Machine Learning Basics: Understanding Overfitting and Underfitting

This slide provides a high-level overview of the concepts of overfitting and underfitting in machine learning models, using a house price prediction example.

Predicting House Prices

- **PROBLEM DEFINITION**

  Predicting the price of a house based on various features like size, number of bedrooms, and age of the house.

- **FEATURES (INPUTS)**

  The model uses the size of the house in square feet, the number of bedrooms, and the age of the house (in years) as input features.

- **OUTPUT LABEL (TARGET)**

  The target or output label is the price of the house in dollars.

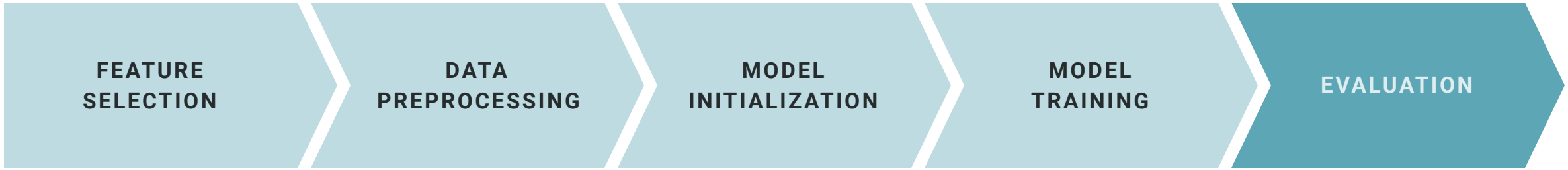
- **TRAINING DATA EXAMPLE**

  The model is trained on a dataset that includes the size, number of bedrooms, age, and price of several houses.

- **MODEL TRAINING**

  A linear regression model is used to learn the relationship between the input features and the target house price.

Model Training



Identify the relevant features that influence house prices, such as size (in square feet), number of bedrooms, and age of the house (in years).

Prepare the training data by ensuring the features and target variable (house price) are in the correct format and scale.

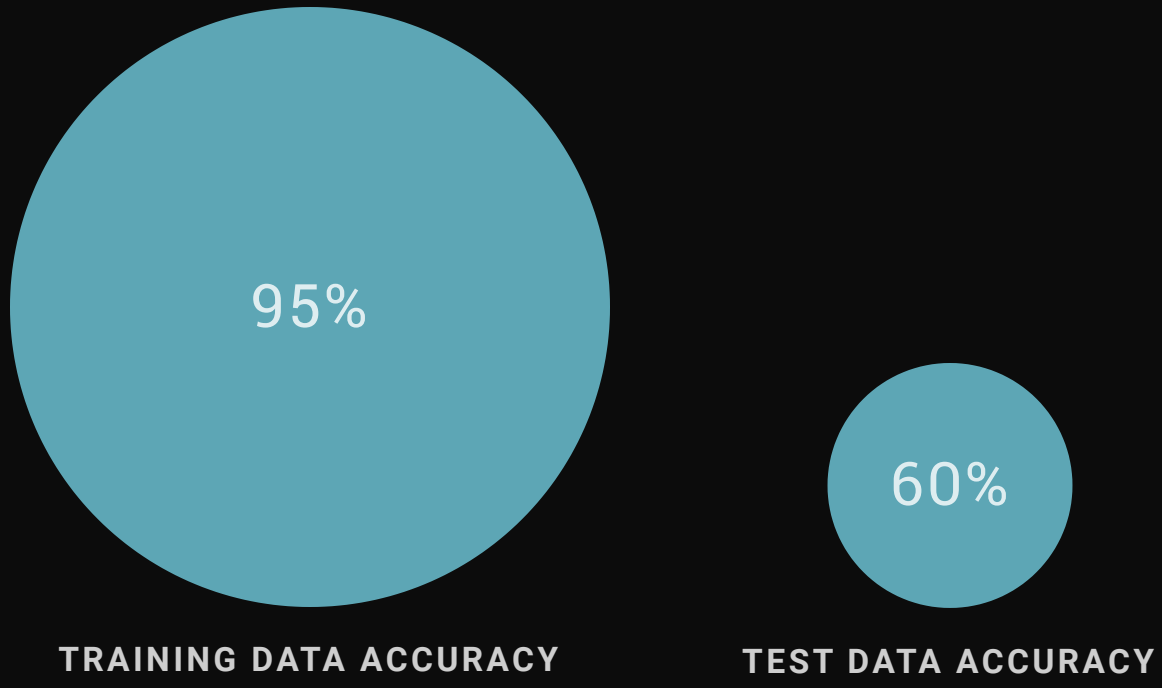
Initialize a Linear Regression model, which will learn the relationship between the features and the target variable.

