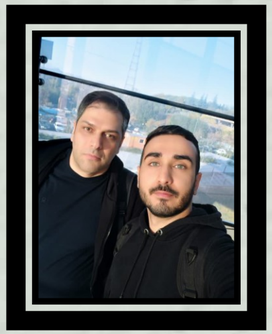
Teaching Assistantt :
Saeed Babagolzadeh & Hamed Jamshidi
Associate Professor : Dr.Mirzarezaee



Machine learning :
KNN & Decision Trees

دانشگاه آزاد اسلامی واحد علوم و تحقیقات
Islamic Azad University - Science and Research Branch (SRBIAU)

دانشگاه آزاد اسلامی واحد علوم و تحقیقات

Introduction to Machine Learning Models (KNN and Decision Trees)

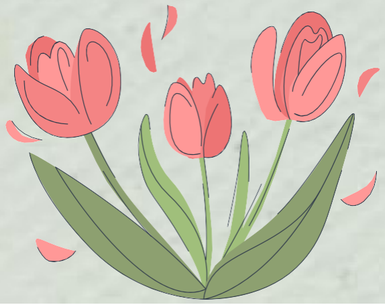
KNN (K-Nearest Neighbors) is a supervised learning algorithm used for classification and regression. It classifies data points based on the majority class among their nearest neighbors. Decision Trees, on the other hand, split data into branches based on feature values. Each node represents a decision, leading to an outcome. Both algorithms are intuitive and widely used. They handle different types of data effectively, making them popular choices in machine learning.

Classification Metrics

Classification problems in machine learning revolve around categorizing data points into predefined classes or groups. For instance, determining whether an email is spam is a classic example of a binary classification problem. As the complexity of the data and the number of classes increases, so does the intricacy of the model. However, building a model is only half the battle. Key metrics like accuracy, precision, and recall from the confusion matrix are essential to assess its performance. Metrics provide insights into how well the model achieves its classification goals. They help identify improvement areas to show if the model aligns with the desired outcomes. Among these metrics, accuracy, precision, and recall are foundational.

What is a Confusion Matrix?

A **confusion matrix** is a powerful tool for visualizing the performance of a classification model. It summarizes the true positives, false positives, true negatives, and false negatives, allowing for a comprehensive evaluation of model accuracy and error types.



The Confusion Matrix

Actual Values \ Predicted Values	0	1
0	True Negative y_true : 0 y_pred : 0	False Positive y_true : 0 y_pred : 1
1	False Negative y_true : 1 y_pred : 0	True Positive y_true : 1 y_pred : 1

What is Accuracy in Machine Learning?

Accuracy is a fundamental metric in classification, providing a straightforward measure of how well a model performs its intended task.
Accuracy represents the ratio of correctly predicted instances to the total number of instances in the dataset. In simpler terms, it answers the question: "Out of all the predictions made, how many were correct?"

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

Diving into Precision

Precision is a pivotal metric in classification tasks, especially in scenarios with a high cost of false positives. It provides insights into the model's ability to correctly predict positive instances while minimizing the risk of false alarms.Precision, often referred to as the positive predictive value, quantifies the proportion of true positive predictions among all positive predictions made by the model. It answers the question:"Of all the instances predicted as positive, how many were positive?"

$$\text{Precision} = \frac{TP}{TP + FP}$$

TP = True Positives

FP = False Positives

What is Recall?

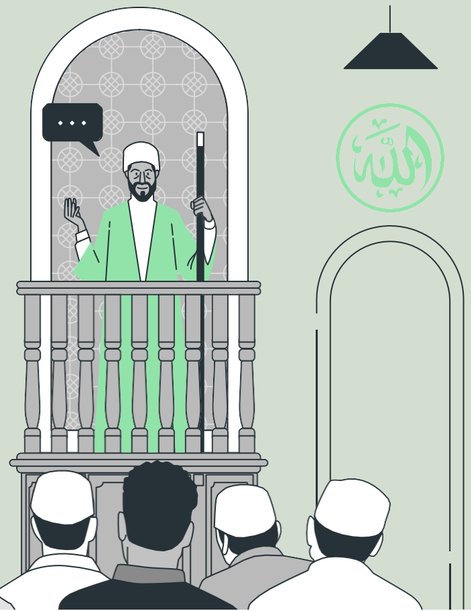
Recall, also known as sensitivity or true positive rate, is a crucial metric in classification that emphasizes the model's ability to identify all relevant instances. Recall measures the proportion of actual positive cases correctly identified by the model. It answers the question: "Of all the actual positive instances, how many were correctly predicted by the model?"

