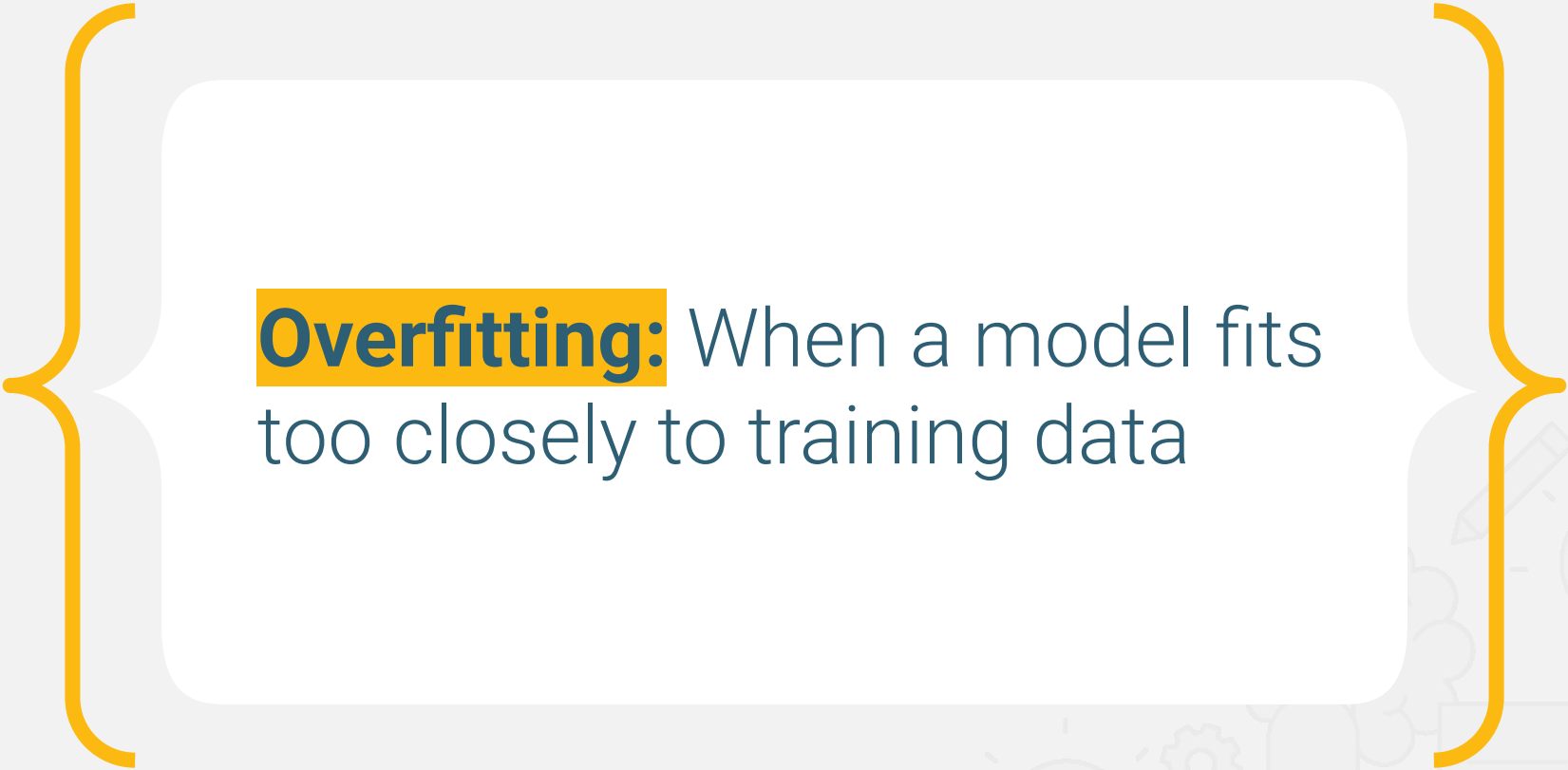
We train models on historic data, but we need them to perform predictions based on new data from the environment.

02


Models can perform well on historic data and fail to perform on new data.


03

Let's cover common pitfalls such as misplaced confidence in a model.




