Nearest Neighbor Classification Quiz

Congratulations! You passed!

Grade received 100% | Latest Submission Grade 100% | To pass 60% or higher

Go to next item

Review Learning Objectives

Submit your assignment

Due Feb 4, 11:59 PM IST

Receive grade

To Pass 60% or higher

Try again

Your grade 100%

View Feedback

We keep your highest score

Like Dislike Report an issue

1. What is the primary goal of the Nearest Neighbors algorithm? **1 / 1 point**

- (●) To classify data points based on the class labels of their k-nearest neighbors.
- ( ) To find the average of the target variable for the k-nearest data points.
- ( ) To reduce the dimensionality of the dataset for faster computation.
- ( ) To perform clustering and group similar data points together.

Correct
Correct! The Nearest Neighbors algorithm classifies data points based on the class labels of their k-nearest neighbors.

2. What does "k" represent in k-Nearest Neighbors (KNN)? **1 / 1 point**

- ( ) The number of features in the dataset.
- (●) The number of nearest neighbors used to classify a data point.
- ( ) The number of clusters in the data.
- ( ) The dimensionality of the data.

Correct
Correct! "k" in KNN represents the number of nearest neighbors used to classify a data point.

3. How does the value of "k" in KNN affect the model's performance? **1 / 1 point**

- ( ) A larger value of "k" improves model accuracy by considering more neighbors.
- ( ) The value of "k" only affects the training time of the model.
- ( ) The value of "k" does not affect the model's performance.
- (●) A smaller value of "k" can lead to overfitting by considering noise in the data.

Correct
Correct! A smaller "k" in KNN can lead to overfitting, as the model might consider noise in the data.

4. In KNN, how are new data points classified? **1 / 1 point**

- ( ) By calculating the mean of the target variable for the k-nearest data points.
- (●) By assigning the class label that is most frequent among the k-nearest neighbors.
- ( ) By fitting a linear regression model to the k-nearest data points.
- ( ) By averaging the features of the k-nearest data points.

Correct
Correct! KNN classifies new data points based on the majority class among their k-nearest neighbors.

5. What is the primary difference between KNN and Radius Neighbors (RNN)? **1 / 1 point**

- ( ) RNN is a parametric classifier, whereas KNN is a non-parametric classifier.
- ( ) RNN is a supervised learning algorithm, whereas KNN is an unsupervised learning algorithm.
- (●) RNN uses a fixed radius to find neighbors, whereas KNN uses a variable number of neighbors.
- ( ) RNN is a distance-based classifier, whereas KNN is a tree-based classifier.

Correct
Correct! RNN uses a fixed radius to identify neighbors within a specified distance, while KNN uses a variable number of neighbors based on the value of "k."

6. What does the "R" represent in Radius Neighbors (RNN)? **1 / 1 point**

- ( ) The number of neighbors used for classification.
- (●) The maximum distance from a data point within which neighbors are considered.
- ( ) The regularization parameter used to control model complexity.
- ( ) The radius of the hypersphere used for clustering in the data.

Correct
Correct! "R" in RNN represents the maximum distance from a data point within which neighbors are considered for classification.

7. How can you create a KNN classifier using Scikit-learn in Python? **1 / 1 point**

- (●) By importing the "KNeighborsClassifier" class and calling the "fit" method with the training data.
- ( ) By importing the "KMeans" class and calling the "predict" method with the training data.
- ( ) By importing the "RandomForestClassifier" class and calling the "fit" method with the training data.
- ( ) By importing the "LogisticRegression" class and calling the "predict" method with the training data.

Correct
Correct! You can create a KNN classifier using Scikit-learn by importing the "KNeighborsClassifier" class and calling the "fit" method with the training data.

8. Which of the following statements about KNN and RNN is true? **1 / 1 point**

- ( ) KNN is more suitable for high-dimensional data, while RNN is more suitable for low-dimensional data.
- (●) KNN and RNN have similar computational complexity for finding neighbors.
- ( ) RNN is more robust to outliers compared to KNN.
- ( ) KNN performs better on imbalanced datasets, while RNN performs better on balanced datasets.

Correct
Correct! Both KNN and RNN have similar computational complexity for finding neighbors, but KNN varies in complexity depending on the value of "k."

9. In Scikit-learn, how can you set the value of "k" for a KNN classifier? **1 / 1 point**

- (●) By specifying the "n_neighbors" parameter when creating the KNeighborsClassifier object.
- ( ) By using the "set_k" method after training the KNN classifier.
- ( ) By calling the "k_neighbors" method with the desired value of "k."
- ( ) By setting the "k" attribute of the KNN classifier object directly.

Correct
Correct! You can set the value of "k" by specifying the "n_neighbors" parameter when creating the KNeighborsClassifier object.

10. What Scikit-learn method is used to set the value of "R" for an RNN classifier? **1 / 1 point**

- ( ) set_radius()
- (●) By specifying the "radius" parameter when creating the RadiusNeighborsClassifier object.
- ( ) By calling the "radius_neighbors" method with the desired value of "R."
- ( ) By setting the "R" attribute of the RNN classifier object directly.

Correct
Correct! You can set the value of "R" by specifying the "radius" parameter when creating the RadiusNeighborsClassifier object.