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

It is better to know what F1 Score is.

F1 Score is a harmonic means between the Precision and Recall score.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$



The metric would give a single number that would take account of both false positive and false negative cases.

Choosing the Right Metric

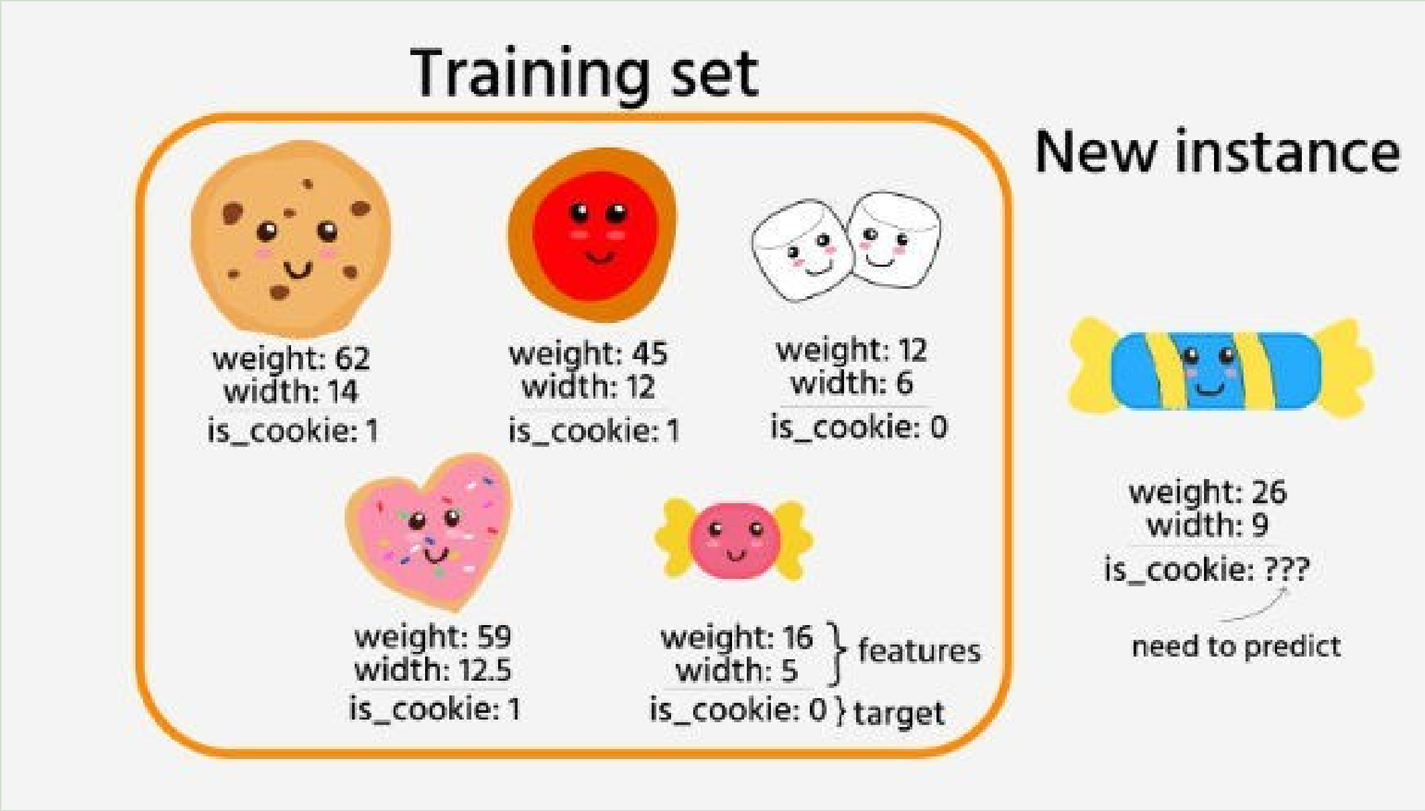
Choosing the right metric depends on the problem context. For instance, in medical diagnoses, **recall** may be prioritized to minimize false negatives, while in spam detection, **precision** could be more critical to reduce false positives. Understanding the trade-o s is essential.

