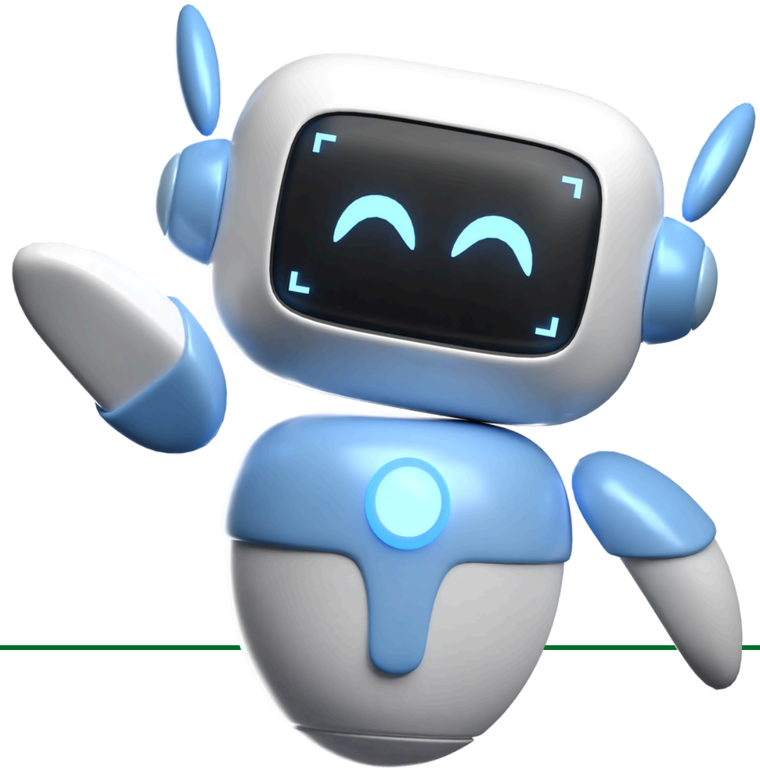
BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