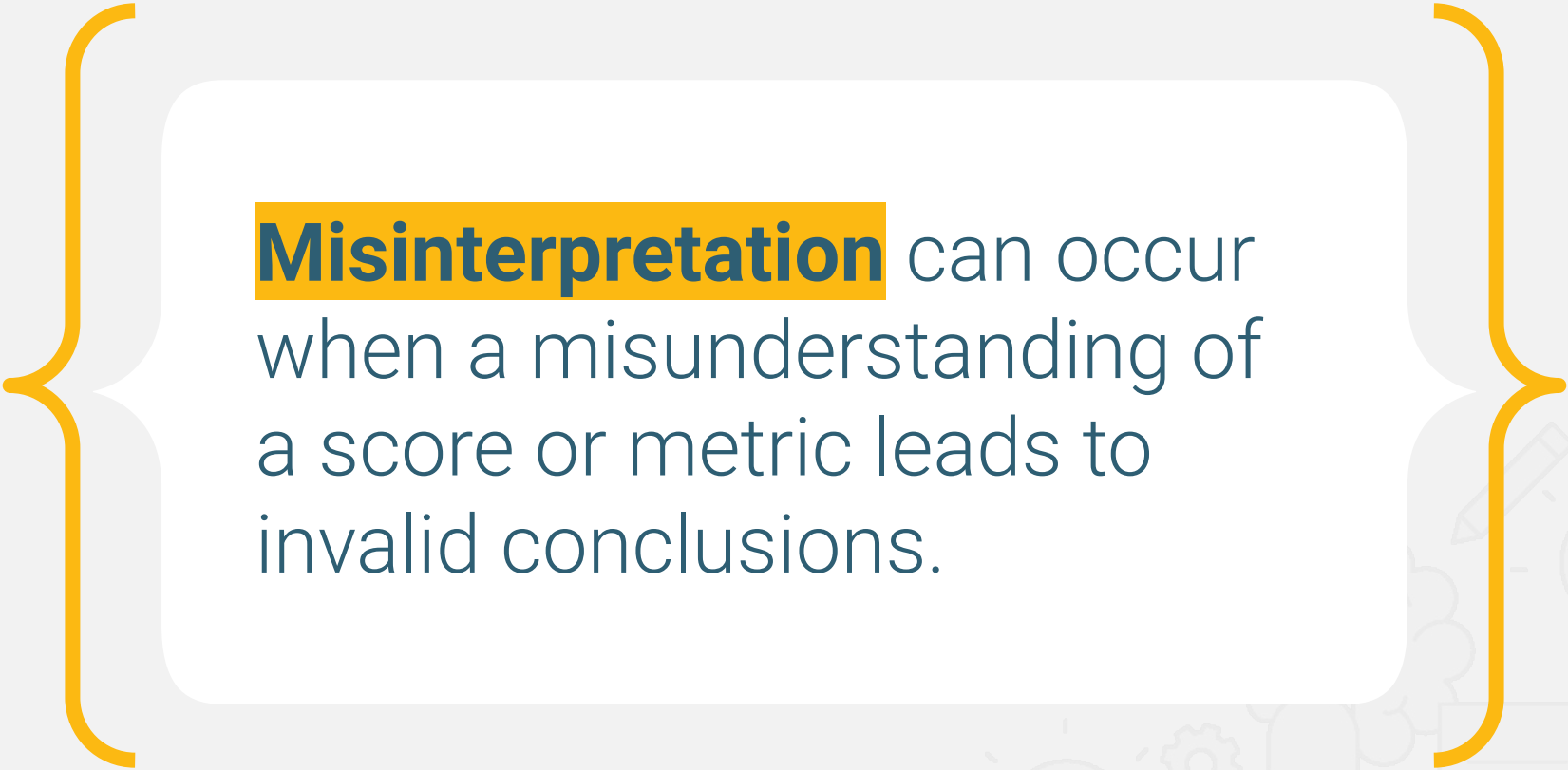
Overfitting: When a model fits too closely to training data






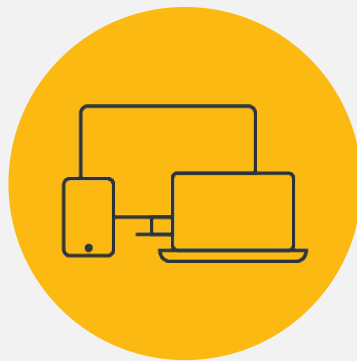
Underfitting: When a model fails to capture meaningful relationships in the data





Misinterpretation can occur
when a misunderstanding of
a score or metric leads to
invalid conclusions.





Instructor **Demonstration**

Introducing Bank Marketing Data



Bank Marketing data source:

<https://archive.ics.uci.edu/dataset/222/bank+marketing>





Bank Marketing

1

Consists of 17 columns and 4,522 rows

2

Columns represent variables that consist of integer, categorical, binary, and date formats.





Bank Marketing

01

Age column: Records the participant's age in integer format

02

Job column: Records the occupation of participant [Admin, blue-collar, entrepreneur, housemaid, mana employed, services, student, technician, unemployed, unknown]

03

Y column: Has the client subscribed a term deposit?



Activity:

First Model

In this activity, you rush through the creation of a basic Random Forest model for classification. Along the way, you'll ignore best practices and make a variety of errors. As you progress through the activity, consider how the data, model, and analysis could be improved.

Suggested Time:

15 Minutes





Time's up!
Let's review



Activity:

Target Selection

In this activity, you will be introduced to three hypothetical datasets. You are encouraged to discuss each target column choice, along with the implications of each choice.

Suggested Time:

15 Minutes



Pitfalls around poor target selection

01

Misinterpretation of results

02

Results misaligned to stakeholder requirements

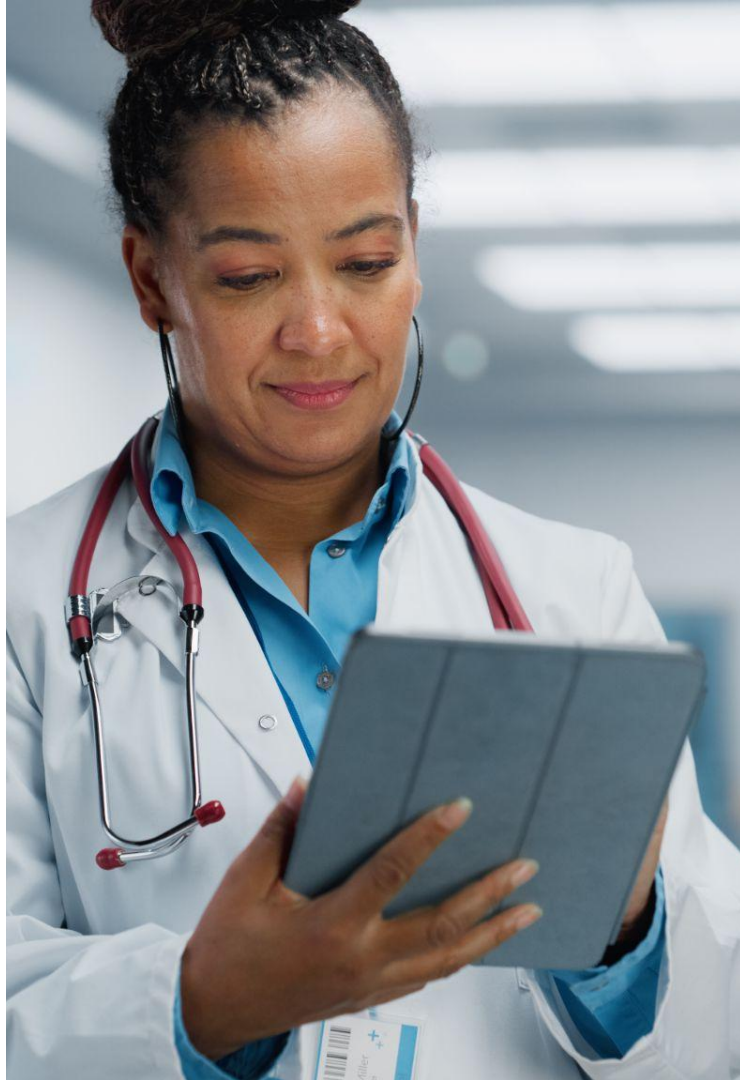


Prescription Antibiotics

01

Scenario 1

A medical dataset contains hospital intake information on thousands of patients. The hospital would like to use machine learning to help them make better decisions about which patients should be prescribed antibiotics.