Common Challenges in Evaluation

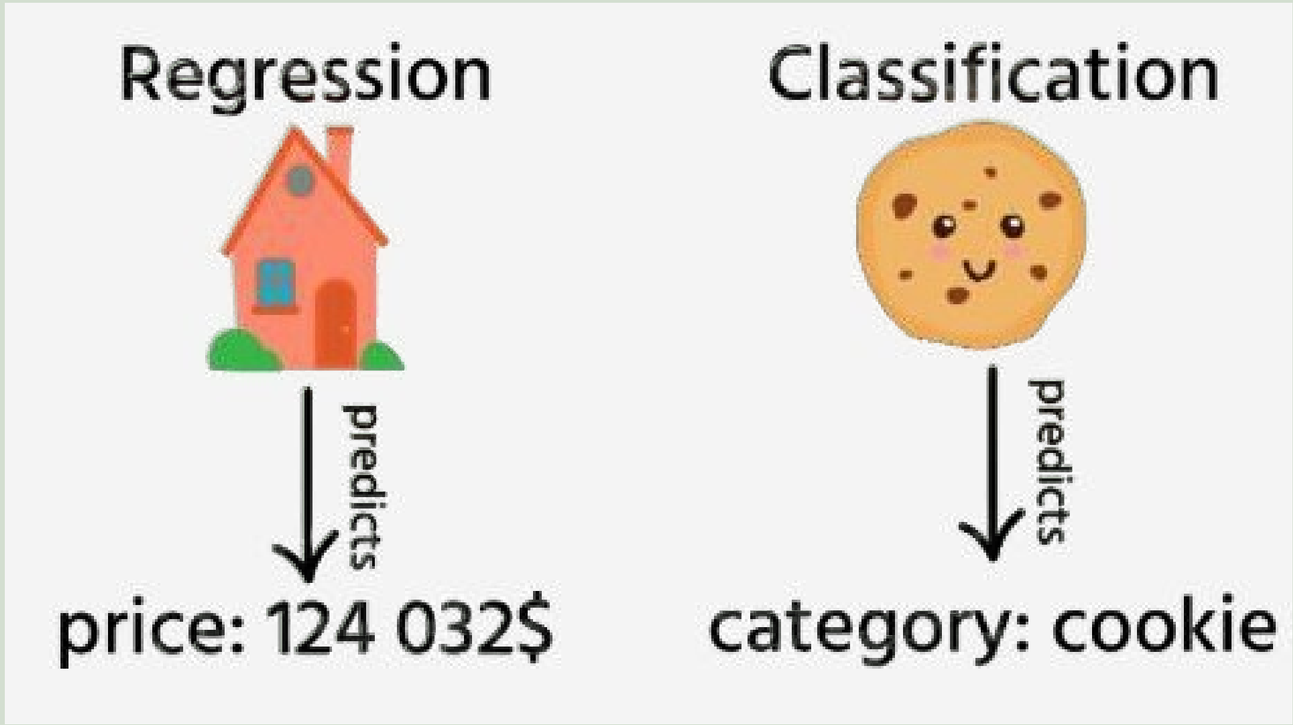
Evaluating machine learning models presents challenges such as **class imbalance**, overfitting, and underfitting. Addressing these issues is vital for accurate model assessment and ensuring the model generalizes well to unseen data.