Use the training data to train the Linear Regression model, allowing it to learn the coefficients (weights) that best fit the relationship between the features and house prices.

Assess the performance of the trained model on the training data to ensure it is learning the patterns in the data effectively.

Overfitting

Accuracy on Training vs. Test Data (%)



Overfitting Example

Overfitting occurs when a machine learning model learns the training data too well, capturing not only the true relationships but also the noise or randomness in the data. This results in the model performing exceptionally well on the training data but poorly on new, unseen data.

Signs of Overfitting

- **HIGH ACCURACY ON TRAINING DATA**

  The model achieves very high accuracy, such as 95% or higher, on the training data, indicating that it has learned the training data too well.

- **POOR PERFORMANCE ON TEST DATA**

  The model performs poorly on unseen test data, with accuracy significantly lower than on the training data, typically around 60% or less.

- **COMPLEX MODEL WITH TOO MANY FEATURES**

  The model is overly complex, with a large number of features or parameters that allow it to memorize the training data rather than learning the underlying patterns.

Solutions for Overfitting

REDUCE MODEL COMPLEXITY

Simplify the model architecture by reducing the number of features or the depth/complexity of the neural network, limiting the model's ability to memorize the training data.

APPLY REGULARIZATION TECHNIQUES

Use methods like Lasso (L1) or Ridge (L2) regularization to add a penalty for model complexity, encouraging a simpler and more generalizable model.

