AI for Product Managers:
Model Evaluation for Classification

Classification

In classification, every item in a dataset is assigned a **class** or **label**.

Training a Classification Model

- Requires a training dataset of ***(input, actual label)*** pairs.
- Model is fed these pairs.
- Model processes the ***inputs***.
- Model produces ***predicted labels***.
- Model updates its weights to minimize the mismatch between ***actual labels*** and ***predicted labels***.

Binary Classification

- **Actual label** has only two values representing the presence or absence of something.
- Example: in diagnosing pneumonia, the labels are *pneumonia* or *normal*.

Positive and Negative Labels

- Positive (actual) labels represent the presence of a condition/pattern/quality.
- Negative (actual) labels represent the absence of the condition/pattern/quality.

- ***#Dataset = #Actual positives + #Actual negatives***[†]

Actual Positives	
Actual Negatives	

Evaluating a Classification Model

- Requires a validation or test dataset of ***(input, actual label)*** pairs.
- The model is fed only inputs, the actual labels are withheld: ***(input,*** *actual label****)***.
- Model processes the ***inputs***.
- Model produces ***predicted labels.***
- Evaluation compares ***predicted labels*** against withheld ***actual labels***.

True Positives and False Negatives

- When a model classifies an actual positive:
  - It either correctly classifies the input as a positive (true positives).
  - Or misclassifies it as a negative (false negative).

	Predicted as Positives	**Predicted as Negative**
Actual Positives	True Positives(TP)	False Negatives(FN)

[†] # represents count

** Some software may produce a rotated table, in which actual positives and negatives are in the columns and predicted positives and negatives are in the rows.

#Actual positives = #True positives + #False negatives†

True Negatives and False Positives

- When a model classifies an actual negative:
  - It either correctly classifies the input as a negative (true negative).
  - Or misclassifies it as a positive (false positive).

	Predicted as Negatives	**Predicted as Positive**
Actual Negatives	True Negatives (TN)	False Positives (FP)

#Actual negatives = #True negatives + #False positives†

Remembering the Correspondence

- *Actual label == Predicted label => **True** (Positive/Negative)*
- *Actual label **!=** Predicted label => **False** (Positive/Negative)*

The Confusion Matrix

- Represents the outcome of a model's evaluation.
- For binary classifiers, it is a 2x2 matrix or table.

**	**Predicted Positives**	**Predicted Negatives**
Actual Positives	***TP***	***FN***
Actual Negatives	***FP***	***TN***
TP+FN = total number of positive inputs in the dataset TN+FP = total number of negative inputs in the dataset TP+FN+TN+FP = size of the dataset		

Evaluation Metrics

- **Accuracy**: Fraction of correct predictions.

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

- **Precision:** Ratio of true positives to predicted positives.

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

- **Recall:** Ratio of true positives to actual positives.

$$Recall = \frac{TP}{(TP + FN)}$$

- **F1 Score** combines precision and recall into a single number, which makes comparing two models easier.

$$F1 = \frac{2 * Precision * Recall}{(Precision + Recall)}$$

Multiclass Classifiers

- **Actual labels** have more than 2 values.
- E.g. Car, Bus, Truck.
- Metrics are calculated for each of the classes and then their average is taken.

† # represents count

** Some software may produce a rotated table, in which actual positives and negatives are in the columns and predicted positives and negatives are in the rows.