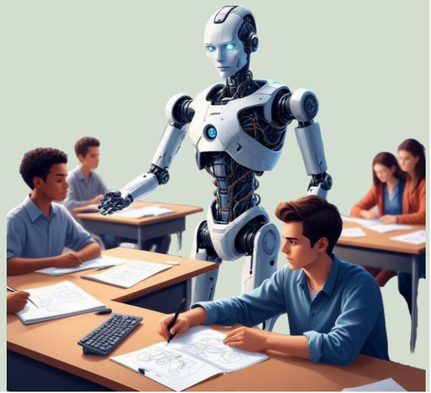
What is Classification?

Classification is a supervised learning task. Its goal is to predict the class to which the instance belongs based on a set of parameters(features). You need to give many labeled examples of data(called training set) for the computer to learn before it can predict the class of a new instance.



The difference between classification and regression is that regression predicts a continuous numerical value, for example, a price. It can be any real (only positive for a price) number. In contrast, classification predicts a categorical value, for example, the type of a sweet. There is a finite set of values, and the model tries to classify each instance into one of these categories.



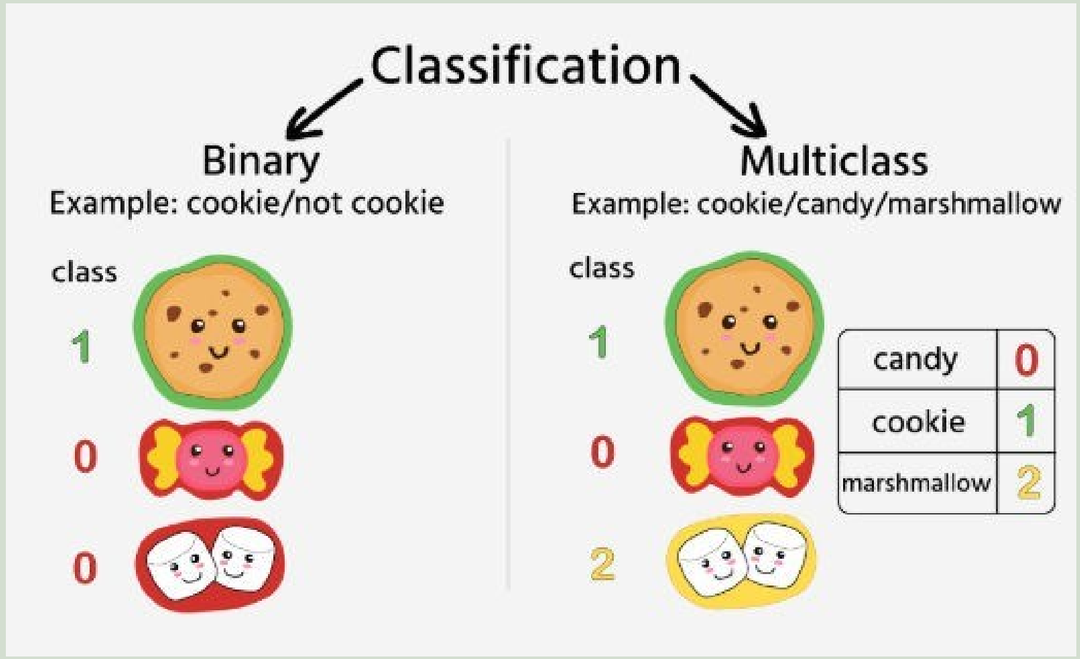


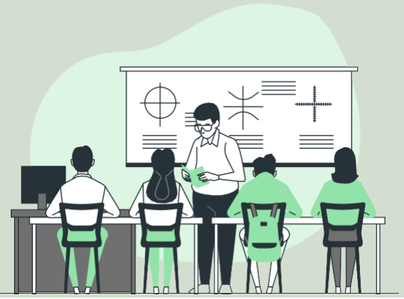
Based on the formulation of a problem, there are two types of classification:

- Binary classification: In binary classification, a target is one of two possible outcomes. For example, email: spam/not spam, sweet: cookie/not cookie;

- Multi-class Classification: In Multi-class Classification, there are three or more possible outcomes for a target. For example, email: spam/important/ad/other, sweet: cookie/marshmallow/candy.