IDEAS Emerging Technology Skills Scholarship Program

RECAP OF FUNDAMENTALS OF MACHINE LEARNING

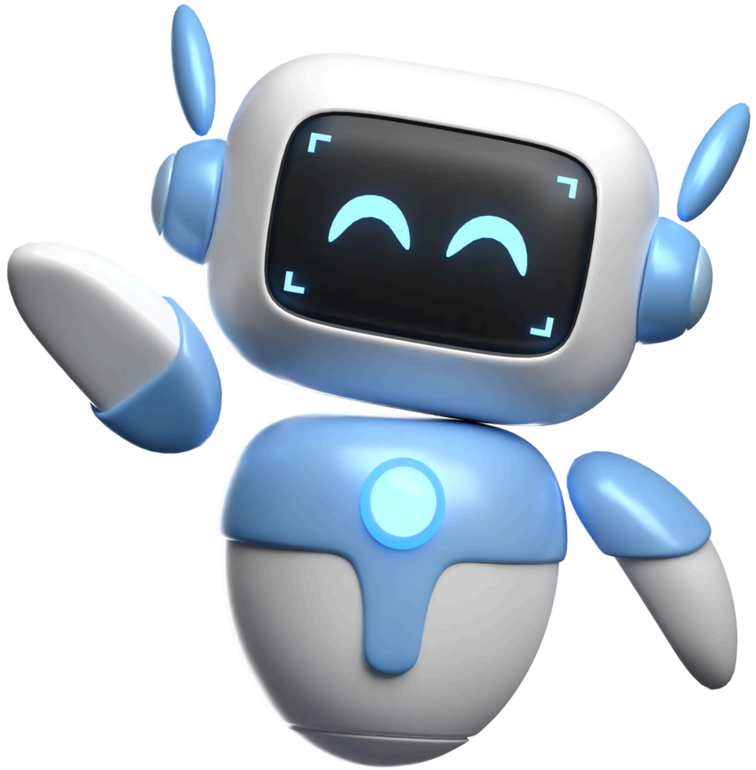
Presented by: Khadijah Saad Mohammed



BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

Content

- Recap
- Additional Code

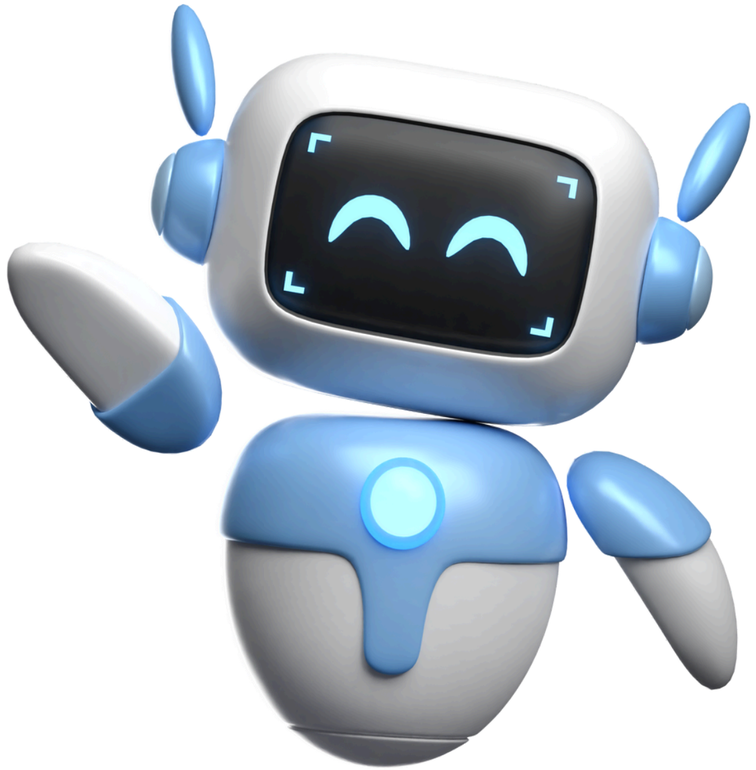


BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

Recap

We'll review the key concepts and techniques we've covered in this module, ensuring you have a solid foundation in machine learning.

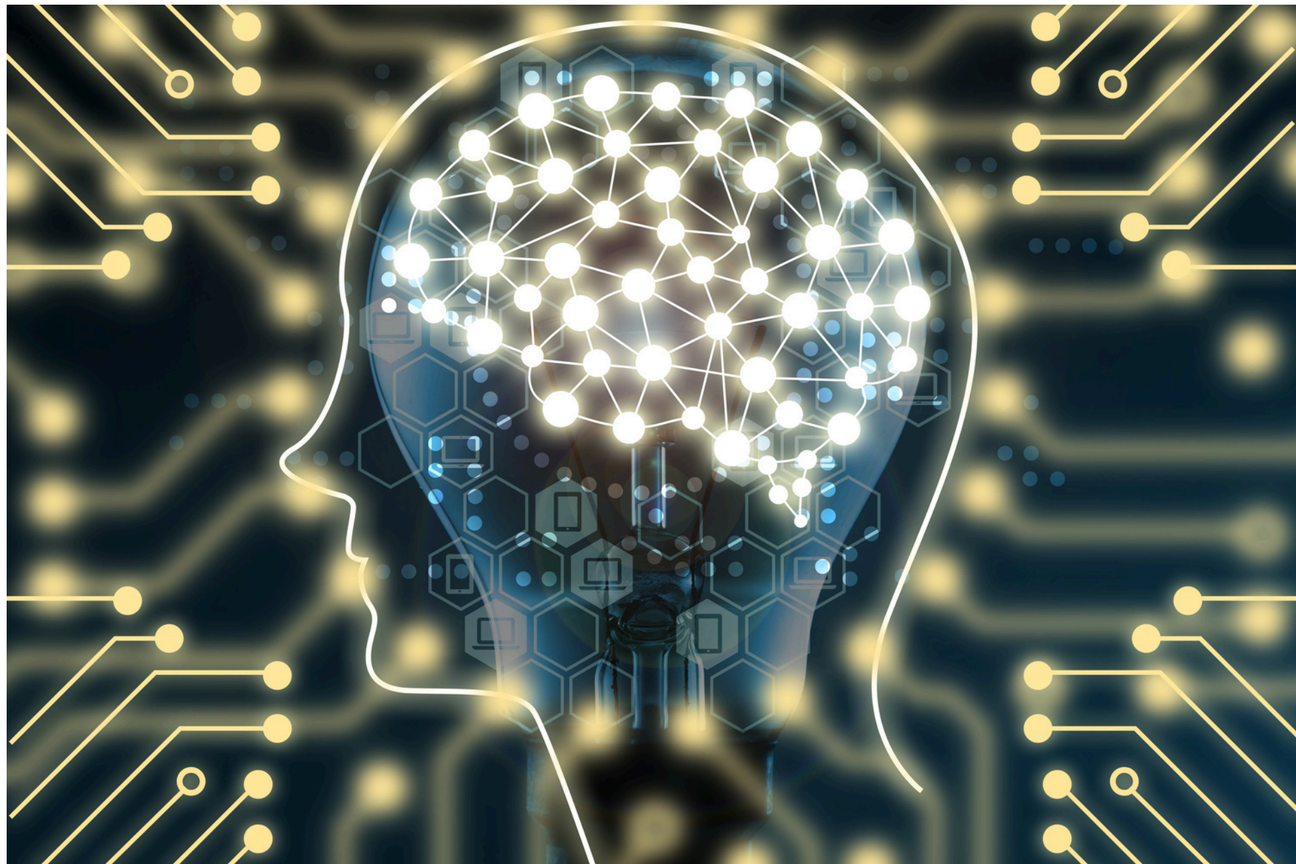
- Introduction
- Types of Machine Learning
- Common Algorithms
- Preparing Data and Engineering Features
- Evaluating Model Performance