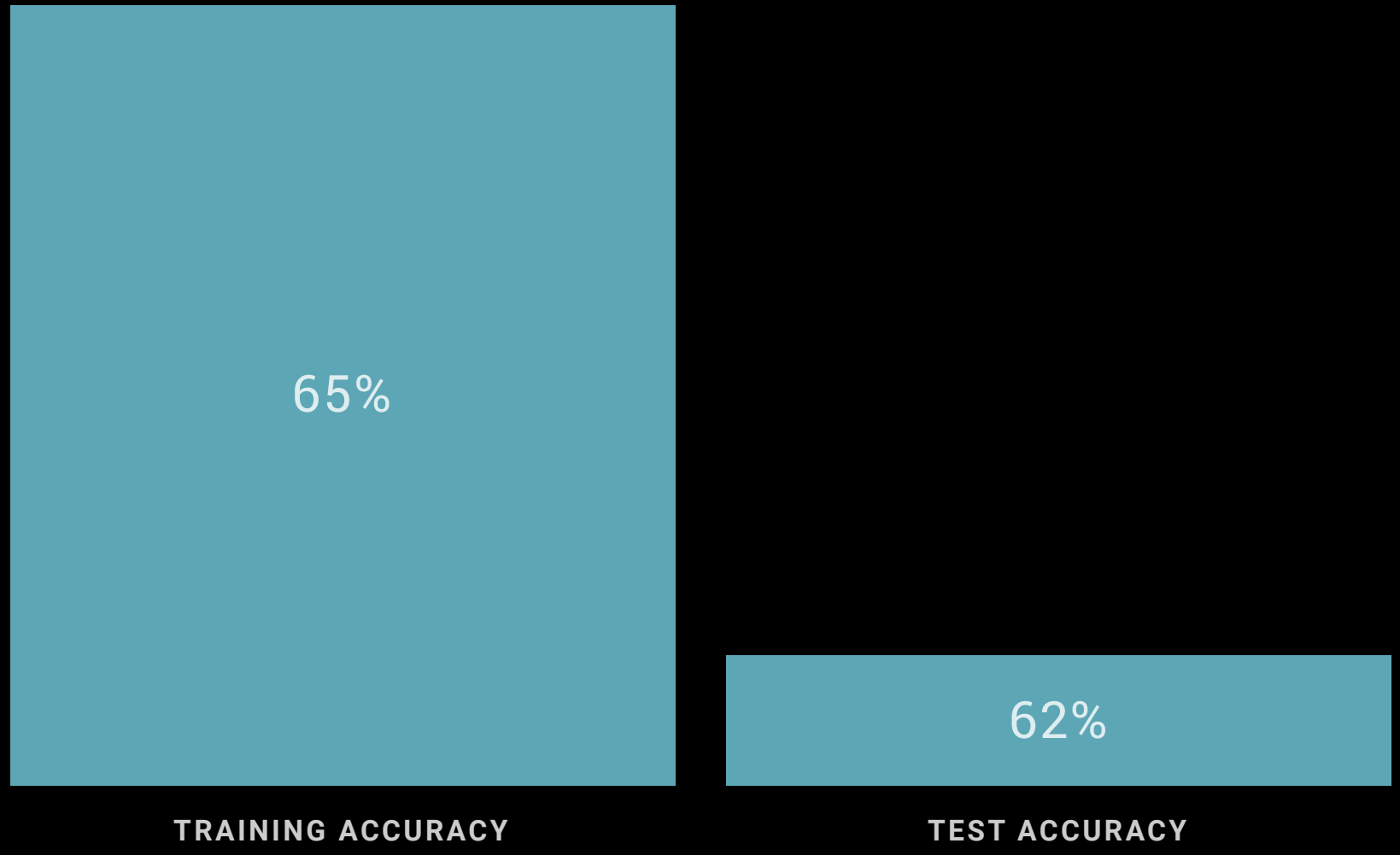
IMPLEMENT CROSS-VALIDATION

Perform cross-validation to get a more realistic estimate of the model's performance on unseen data, and tune hyperparameters to optimize for generalization.

BY IMPLEMENTING THESE SOLUTIONS, YOU CAN HELP YOUR MACHINE LEARNING MODEL GENERALIZE BETTER AND AVOID THE PITFALLS OF OVERFITTING, ENSURING IT PERFORMS WELL ON NEW, UNSEEN DATA.

