Kernelized Weighted K Nearest Neighbors

Mateus Maia Marques[1], **Anderson Ara**[2]

Programa de Pós-Graduação em Matemática - UFBA

Instituto de Matemática e Estatística

[1]mateusmaia11@gmail.com ,[2]anderson.ara@ufba.br

1. Introdução

K Nearest Neighbor (KNN) is one of the simplest and most general statistical learning techniques (Guo *et al.*, 2003). It is able to perform predictions of new configurations using a k similarity criterion that includes those closest to the training model. In general, rank by considering the class that is most in the k nearest neighbors. The most common similarity criterion is Euclidean distance. Through the *Kernel Trick*, developed by (Cortes and Vapnik, 1995), it is possible to execute a larger dimensional space without having to explicitly, being possible explicitly from the internal product. Thus, considering this internal product of higher dimensional space configurations as a measure of similarity, or the Kernel weighted KNN, allows you to use this measure as a weighting factor to try to improve as predictions.

2. K Nearest Neighbors

As stated earlier, KNN is a non-parametric statistical learning method whose decision rule is given by

$$g(x) = \underset{i \in N}{\text{mode}}\, y_i = \underset{c \in Y}{\text{argmax}} \sum_{i \in N} \delta(c = y_i) \tag{1}$$

where N is the set k observations closest to **x**, and $\delta(c = y_i)$ is the Dirac Delta function that assumes a when $y_i = c$ and zero otherwise. There are several ways to define the "proximity" criterion between observations, with Euclidean distance being the most common in applying this method.

$$D(x_i, \mathbf{x_j}) = \sqrt{\mathbf{x_i} \cdot \mathbf{x_i} - 2\mathbf{x_i} \cdot \mathbf{x_j} + \mathbf{x_j} \cdot \mathbf{x_j}} \tag{2}$$

Another widely used approach, capable of improving the technique prediction, is the Weighted KNN, in which there is a w_i factor that gives more weight to the most similar observations. So the new decision rule is given by the equation

$$g(x) = \underset{c \in Y}{\text{argmax}} \sum_{i \in N} w_i \delta(c = y_i). \tag{3}$$

Therefore, for the KNN Kernelized Weighted method, we modify the method so that the w_i factor considers the similarity of observations in other dimensional spaces.

3. *Kernel Trick*

Defining $\phi : x \to \phi(x), x \in X, \phi \in F$. The $kerneltrick$ makes it possible, through the internal product of observations in a ϕ space, to implicitly perform nonlinear mapping to a large space. This internal product is then represented by the kernel functions $K(x_i, x_j)$.

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \tag{4}$$

There are several types of kernel functions, among them the functions presented below, which are used in this work.

- Gaussian:

$$K(x, y) = \frac{1}{\sqrt{2\pi}} e^{\frac{1}{2} - ||x-y||^2}$$

- Exponential:

$$K(x, y) = \frac{1}{\sqrt{2\pi}} e^{\frac{1}{2} - ||x-y||}$$

- Epanechnikov:

$$K(x, y) = \frac{3}{4}(1 - ||x - y||^2)$$

- Tricubic:

$$K(x, y) = \frac{70}{81}(1 - |||x - y|||^3)^3$$

- Logistic:

$$K(x, y) = \frac{1}{e^{||x-y||} + 2 + e^{||x-y||}}$$

4. Kernel Weighted KNN

The *Kernelized Weighted K Nearest Neighbors* method is to use Equation 3, where the term similarity / weight (w_i) will now be given by the kernel function (Equation 4). Thus, it is possible to assign a weight in the classification of observations using the other dimensional spaces. So the new decision rule for this technique is

$$g(x) = \underset{c \in Y}{\text{argmax}} \sum_{i \in N} K(x, x_i)\delta(c = y_i) \tag{5}$$

5. Results in *Beachmark datasets*

The KWKNN technique was used in three different databases: Ionosphere, Seeds, Wine.

- **Ionosphere:** consists of 354 observations, with 34 parameters whose purpose is to classify binary.
- **Seeds:** The Seeds database refers to the database of 210 seeds collected, in which 7 parameters were registered and whose purpose is to classify three different types of seeds.
- **Wines**: consists of 183 observations of different wines, described through 13 parameters whose objective is also to classify in three different types of wines.

For each database the classic KNN was applied, as well as the Kernelized Weighted K Nearest Neighbors. For KWKNN, all kernel functions presented here have been applied to compare the performance of each method through accuracy. The configuration for validation was through a Holdout repeated 80 times, with a split ratio of 80% - 20% training and testing. The final result is shown in Figures 1, 2 and 3.