BAZE UNIVERSITY ABUJA
IN PARTNERSHIP WITH
DOMINEUM

Machine Learning

Machine learning is a subset of artificial intelligence that involves training algorithms to predict outcomes, identify patterns, and make decisions based on data.



BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

Types of Machine Learning

- **Supervised Learning**: Where the model learns from labeled data.
- **Unsupervised Learning**: Where the model identifies patterns in unlabeled data.
- **Reinforcement Learning**: Where an agent learns to make decisions by performing actions and receiving rewards.

BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

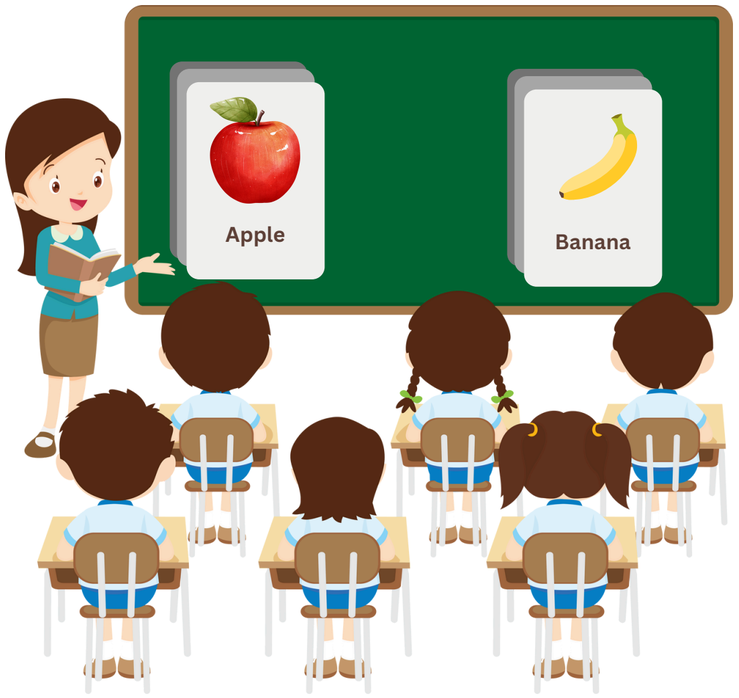
Types of Machine Learning

Supervised Learning

- Where the model learns from labeled data.

In supervised learning, the model uses input-output pairs to learn a function that can predict outcomes for new data.

- **Classification**: Predicting categorical labels (e.g., spam or not spam).
- **Regression**: Predicting continuous values (e.g., price of a house).
  - .



BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

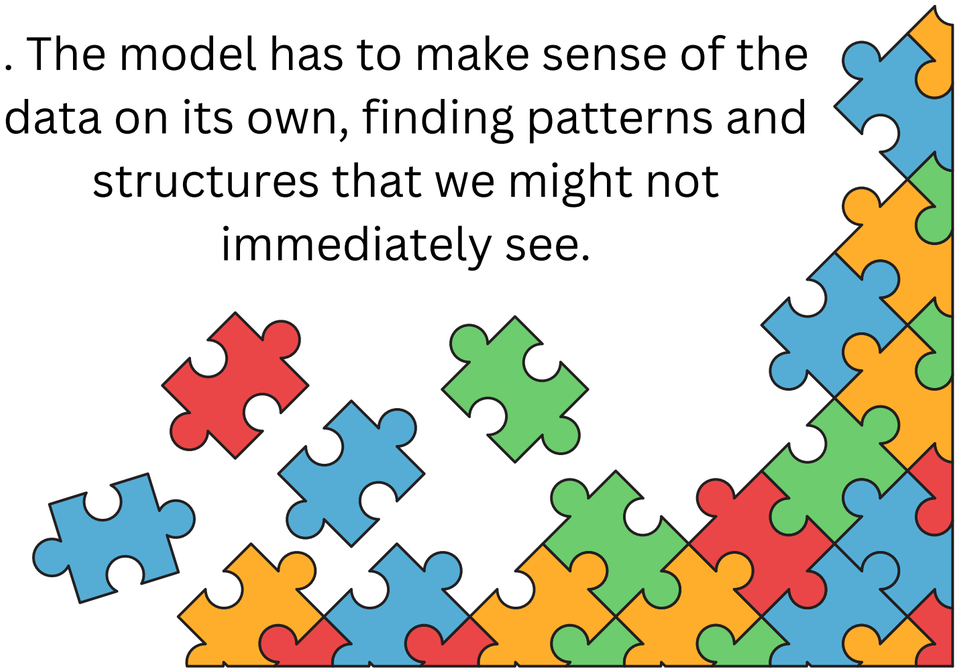
Types of Machine Learning

Unsupervised Learning

Unsupervised learning involves models that infer patterns from unlabeled data without reference to known or labeled outcomes.

- **Clustering**: Grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups.
- **Association**: Discovering rules that describe portions of your data, such as people that buy X also tend to buy Y.

. The model has to make sense of the data on its own, finding patterns and structures that we might not immediately see.

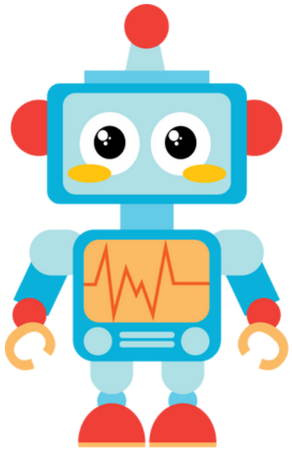


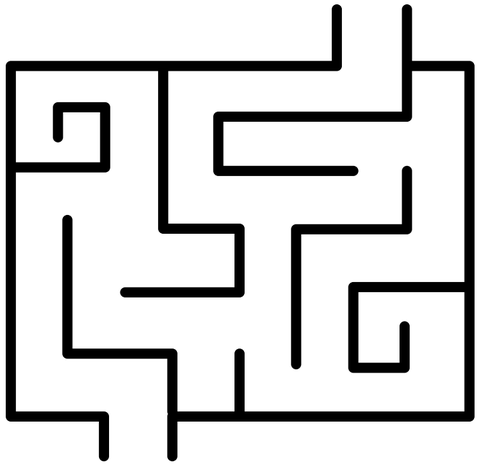
BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

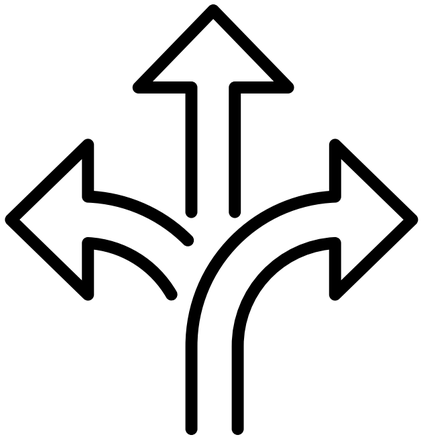
Types of Machine Learning

Reinforcement Learning

- :Where an agent learns to make decisions by performing actions and receiving rewards.







BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

Preparing Data and Engineering Features

Data Pre-processing

- **Cleaning**: Removing errors and inconsistencies from the data.
- **Transformation**: Scaling or normalizing features to ensure uniformity.
- **Encoding:** Converting categorical variables into numerical representations.
- **Feature Engineering:** Creating new features or transforming existing ones to improve model performance.