For most ML models, you need to encode the target to a number. For binary classification, outcomes are usually encoded as 0/1 (e.g., 1 – cookie, 0 – not a cookie). For a multi-class classification, outcomes are usually encoded as 0, 1, 2, … (e.g., 0 – candy, 1 – cookie, 2 – marshmallow)





Many different models perform classification. we will discuss the following models:

- k-Nearest Neighbors;
- Decision Tree;

k-Nearest Neighbors;

KNN

K-Nearest Neighbors (KNN) is a supervised machine learning algorithm used for classification and regression tasks. Instead of building a predictive model, it makes predictions based on the similarity of data points. The core idea is that similar data points are often close to each other.

How Does KNN Work?

1. Store Training Data:

KNN keeps all the training data in memory. There's no explicit training phase. This is why KNN is called a lazy learning algorithm.

2. Find the Distance:

For a new data point, KNN calculates the distance between it and every point in the training set using metrics like:

- Euclidean Distance: The straight-line distance.
- Manhattan Distance: The sum of absolute differences in each dimension.
- Minkowski Distance: A generalization of distance measures.

3. Select the K Nearest Neighbors:

It identifies the k closest data points to the query point. The value of k is a parameter you choose (e.g., k=3).

4. Make a Prediction:

- For Classification: KNN assigns the most common class (majority vote) among the k neighbors.
- For Regression: KNN calculates the average of the target values of the k neighbors.

Key Concepts

- Lazy Learning: KNN doesn’t create a model during training; all computations are deferred to the prediction phase.
- Distance-Based: The algorithm depends heavily on the choice of the distance metric.
- Hyperparameter K: The number of neighbors (k) is critical. A small k makes the algorithm sensitive to noise, while a large k may oversimplify.

Advantages of KNN

- Easy to Implement:

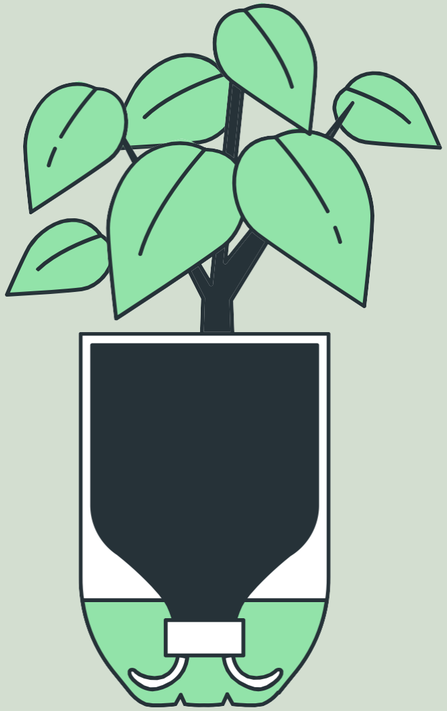
Simple to understand and requires no complex training process.

- Flexible:

Works for both classification and regression.

- Non-Parametric:

Makes no assumptions about the underlying data distribution.



Disadvantages of KNN

Computationally Expensive: Requires calculating distances to all points in the dataset for every query, which can be slow for large datasets.

Sensitive to Noisy Data: Outliers or irrelevant features can affect predictions.

Scaling Required: Features with larger values dominate the distance calculation, so normalization is necessary.

Applications of K-Nearest Neighbors (KNN)

1. Image Classification: Used to classify objects in images, such as identifying handwritten digits.
2. Recommendation Systems: Suggests products or media based on user preferences and behavior.
3. Medical Diagnosis: Predicts diseases like diabetes or cancer based on patient data.
4. Anomaly Detection: Flags unusual patterns, such as fraud in banking or cybersecurity threats.
5. Customer Segmentation: Groups customers into categories to improve targeted marketing.
6. Pattern Recognition: Identifies handwriting, speech, or signatures in authentication systems.
7. Agriculture: Classifies soil types or detects pests for smarter farming.
8. Financial Analysis: Predicts stock trends or assesses credit risk for loan decisions.

