k-Nearest Neighbors

Paulo S. A. Sousa

2022-03-27

Context

- The k-nearest-neighbors algorithm can be used in both classification and regression problems.
- The method relies on finding "similar" records in the training data:
  - These "neighbors" are then used to get a prediction for the new record:
    - By voting (for classification) or averaging (for regression).

The k-NN Classifier

- The idea in k-nearest-neighbors methods is to identify k records in the training dataset that are similar to a new record that we wish to classify.
- We then use these similar (neighboring) records to classify the new record into a class, assigning the new record to the predominant class among these neighbors.
- Denote the values of the predictors for this new record by $x_1, x_2, \ldots, x_p$. We look for records in our training data that are similar or "near" (i.e., with values close to $x_1, x_2, \ldots, x_p$) the record to be classified.
- Then, based on the classes to which those proximate records belong, we assign a class to the record that we want to classify.

Determining Neighbors

- The k-nearest-neighbors algorithm does not make assumptions about the form of the relationship between the class membership (Y) and the predictors $X_1, X_2, \ldots, X_p$.
- Since it does not involve the estimation of parameters, this method is regarded as a *nonparametric* method.
- To measure the distance, we can use the popular *Euclidean distance*.
- Given two records $(x_1, x_2, \ldots, x_p)$ and $(u_1, u_2, \ldots, u_p)$, the Euclidean distance is given by

$$\sqrt{(x_1 - u_1)^2 + (x_2 - u_2)^2 + \cdots + (x_p - u_p)^2}.$$

- There is other distance metrics, which we will see later:
  - However, since this algorithm calculate distances numerous times, this Euclidean distance is very popular regarding k-nearest-neighbors algorithm.

Determining Neighbors

- To harmonize scales, in most cases, predictors should first be standardized before computing a Euclidean distance.
- This standardizing procedure must use *only* the training data, excluding validation data.

Classification Rule

The classification rule works as follows:

1. Find the nearest k neighbors to the record to be classified.
2. Use a majority decision rule to classify the record, where the record is classified as a member of the majority class of the k neighbors.

Choosing k

- If k is too low, we may be fitting to the noise in the data -- overfitting.
- However, if k is too high, we will miss out on the method's ability to capture the local structure in the data -- one of its main advantages.
- So there is a tradeoff on which we must weight up.
- A balanced choice greatly depends on the nature of the data:
  - The more complex and irregular the structure of the data, the lower the optimum value of k.
  - Typically, values of k fall in the range 1 to 20.
  - To avoid ties, we will use odd numbers.

Setting the Cutoff Value

- k-NN uses a majority decision rule to classify a new record.
- The definition of "majority" is directly linked to the notion of a *cutoff value* applied to the class membership probabilities.
- In the binary outcome case, for a new record, the proportion of class 1 members among its neighbors is an estimate of its propensity (probability) of belonging to class 1.
- Using a simple majority rule is equivalent to setting the cutoff value to 0.5.
- Changing the cutoff value affects the error rates and, therefore, we might want to choose a cutoff other than the default 0.5 for the purpose of maximizing accuracy or for incorporating misclassification costs.

k-NN with More Than Two Classes

- The "majority rule" means that a new record is classified as a member of the majority class of its k neighbors.
- An alternative, when there is a specific class that we are interested in identifying, is to calculate the proportion of the k neighbors that belong to this class of interest:
  - Use that as an estimate of the probability (propensity) that the new record belongs to that class.
  - And then refer to a user-specified cutoff value to decide whether to assign the new record to that class.

Converting Categorical Variables to Binary Dummies

- It usually does not make sense to calculate Euclidean distance between two non-numeric categories.
- Therefore, before k-NN can be applied, categorical variables must be converted to binary dummies.
- In contrast to the situation with statistical models such as regression, all m binaries should be created and used with k-NN.
  - While mathematically redundant, since $m - 1$ dummies contain the same information as m dummies, multicollinearity is not a problem.
- Moreover, in k-NN the use of $m - 1$ dummies can yield different classifications than the use of m dummies:
  - Imbalance in the contribution of the different categories to the model.

k-NN for a Numerical Outcome

- We can easily modify k-NN to predict a continuous value.
- The first step of determining neighbors by computing distances remains the same.
- In the second step, the majority rule is modified such that we take the average outcome value of the k-nearest neighbors to determine the prediction.
    - Often, this average is a weighted average, with the weight decreasing with increasing distance from the point at which the prediction is required.
- Another modification is in the error metric used for determining the optimal k.

Advantages and Disadvantages

- The main advantage of k-NN methods is their simplicity and lack of parametric assumptions.
- For large enough training sets, this algorithm performs surprisingly well,
  - especially when each class is characterized by multiple combinations of predictor values.

Advantages and Disadvantages

There are three difficulties with this algorithm:

1. The time to find the nearest neighbors in a large training set can be prohibitive. To mitigate this problem:
    - Apply dimension reduction techniques such as principal components analysis to reduce the dimension of the prediction problem.
    - Use sophisticated data structures such as search trees to speed up identification of the nearest neighbor.
        - This approach often settles for an “almost nearest” neighbor to improve speed.
        - An example is using bucketing, where the records are grouped into buckets so that records within each bucket are close to each other.

Advantages and Disadvantages

2. The number of records required in the training set to qualify as large increases exponentially with the number of predictors p.

- This is because the expected distance to the nearest neighbor goes up dramatically with p unless the size of the training set increases exponentially with p.
- This phenomenon is known as the *curse of dimensionality*, a fundamental issue pertinent to all classification, prediction, and clustering techniques.
- This is why we often seek to reduce the number of predictors through methods such as selecting subsets of the predictors for our model or by combining them using methods such as principal components analysis.

3. The time-consuming computation is done at the time of prediction. Thus, this behavior may prohibit using this algorithm for real-time prediction of a large number of records simultaneously.