Model Evaluation

Why Evaluate a Model?

After training a model, you need to know how well it predi
new data

Basic Metrics

For Classifying Data (e.g., spam or not spam): Look at Accuracy (how many predictions were correct).

For Predicting Values (e.g., house prices): Use MSE (Mean Squared Error), which tells you how far off your predictions are on average.

		Actual Values: Positive (1)	Actual Values: Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

In classification, the data is split into training sets for learning and testing sets for evaluating model performance

BAZE UNIVERSITY ABUJA IN PARTNERSHIP WITH DOMINEUM

Model Evaluation

Evaluating a model's accuracy and performance is crucial for verifying its effectiveness.

- **Key Metrics**
  - For Classification: Accuracy, Precision, Recall, and F1 Score.
  - For Regression: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE).
- **Validation Techniques**
  - Train/Test Split
  - Cross-Validation: Using parts of the data to train and validate the model to ensure reliability.

Overfitting and Underfitting

What are These?

- **Overfitting**: Imagine memorizing answers for a test without understanding the concepts. You might fail on different questions. Overfitting is similar; the model performs well on training data but poorly on unseen data.
- **Underfitting:** This is like not studying enough. Underfitting means the model is too simple to learn the underlying pattern of the data.

Overfitting and Underfitting

- Overfitting: Occurs when a model learns the training data too well, including the noise, which hampers its performance on new data.
- Underfitting: Occurs when a model is too simple to capture the underlying data patterns, resulting in poor performance on both training and new data.

- **Solutions**:
  - Regularization: Techniques like L1 (Lasso) and L2 (Ridge) that help reduce overfitting by penalizing large coefficients.
  - Pruning: Used in decision trees to reduce the size of the tree and improve model simplicity.

Regularization Techniques

What is Regularization?

- Think of regularization as a way to prevent your model from studying "too hard" and just memorizing data. It gently nudges the model to be more general.

Best Practices in Machine Learning

- **Data Quality:** High-quality data is critical for building robust models.
- **Algorithm Selection**: There is no one-size-fits-all algorithm. Testing different algorithms based on the problem context is essential.

MORE CODING

The next slides contain snippets of more coding examples.

```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
model.fit(X_train, y_train)


predictions = model.predict(X_test)
from sklearn.metrics import accuracy_score
print("Accuracy:", accuracy_score(y_test, predictions))
```

More Data Processing

```
# Create a new column 'is_child' to indicate whether the passenger is a child
df['is_child'] = (df['Age'] < 18).astype(int)


# Fill missing values
df['Age'].fillna(df['Age'].median(), inplace=True)
df['Embarked'].fillna(df['Embarked'].mode()[0], inplace=True)
# Extract title from the Name column
df['Title'] = df['Name'].str.extract(' ([A-Za-z]+)\.', expand=False)


# Convert categorical variables using one-hot encoding
df = pd.get_dummies(df, columns=['Sex', 'Embarked', 'Title'], drop_first=True)


# Drop columns that are not needed
df.drop(['PassengerId', 'Name', 'Ticket', 'Cabin'], axis=1, inplace=True)
# Creating a new feature 'FamilySize'
titanic['FamilySize'] = titanic['SibSp'] + titanic['Parch'] + 1
```

Other Algorithms

```
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
```

Accuracy with other Algorithms

```
# Logistic Regression
log_reg = LogisticRegression()
log_reg.fit(X_train, y_train)
log_reg_preds = log_reg.predict(X_test)
print("Logistic Regression Accuracy:", accuracy_score(y_test, log_reg_preds))

# Random Forest Classifier
rf_clf = RandomForestClassifier()
rf_clf.fit(X_train, y_train)  # no need to scale data for tree-based models
rf_preds = rf_clf.predict(X_test)
print("Random Forest Accuracy:", accuracy_score(y_test, rf_preds))

# Support Vector Machine
svm = SVC()
svm.fit(X_train, y_train)
svm_preds = svm.predict(X_test)
print("SVM Accuracy:", accuracy_score(y_test, svm_preds))
```

BAZE UNIVERSITY ABUJA

IN PARTNERSHIP WITH

DOMINEUM

THANK YOU

Q&A