Prescription Antibiotics

Target Column Choice A

A binary column that indicates whether the patient was diagnosed with an infection.

Target Column Choice B

A binary column that indicates whether the patient responded well to antibiotics while in the hospital.

Target Column Choice C

A column that indicates the patient's self-reported health 30 days after antibiotic treatment on a 1 to 10 scale.



02

A stock market dataset contains data on every trade a particular company has made in the previous 5 years. The company would like to use machine learning to predict whether a trade will be profitable.



Stock Market Profits

Target Column Choice A

A column that gives the total profit (or loss) from each trade.

Target Column Choice B

A column that gives the percentage profit (or loss) from each trade.

Target Column Choice C

A binary column that indicates whether a trade made at least 10% profit.



Earthquake Prediction

03

Scenario 3

A dataset has earthquake records from the past 50 years. The United States Geological Survey would like to use machine learning to better predict aftershock impacts after an earthquake.



Earthquake Prediction

Target Column Choice A

A column indicating whether there was an aftershock after each earthquake.

Target Column Choice B

A column with the total economic impact in dollars from the aftershocks after each earthquake.

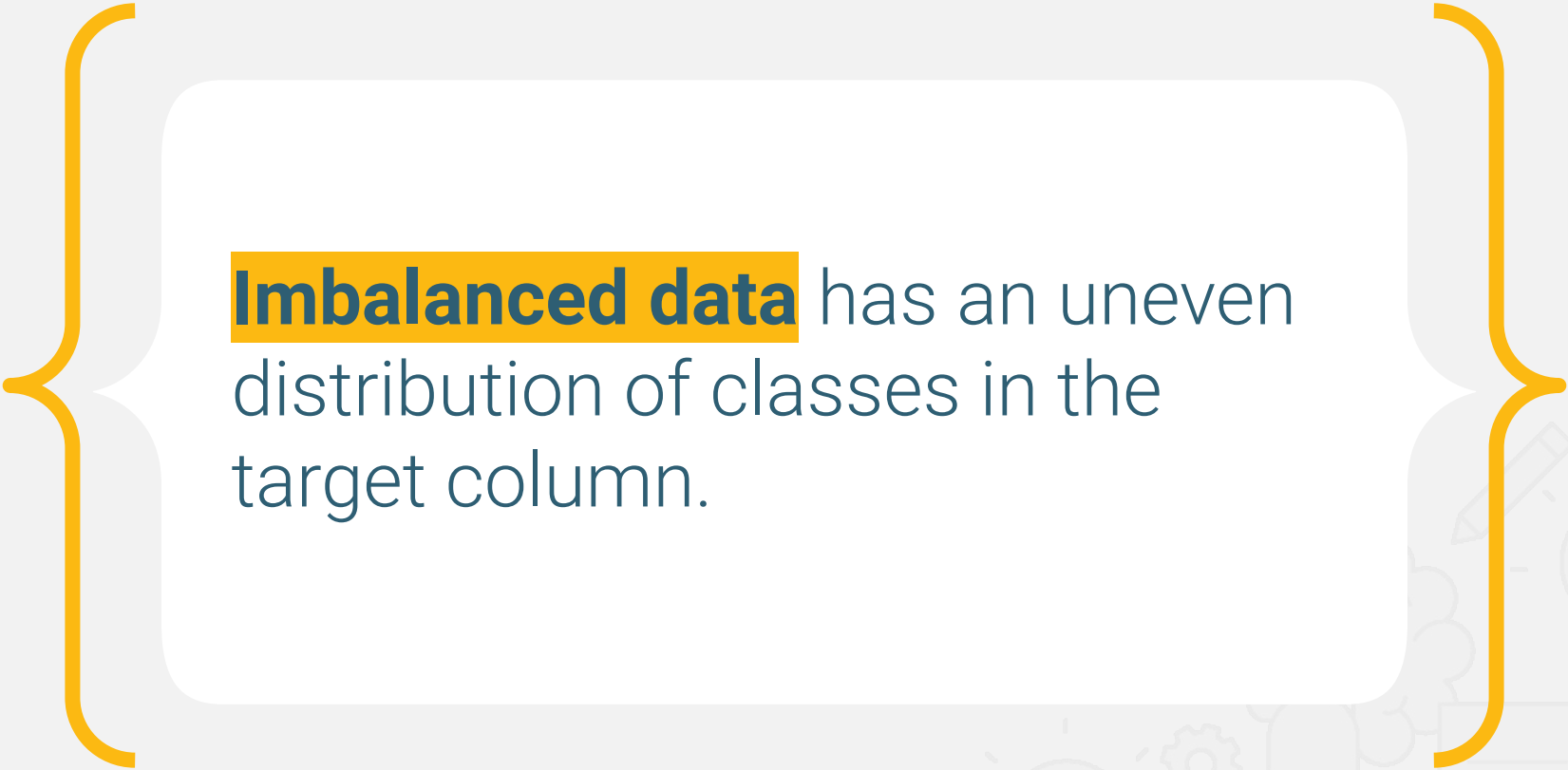
Target Column Choice C

A column indicating the number of lives lost in aftershocks after each earthquake.