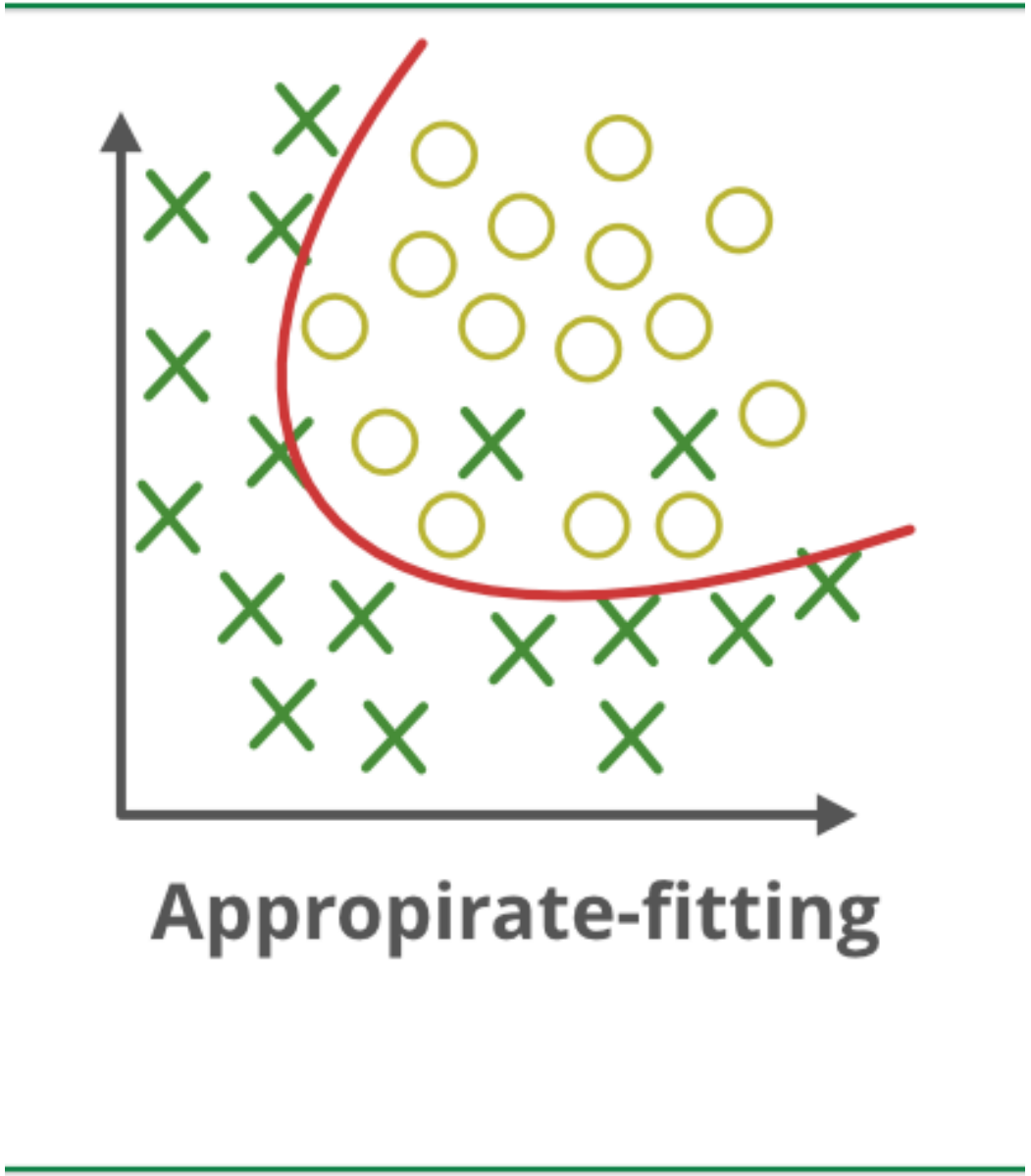
Underfitting

Accuracy on Training vs Test Data (%)



Underfitting Example

This slide provides an example of how underfitting can occur in a machine learning model when it is too simple to capture the underlying patterns in the data, resulting in poor performance on both the training and test data.



Signs of Underfitting

- **LOW ACCURACY ON TRAINING DATA**

  The model performs poorly on the training data, indicating it has not learned enough from the available data.

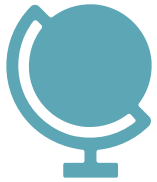
- **LOW ACCURACY ON TEST DATA**

  The model also performs poorly on unseen test data, confirming it has not generalized well to new examples.

- **OVERSIMPLIFIED MODEL**

  The model is too simple and lacks the complexity to capture the underlying patterns in the data, resulting in underfitting.

Solutions for Underfitting



USE A MORE COMPLEX MODEL

Increase the complexity of the machine learning model, such as using a higher-degree polynomial regression or a neural network, to better capture the underlying patterns in the data.

ADD MORE MEANINGFUL FEATURES

Identify and include additional relevant features that can provide more information to the model, allowing it to make more accurate predictions.

INCREASE TRAINING DURATION

Ensure the model is trained for a sufficient number of iterations or epochs, allowing it to converge and learn the underlying relationships in the data.

BY IMPLEMENTING THESE SOLUTIONS, YOU CAN ADDRESS THE ISSUE OF UNDERFITTING AND IMPROVE THE PERFORMANCE OF YOUR MACHINE LEARNING MODEL ON BOTH THE TRAINING AND TEST DATA.