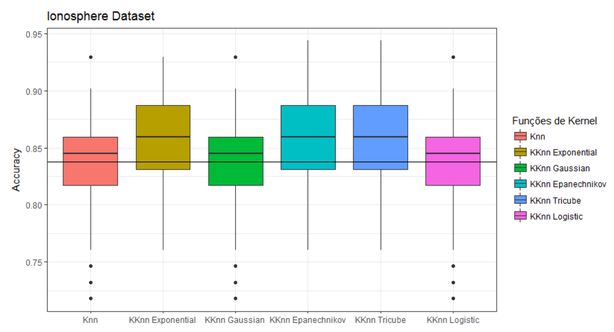

Figure 1: *Comparison between methods through database accuracy* ***Ionosphere.***

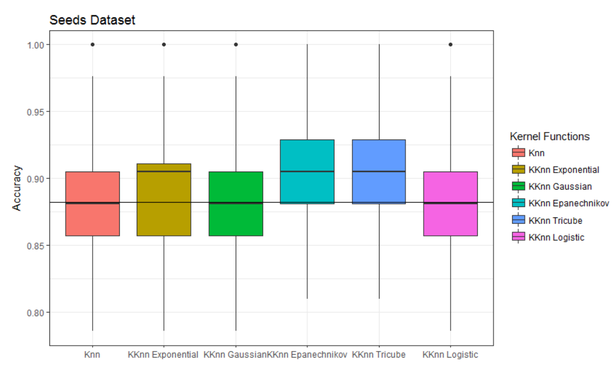

Figure 2: *Comparison between methods through database accuracy* ***Seeds.***

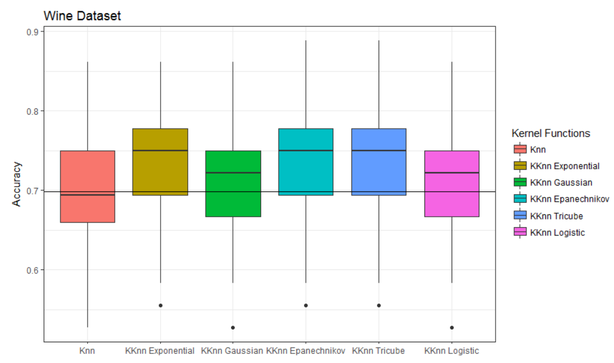

Figure 3: *Comparison between methods through database accuracy* ***Wines.***

It is easy to see that for all cases, Kernelized Weighted KNN, with Epanechnikov and Tricubic kernel functions, had better performance compared to the others.

6. Application in *Text Mining*

The area of text mining has been growing in recent decades, and generally consists of meeting certain standards, or extracting information from large volumes of text or documents. In this sense, the use of statistical machine learning techniques can assist in the automatic identification of certain classes or categories of thousands of documents. In this sense, traditional KNN has already been used as a tool for this purpose (Bijalwan *et al.*, 2014), and therefore, proposing an adaptation to this method may bring better results to what already exists in the literature.

In the context of Text Mining the main statistical modeling object is the *Bag of Words* which is a matrix whose lines represent each of the documents, and the columns represent each of the words, therefore, each element of this matrix indicates the frequency of each of the terms in each of the documents.

Due to the variety of texts to generate this matrix some text preprocessing steps are required. These steps vary for each occasion, but generally follow a framework with the following steps: **Stemming**, **Removal of *StopWords***, e **Formatação de. Texto**

In addition to processing, frequency weighting was done using *tf_idf* which is the product term of the word frequency and inverse frequency per document (Equation 6).

$$idf(W) = log \frac{\#(documents)}{\#\text{documents containing the word}(W)} \tag{6}$$

For the case in question, we used the database corresponding to 400 BBC News news, whose 200 belonged to the Sports category, while another 200 belonged to the Business category. After the preprocessing and term weighting steps, the data was divided in the proportion 80 % - 20 % in training and testing, respectively, and the validation was done through *Holdout* with 50 repetitions. The value of the number of neighbors was chosen using *tuning*, where $k = 3$ is the optimal value. The result is represented in Figure 4.

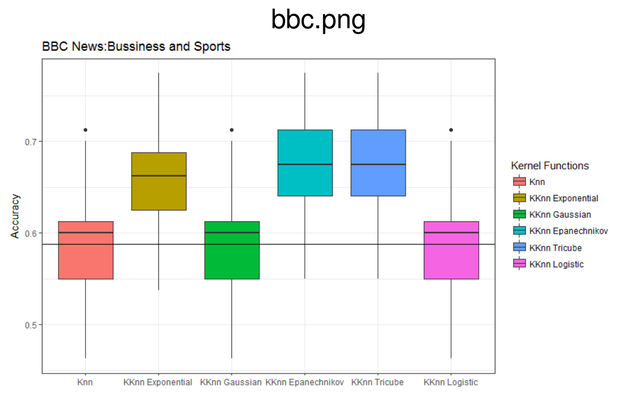

Figure 4: *Comparison between methods by database accuracy* ***BBC News.***

7. Final Comments

It can be seen that, in general, the Kernelized Weighted K Nearest Neighbors method showed a subtle improvement in the classification of the proposed databases. However, it is also necessary to analyze strategies to systematize the choice of the best kernel functions as well as their respective hyperparameters.

References

Bijalwan, V., Kumar, V., Kumari, P., and Pascual, J. (2014). Knn based machine learning approach for text and document mining. *International Journal of Database Theory and Application*, **7**(1), 61–70.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, **20**(3), 273–297.

Guo, G., Wang, H., Bell, D., Bi, Y., and Greer, K. (2003). Knn model-based approach in classification. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 986–996. Springer.