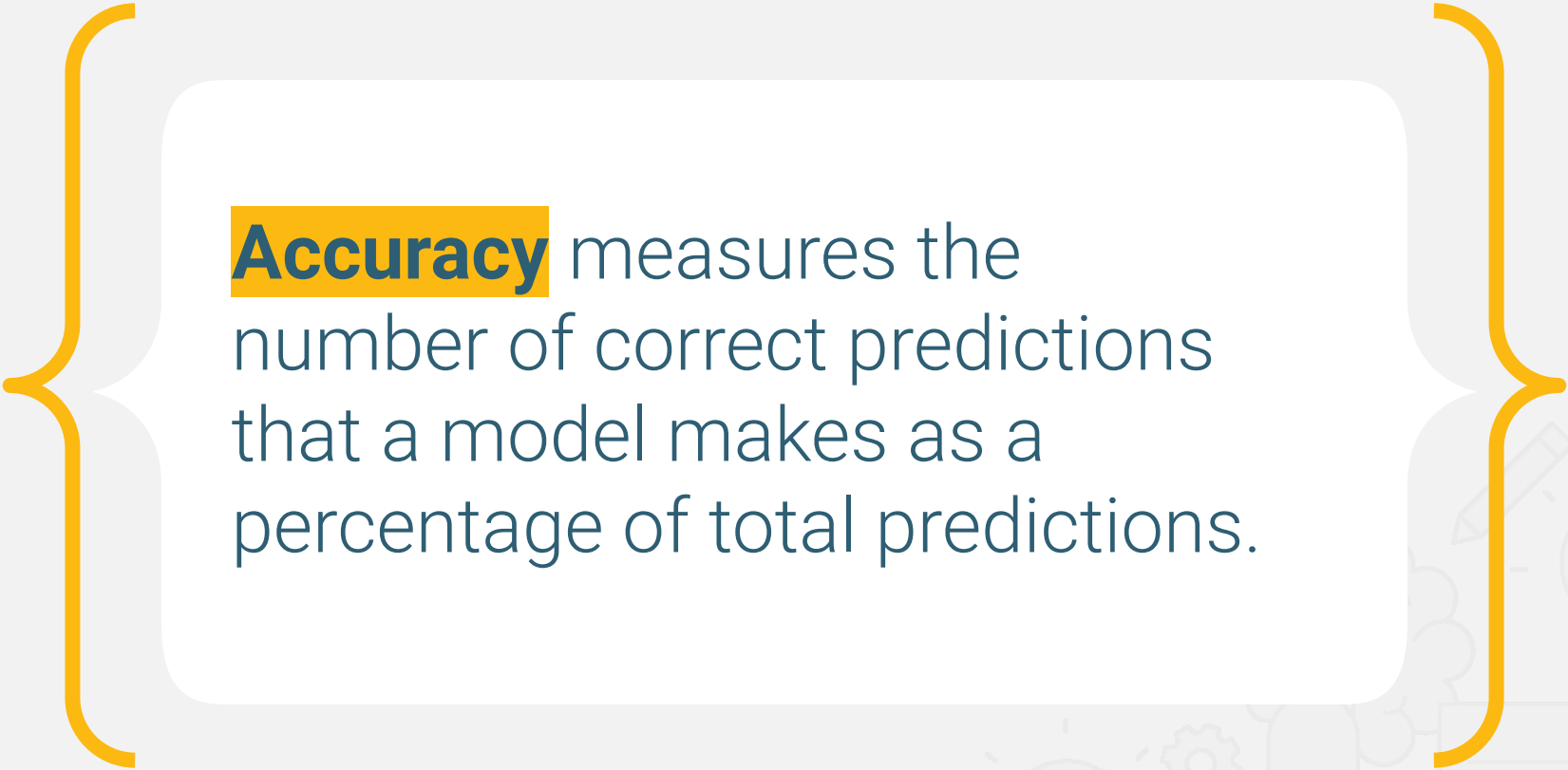
Metrics






Imbalanced data has an uneven distribution of classes in the target column.





Accuracy measures the number of correct predictions that a model makes as a percentage of total predictions.



Accuracy

Pros

- Simple output and easy to calculate

Cons

- With imbalanced data, a model can achieve high accuracy scores by only predicting the majority class.
- Accuracy doesn't explain which errors are being made (false positives or false negatives), and the costs of a false positive and a false negative are rarely equal in real-world situations.
- Accuracy does not take into account the certainty of the model in its calculation.

Confusion Matrix

	Predicted to be false	Predicted to be true
Actually false	True negative (TN)	False positive (FP)
Actually true	False negative (FN)	True positive (TP)

Generating one of these matrices for a model is simple as scikit-learn already includes a `confusion_matrix` function in the `metrics` package.

```
from sklearn.metrics import confusion_matrix  
  
confusion_matrix(y_test, predictions)
```

Accuracy

	Predicted to be false	Predicted to be true
Actually false	True negative (TN)	False positive (FP)
Actually true	False negative (FN)	True positive (TP)

Accuracy can be calculated from the confusion matrix.

$$Accuracy = \frac{(TN + TP)}{(TN + FP + FN + TP)}$$

Sensitivity (or Recall)

	Predicted to be false	Predicted to be true
Actually false	True negative (TN)	False positive (FP)
Actually true	False negative (FN)	True positive (TP)

Using sensitivity as the main unit of measurement allows you to find out how many of the **actually true** data points were identified correctly.

$$\text{Sensitivity} = \frac{TP}{(FN + TP)}$$

Specificity

	Predicted to be false	Predicted to be true
Actually false	True negative (TN)	False positive (FP)
Actually true	False negative (FN)	True positive (TP)

Using specificity as the main unit of measurement allows you to find out how many of the **actually false** data points were identified correctly.

$$\textit{Specificity} = \frac{TN}{(TN + FP)}$$

Precision

	Predicted to be false	Predicted to be true
Actually false	True negative (TN)	False positive (FP)
Actually true	False negative (FN)	True positive (TP)

Precision identifies how many of the predicted true results were actually true.

$$\textit{Precision} = \frac{TP}{(TP + FP)}$$



Classification Report

A classification report provides an overview of multiple metrics.