Decision Trees

Understanding Decision Trees

Imagine you're trying to decide whether to go outside or stay inside. You might consider factors like the weather, temperature, and whether you have homework. This decision-making process can be represented as a decision tree.

A decision tree is a flowchart-like structure used to visualize decision-making processes. It starts with a root node (the initial decision point) and branches out into different possibilities. Each branch can lead to another decision node or a final decision (a leaf node).

ID3 and C4.5 Algorithms

ID3 and C4.5 are algorithms used to build decision trees from data. They work by selecting the best attribute at each node to split the data.

- ID3 (Iterative Dichotomiser 3):

This algorithm selects the attribute with the highest information gain to split the data. Information gain measures how much information is gained by splitting the data on a particular attribute.

- C4.5 (Classification and Regression Tree):

This algorithm is an improvement over ID3. It uses a metric called gain ratio, which is similar to information gain but penalizes attributes with many values. C4.5 can also handle missing values and continuous attributes.

Key Differences Between ID3 and C4.5

Random Forest

A random forest is an ensemble learning method that combines multiple decision trees. It works as follows:

- Random Sample Selection:

A random subset of data is selected from the original dataset.

- Decision Tree Creation:

A decision tree is built on the selected subset.

- Repeat:

Steps 1 (Understanding Decision Trees) and 2 (ID3 and C4.5 Algorithms) are repeated multiple times to create a forest of decision trees.

- Prediction:

To make a prediction, each decision tree in the forest votes, and the majority vote is taken as the final prediction.

Why is random forest better?

Random forests often outperform ID3 and C4.5 because:

Reduced Overfitting:

By using multiple trees, random forests are less prone to overfitting the training data.

Handles Missing Values and Continuous Attributes:

Random forests can handle both missing values and continuous attributes effectively.

Improved Accuracy:

Due to the ensemble nature of random forests, they often achieve higher accuracy than single decision trees.

Conclusion

(Decision trees)

Decision trees are a powerful tool for classification and regression tasks. Random forests, as an ensemble method, offer significant advantages over individual decision trees like those built by ID3 and C4.5.They are more robust, accurate, and versatile, making them a popular choice for various machine learning applications.

Conclusion

Based on the results obtained from the project code

Algorithm	Standardized	Balanced	Selected features	ACC	Recall	Precision	AUC
KNN	✗	✗	✗	0.9983	0.02	0.9991	0.53
KNN (K=7)	✓	✓	✗	0.9998	0.84	0.61	0.92
KNN	✗	✗	✓	0.9995	0.77	0.94	0.89
KNN (K=9)	✓	✓	✓	0.9982	0.88	0.48	0.94
ID3	✗	✗	✗	0.9991	0.73	0.78	0.86
ID3	✓	✓	✗	0.9991	0.79	0.72	0.90
ID3	✗	✗	✓	0.9968	0.73	0.82	0.86
ID3	✓	✓	✓	0.9992	0.79	0.75	0.90
ID3 - max depth 3	✓	✓	✗	0.9481	0.91	0.03	0.93

Algorithm	Standardized	Balanced	Selected features	ACC	Recall	Precision	AUC
C4.5	✗	✗	✗	0.9991	0.73	0.78	0.86
C4.5	✔	✔	✗	0.9990	0.76	0.68	0.88
C4.5	✗	✗	✔	0.9967	0.72	0.74	0.86
C4.5	✔	✔	✔	0.9992	0.77	0.74	0.89
C4.5 – max depth 7	✔	✔	✗	0.9875	0.84	0.10	0.91
RF	✗	✗	✗	0.9995	0.76	0.96	0.88
RF	✔	✔	✗	0.9996	0.81	0.96	0.90
RF	✗	✗	✔	0.9995	0.78	0.97	0.89
RF	✔	✔	✔	0.9995	0.83	0.90	0.92
RF – max depth 9	✔	✔	✗	0.9993	0.89	0.73	0.94

Thanks!

Do you have any questions?

Hamed.jm99@gmail.com
Golzadeh.saeed@gmail.com


Saeed Babagolzadeh


MrGolzadeh


Saeed Babagolzadeh