Build Your First ML Project – From Data to Predictions

Throwback

1. Getting Started with Python
2. Jupyter Notebook & Google Colab
3. Variables, Loops & Functions in Python
4. Introduction to NumPy
5. Introduction to Pandas
6. Data Cleaning
7. Data Visualization with Matplotlib
8. Git & GitHub Overview
9. Resources
10. Next Lesson

Table of Contents

Learning Outcome

- **Understand the fundamentals of machine learning** — including datasets, features, labels, and the overall ML workflow from data preprocessing to model prediction.
- **Differentiate between key types of ML tasks** — such as regression and classification — and gain a light introduction to evaluating models using metrics like accuracy and the confusion matrix.
- **Gain hands-on experience with building ML systems** — including implementing workflows in notebooks and showcasing reproducible, collaborative projects using tools like Google Colab.

What is a Dataset?

- A dataset is a structured collection of data.
- In machine learning, it's typically organized in rows and columns (like a table).
- Each row = one data point (example).
- Each column = a feature, or in some cases, the label.
- Can be image, text, audio as well.

Features vs Labels

- **Features** = Input data (independent variables)
    - Used by the model to learn patterns.
- **Label** = Target or output (dependent variable)
    - What the model tries to predict.

In supervised learning, we split the dataset into features and labels. Features are what we feed into the model, and the label is what we want the model to predict.

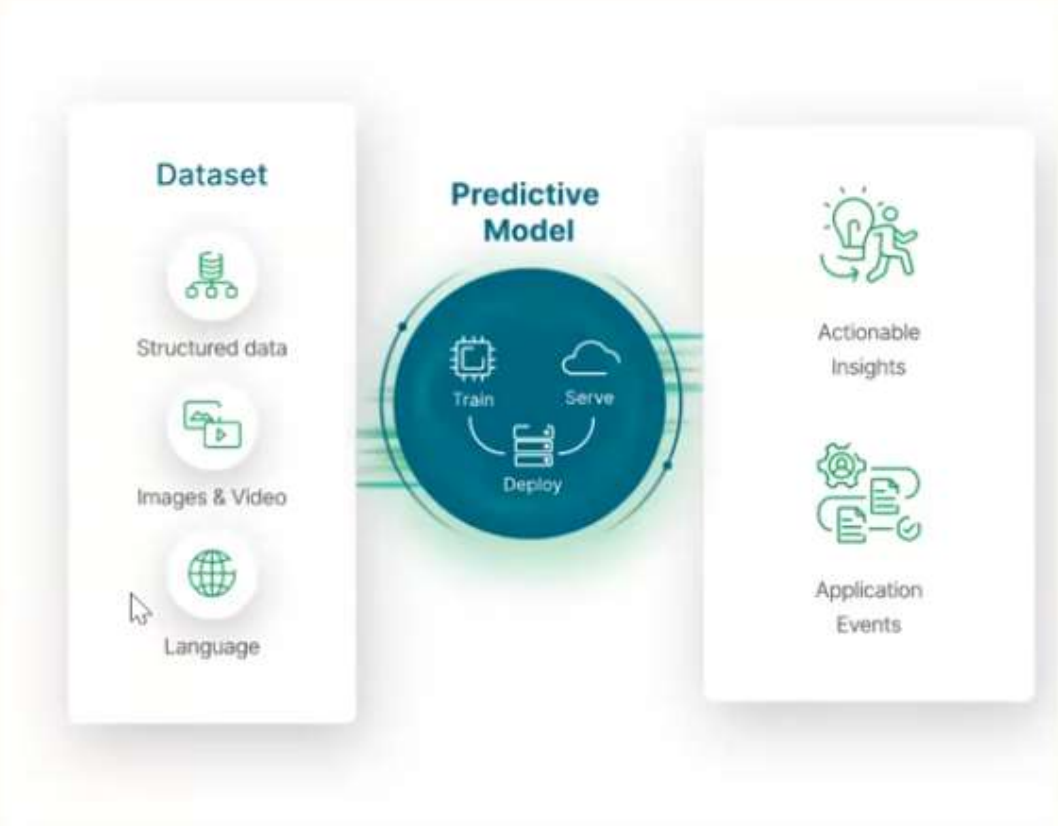
Features vs Labels (Example)

Size (sq ft)	Bedrooms	Age (years)	Price ($)
1000	2	10	200,000
1500	3	2	320,000
1200	2	8	250,000
1800	3	20	275,000

- **Features**: Size, Bedrooms, Age
- **Label**: Price

Machine Learning Workflow

- A machine learning workflow is the step-by-step process used to build, train, and use a machine learning model.
- It starts with data and ends with a prediction.
- Helps standardize and organize ML development.



Types of ML: Regression vs Classification

Regression

- Predicts continuous numerical values
- Example use cases:
  - Predicting house prices
  - Forecasting stock prices
  - Estimating temperature

Classification

- Predicts categories or classes
- Example use cases:
  - Email spam detection (Spam / Not Spam)
  - Image recognition (Cat / Dog)
  - Disease prediction (Positive / Negative)

Evaluation Metrics: Accuracy

- Accuracy = (Correct Predictions) ÷ (Total Predictions)
- Example: If a model makes 90 correct predictions out of 100 → Accuracy = 90%
- Most useful when classes are balanced
- Problem: Accuracy can be misleading when data is imbalanced
- Example: If 95% of emails are not spam, a model that always predicts 'not spam' will be 95% accurate — but still useless.

Evaluation Metrics: Confusion Matrix

	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

- A confusion matrix shows how predictions are distributed
- For binary classification (e.g. Spam/Not Spam):

- **True Positive (TP)**: Correctly predicted positive
- **True Negative (TN)**: Correctly predicted negative
- **False Positive (FP)**: Incorrectly predicted positive
- **False Negative (FN)**: Incorrectly predicted negative