Precision



Recall (Sensitivity)



F1 score



Accuracy

```
from sklearn.metrics import classification_report  
  
classification_report(y_test, predictions)
```

	precision	recall	f1-score	support
1	0.74	0.16	0.26	2000
0	0.90	0.99	0.94	14632
accuracy			0.89	16632
macro avg	0.82	0.58	0.60	16632
weighted avg	0.88	0.89	0.86	16632

F1 score balances sensitivity and precision.

$$F1\ score = \frac{2(Precision \times Sensitivity)}{(Precision + Sensitivity)}$$



Balanced accuracy measures the accuracy of each class, then averages the results.

Balanced Accuracy

Pros

- Weights the accuracy of all classes evenly, regardless of the number of instances in each class
- Good metric for imbalanced datasets
- Simple to calculate

Cons

- Does not take into account the certainty of a model
- Does not explain whether errors are from false positives or false negatives

Confusion Matrix

	Prediction: YES	Prediction: NO
Truth: YES	70	0
Truth: NO	30	0

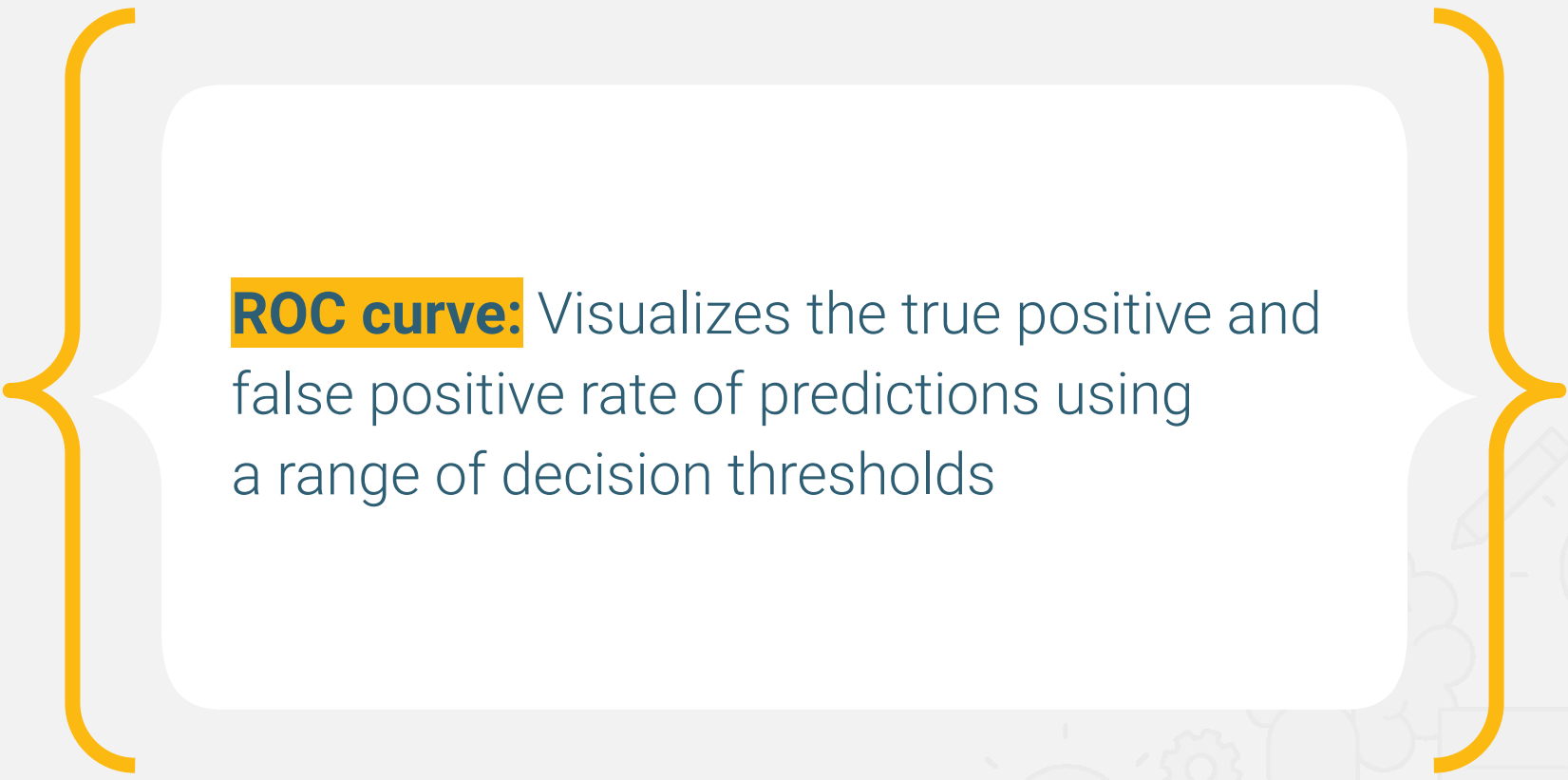
- All predictions are “yes”.
- The accuracy is 70%.
- The balanced accuracy averages the accuracy of positive and negative classes.

