Inf620 - Lecture 7 - Nearest Neighbor - KNN

Department of Computer Science - UFV

Ricardo Ferreira
ricardo@ufv.br

2025

Introduction

- Class Material (click here for Colab)
- **Review**: Supervised Learning.
- **Problems**: Classification or Regression
- KNN Technique:
    - Majority of Nearest Neighbors
    - Explore various examples

Review

- **Supervised Learning**
  - Given a set of examples that are pairs [input, output], an algorithm must find (or learn) a rule that performs well in predicting the output for a new (unseen) input.
- **Classification**
  - Determine to which (categorical) class a given observation belongs.
- **Regression**
  - Model the relationship between independent and dependent variables. The output, in this case, is continuous.

Review

- **Supervised Learning**
  - Given a set of examples that are pairs [input, output], an algorithm must find (or learn) a rule that performs well in predicting the output for a new (unseen) input.
- **Classification**
  - Determine to which (categorical) class a given observation belongs.
- **Regression**
  - Model the relationship between independent and dependent variables. The output, in this case, is continuous.

Review

- **Supervised Learning**
  - Given a set of examples that are pairs [input, output], an algorithm must find (or learn) a rule that performs well in predicting the output for a new (unseen) input.
- **Classification**
  - Determine to which (categorical) class a given observation belongs.
- **Regression**
  - Model the relationship between independent and dependent variables. The output, in this case, is continuous.

Review

- **Supervised Learning**
  - Given a set of examples that are pairs [input, output], an algorithm must find (or learn) a rule that performs well in predicting the output for a new (unseen) input.
- **Classification**
  - Determine to which (categorical) class a given observation belongs.
- **Regression**
  - Model the relationship between independent and dependent variables. The output, in this case, is continuous.

Review

- **Supervised Learning**
  - Given a set of examples that are pairs [input, output], an algorithm must find (or learn) a rule that performs well in predicting the output for a new (unseen) input.
- **Classification**
  - Determine to which (categorical) class a given observation belongs.
- **Regression**
  - Model the relationship between independent and dependent variables. The output, in this case, is continuous.

Review

- **Supervised Learning**
  - Given a set of examples that are pairs [input, output], an algorithm must find (or learn) a rule that performs well in predicting the output for a new (unseen) input.
- **Classification**
  - Determine to which (categorical) class a given observation belongs.
- **Regression**
  - Model the relationship between independent and dependent variables. The output, in this case, is continuous.

Example or Instance or Sample

- Formally, an example or a sample is a pair $[x, f(x)]$, where x is the input and $f(x)$ is the output of the unknown function applied to x.
- An attribute is a characteristic. It can have a continuous or discrete value or a symbol with qualitative value.
- An example is said to be composed of the values of several attributes.
- An example may be called a feature vector.

Examples of Supervised Problems

- A set of photos with information about what is in them, and you train a model to recognize new photos.
- A set of molecules with information about which are drugs, and you train a model to determine whether new molecules are also drugs.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of** k:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of** k:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
    - Computes the distance between the test point and all training points.
    - Selects the k nearest neighbors.
    - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
    - Small k: more sensitive to noise.
    - Large k: may overly smooth the decision boundary.
- **Advantages**:
    - Simplicity and intuition.
    - Effective on data with a clear class separation.
- **Disadvantages**:
    - High computational cost for large datasets.
    - Sensitive to data scaling.

Overview of KNN

- **Definition**: Supervised method used for classification and regression.
  - Computes the distance between the test point and all training points.
  - Selects the k nearest neighbors.
  - Classifies the test point based on the majority of neighbors (for classification) or the average of neighbors (for regression).
- **Common Distance Metrics**: Euclidean or Manhattan Distance
- **Choosing the value of k**:
  - Small k: more sensitive to noise.
  - Large k: may overly smooth the decision boundary.
- **Advantages**:
  - Simplicity and intuition.
  - Effective on data with a clear class separation.
- **Disadvantages**:
  - High computational cost for large datasets.
  - Sensitive to data scaling.

k-Nearest Neighbors (k-NN) Algorithm

- **Definition:** In statistics, the k-Nearest Neighbors (k-NN) algorithm is a non-parametric classification method.

- "Non-parametric" refers to statistical methods that do not assume a specific form for the data distribution.

- **History:** Developed by Evelyn Fix and Joseph Hodges in 1951, later expanded by Thomas Cover.

Additional Resources

- Stat quest - Video Lecture (click here)
- Distance Metrics (click here)
- Cosine Distance (click here)
- KNN Demo (click here)

Jaccard Metric

- **Definition:** Measures the distance between data that have the presence or absence of terms.
- **Formula:**

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Where A and B are two sets, $|A \cap B|$ is the cardinality of their intersection, and $|A \cup B|$ is the cardinality of their union.

Example of Jaccard Distance

Data Samples:

- Sample 1: $\{A, B, C, D, E\}$
- Sample 2: $\{B, D, E, F, G\}$

Steps:

- Intersection: $\{B, D, E\}$
- Union: $\{A, B, C, D, E, F, G\}$

Jaccard Distance Calculation

- Jaccard metric formula:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

- Substituting values:

$$J(\text{Sample 1}, \text{Sample 2}) = \frac{|\{B, D, E\}|}{|\{A, B, C, D, E, F, G\}|} = \frac{3}{7} = 0.43$$

Therefore, the Jaccard distance between Sample 1 and Sample 2 is 0.43. The closer the distance is to 1, the more similar the sets are.

Example of Jaccard Distance - Sample 3

Sample 3:

- Sample 3: {A, C, D, E, H}

Steps:

- Intersection with Sample 1: {A, C, D, E}
- Union with Sample 1: {A, B, C, D, E, H}

Jaccard Distance Calculation - Sample 3

- Jaccard metric formula:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

- Substituting values:

$$J(\text{Sample 1}, \text{Sample 3}) = \frac{|\{A, C, D, E\}|}{|\{A, B, C, D, E, H\}|} = \frac{4}{6} = 0.67$$

Therefore, the Jaccard distance between Sample 1 and Sample 3 is 0.67. The closer the distance is to 1, the more similar the sets are.

Examples of Supervised Problems

- Determine whether an email is spam or not (classification).
- Given a movie on Netflix, predict the rating a user will give to a particular movie (regression).
- Given an image, determine which objects are present in it (dog, cat, computer, buildings, etc.) (classification).

K-Nearest Neighbors (KNN) Algorithm

```
Algorithm KNN:
1. Receive the training dataset with n examples
2. Define the value of k (number of neighbors)
3. For each test point:
   a. Compute the distance from the test point to all points
   b. Select the k closest training points
   c. If classification:
      i. Return the most common class among k neighbors
   d. If regression:
      i. Return the mean value of k neighbors
4. End
```

K-Nearest Neighbors (KNN) Algorithm

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_iris
from sklearn.metrics import accuracy_score
data = load_iris() # Load the dataset
X = data.data
y = data.target
# Split data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)
knn = KNeighborsClassifier(n_neighbors=3) # Train
knn.fit(X_train, y_train)
y_pred = knn.predict(X_test) # Predict on test set
accuracy = accuracy_score(y_test, y_pred) # Compute accuracy
print(f'Accuracy: {accuracy:.2f}')
```