Real world application

Context

- Key steps in real world ML application
  - The ML pipeline
  - Data preprocessing
  - Evaluation metrics
  - PCA for visualization
- Exercise: https://drive.google.com/file/d/1tHEhEM1ESvnf7u-M423v1fhqSOXWRG4C/view?usp=sharing

The big picture for ML pipeline

- Raw data
- Preprocessing
  - Missing data
  - Categorical features
  - Feature scaling
- Train/test split
- Model training
- Evaluation
  - Accuracy, precision, recall, F1-score

Preprocessing

- Missing data
  - Drop missing data when too many values are missing
  - Imputation with median for numerical, mode for categorical
- Categorical features
  - Label encoding: use for binary or ordinal (where the categories have a natural, ranked order or hierarchy)
  - One-hot encoding: A/B/C – [1,0,0]/[0,1,0]/[0,0,1]
- Feature scaling
  - without scaling, feature with larger scale will dominate the model learning and other features becomes irrelevant
  - E.g in KNN, every distance calculation is dominated by feature with larger scale

Evaluation metrics

Confusion matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

$$\text{Accuracy} = \frac{\text{correct classifications}}{\text{total classifications}} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Recall (or TPR)} = \frac{\text{correctly classified actual positives}}{\text{all actual positives}} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{\text{correctly classified actual positives}}{\text{everything classified as positive}} = \frac{TP}{TP + FP}$$

F1 score is the harmonic mean (a kind of average) of precision and recall.

$$\mathbf{F1 = 2 * \frac{precision * recall}{precision + recall} = \frac{2TP}{2TP + FP + FN}}$$

Why we need different metrics?

- Different problems care about different mistakes

  Medical diagnosis, missing a disease (false negative) is critical, recall matters

  Spam detection, marking good emails as spam (false positive) is annoying, precision matters

- Accuracy can be misleading

  A model can achieve high accuracy by predicting only the majority class

- Metrics emphasize different aspects of performance

  Accuracy: overall correctness

  Precision: When I predict positive, how often am I right?

  Recall: How many actual positives did I find?

  F1-score: balance between precision and recall

- Model selection depends on the metric

  The metric directly influences which model you select

PCA for visualization

- What?

  Principal Component Analysis (PCA) is a technique used to **reduce the number of features (dimensions)** in a dataset while keeping as much important information as possible.

- Why?

  Humans can only easily visualize in **2D or 3D**, but real datasets often have **many dimensions (features).** We can't directly "see" patterns in high-dimensional space. PCA helps by reducing data to **2 or 3 dimensions, w**hile preserving the **most important structure.**

- How?
  - Standardize the data
  - Apply PCA
  - Visualization
- Case: https://setosa.io/ev/principal-component-analysis/