True yeses: 70 out of 70 correct = 100%


True nos: 0 out of 30 correct = 0%

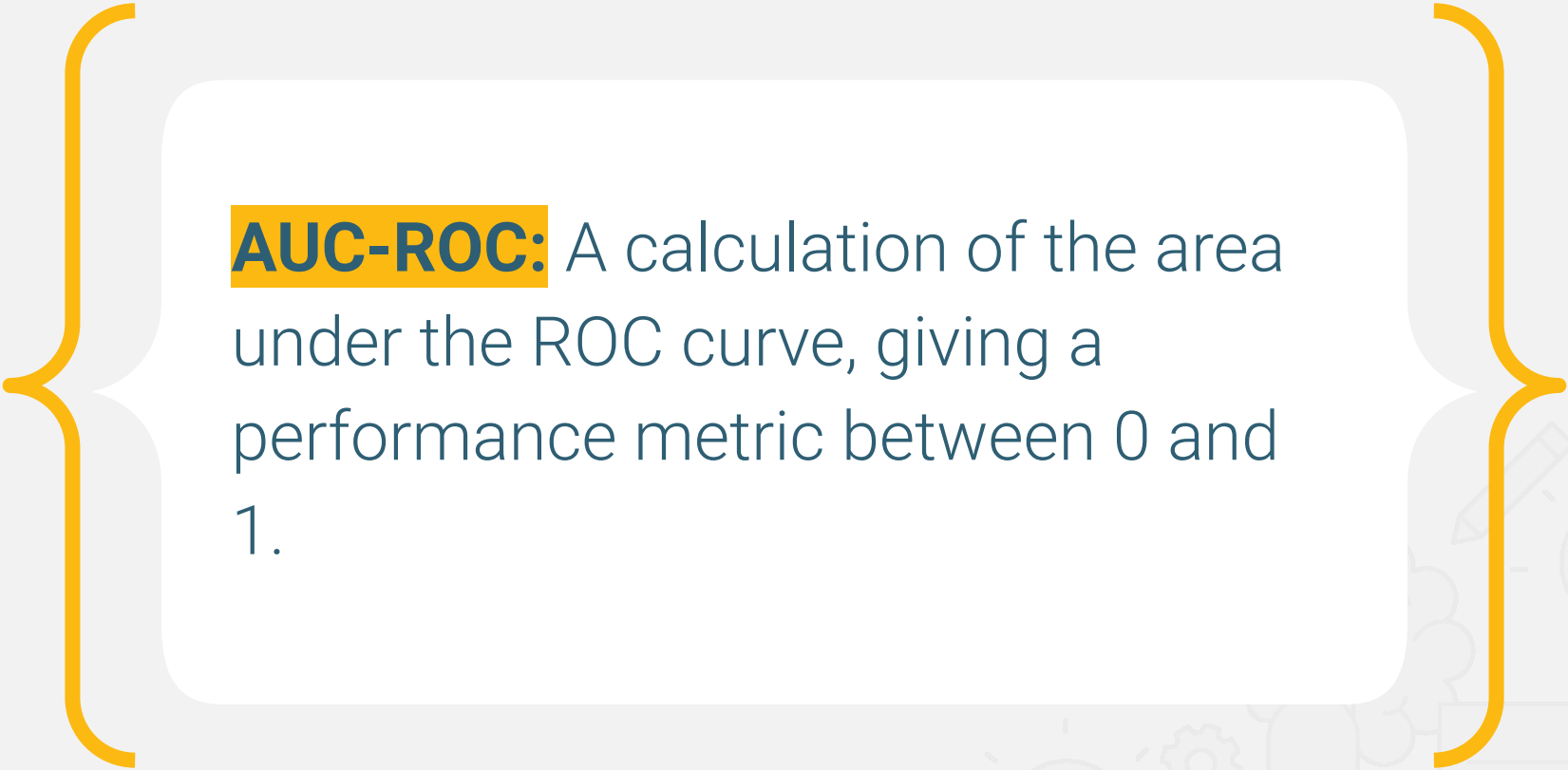
Accuracy: $70/100 = 70\%$

Balanced accuracy: $(100\% + 0\%) / 2 = 50\%$

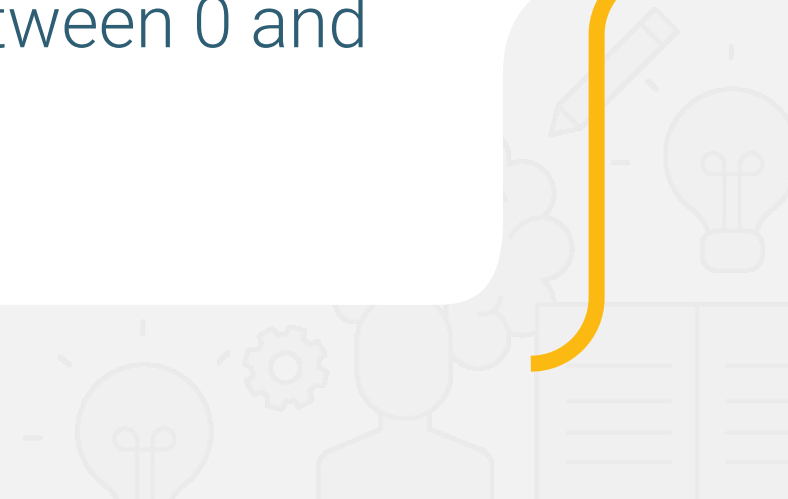


ROC curve: Visualizes the true positive and false positive rate of predictions using a range of decision thresholds





AUC-ROC: A calculation of the area under the ROC curve, giving a performance metric between 0 and 1.





Rewind: Model Prediction

1

A model outputs a decimal value, normally between 0 and 1.

2

That value is **rounded** to the nearest whole number, which becomes the prediction of the model.

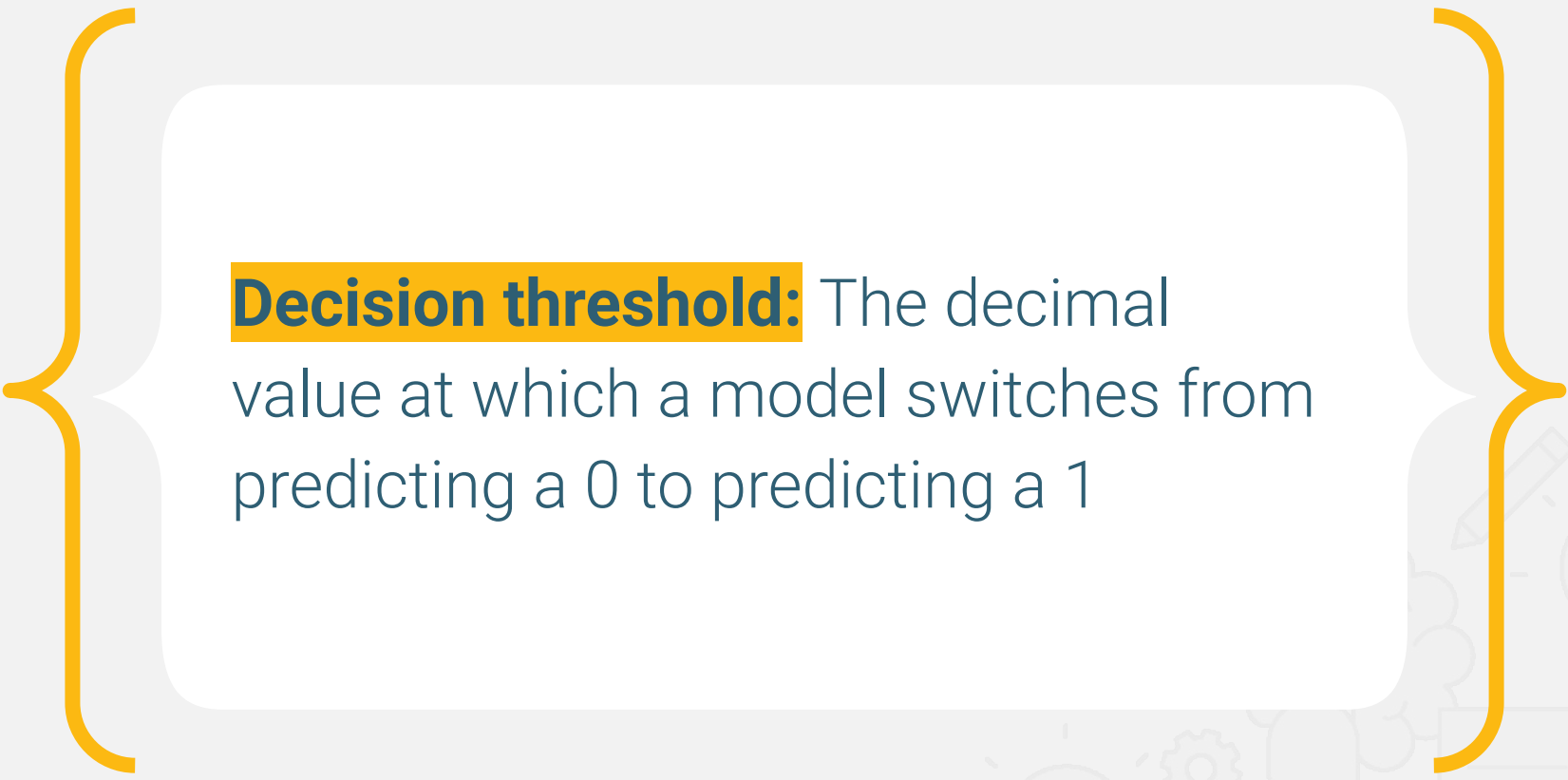
3

In some models, it is possible to output the decimal values *before* rounding.


- With sklearn, this is done using the `predict_proba` method.

4

With the decimal values, you can manually change the point at which a prediction becomes a 1 or a 0.

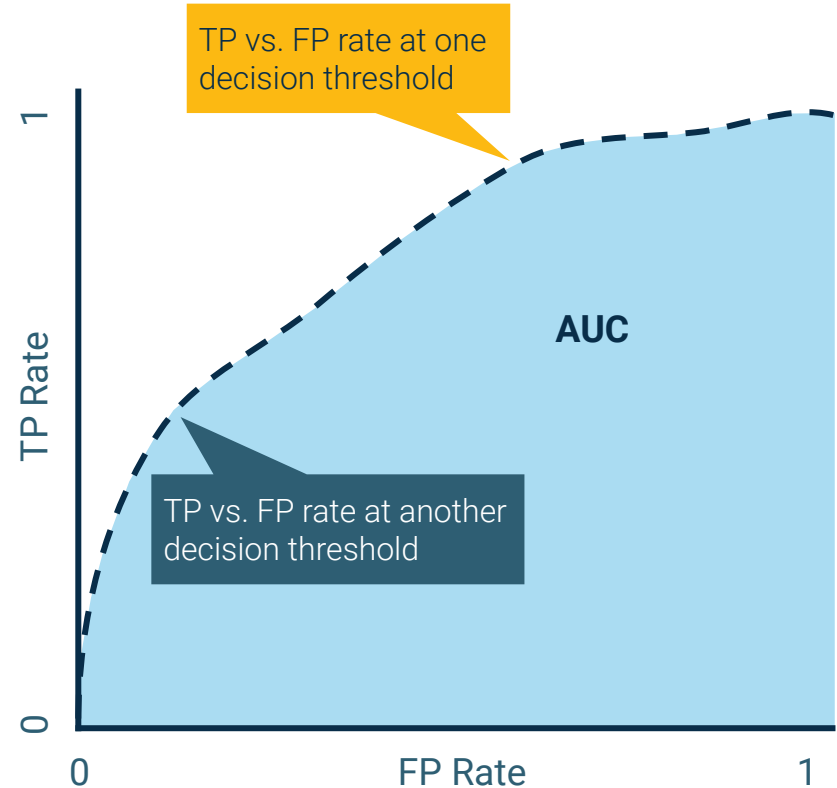


Decision threshold: The decimal value at which a model switches from predicting a 0 to predicting a 1



AUC-ROC Curve

- The y-axis represents the True Positive Rate (TPR).
- The x-axis represents the False Positive Rate (FPR).
- The dotted line shows the relationship between true positives and false positives as the **decision threshold** is increased.
 - This effectively helps measure not just how often the model is correct, but also how certain it is of its correctness.
- A perfect ROC would be vertical at (0,0) and horizontal at (0,1), resulting in a perfect AUC of 1.
- A diagonal line from (0,0) to (1,1) would indicate completely random predictions.
- Like the “accuracy” metric, this means that random guessing would result in an AUC of about 0.5 with balanced datasets.



AUC-ROC Curve

Pros

- Gauges the certainty of the model in its predictions
- AUC measures the degree of separability, or how effective the model is at distinguishing classes.
- Penalizes the model for having predictions with low degrees of certainty

Cons

- Can still be skewed by imbalanced data
- Doesn't differentiate between error types

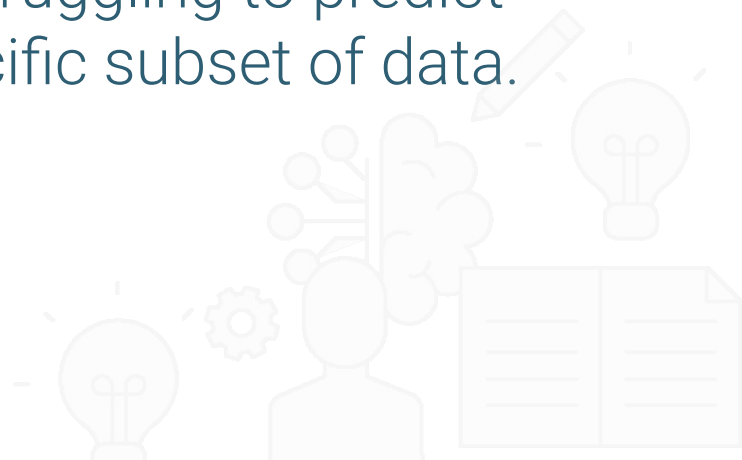
Comparing Metrics

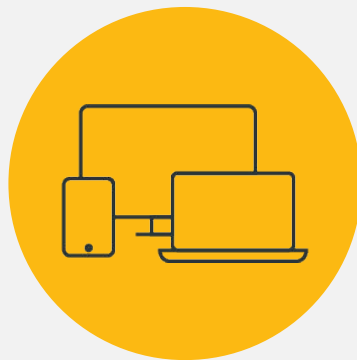
Accuracy	Balanced accuracy	AUC-ROC score
Only for balanced datasets where false positives and false negatives are equally costly	Appropriate for imbalanced datasets, but otherwise similar to accuracy	Accounts for the certainty of the model, gives more information than accuracy, but is difficult to calculate and will still struggle with imbalanced data



Sometimes, we require more info...

- If false positive is more costly than a false negative or vice versa.
- If a model is struggling to predict data for a specific subset of data.





Instructor **Demonstration**

Metrics



Activity:

Metrics

In this activity, you will have an opportunity to apply several metrics to a logistic regression model trained to predict whether crowdfunding projects will reach their target.

Suggested Time:

15 Minutes





Time's up!
Let's review

Overfitting

A sufficiently complex model will always shape itself to the training data.



Some relationships in the training data will help the model make predictions on new data.



Other relationships are meaningless and simply add noise to the model.

Underfitting

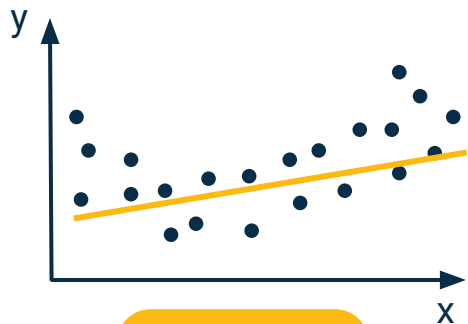
Underfitting occurs when a model fails to capture relationships between inputs and target values.



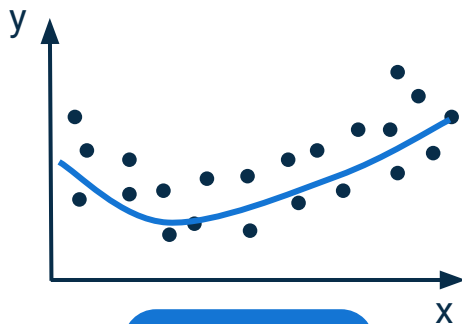
An underfit model will not learn the meaningful relationships in the data and will fail to make meaningful predictions.

Overfitting and Underfitting

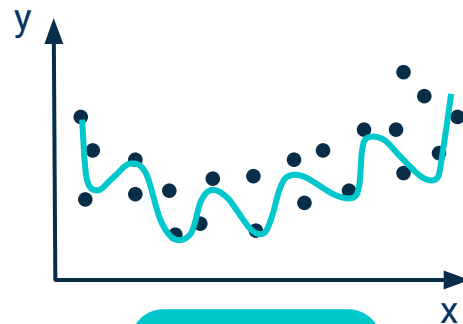
A well-fitted model lies somewhere between an overfitted and underfitted model and has a good balance between bias and variance.



Underfitting



Good Fit



Overfitting



Overfitting & underfitting:

These should be kept in balance.





Instructor **Demonstration**

Overfitting



Activity:

Overfitting

In this activity, you will split the data into training and testing sets to identify overfitting. Then, you will use hyperparameter variation on **max_depth** to determine the optimal value to balance underfitting and overfitting. While completing this activity, consider what other hyperparameters you could modify to improve the fit of the model.

Suggested Time:

15 Minutes





Time's up!
Let's review



Activity:

Bank Targets and Metrics

In this activity, you will discuss the implications of the chosen target column for the Bank Marketing dataset. Then, you will discuss several potential metrics that could be used to evaluate the model, along with the implications of choosing each metric.

Suggested Time:

20 Minutes





Time's up!
Let's review



Activity:

Second Model

In this activity, you will try again to fit a logistic regression model using the Bank Marketing data, but this time you will use balanced accuracy as a metric, test the model for overfitting, and attempt to tune the C parameter in the logistic regression model to prevent over and underfitting.

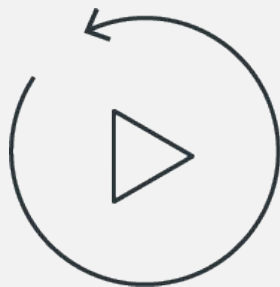
Suggested Time:

25 Minutes





Time's up!
Let's review



Let's recap



Review the Class Objective

In this lesson, you learned how to:

- 1 Select a target
- 2 Choose a metric
- 3 Defend a metric choice
- 4 Describe the limitations of targets and metrics
- 5 Describe overfitting
- 6 Detect overfitting
- 7 Adjust a model to optimize between over and underfitting



Next

In the next lesson, you will explore advanced preprocessing techniques to enhance data quality and usability for machine learning models.



Questions?





The End