# Análisis del algoritmo de los K Vecinos más Cercanos usando diferentes hiper-parámetros

Jason Latouche
Escuela de Ingeniería en Computación
Instituto Tecnológico de Costa Rica
Cartago, Costa Rica
96jaslat@gmail.com

***Resumen*—Este documento analiza la exactitud del algoritmo de K Vecinos más Cercanos usando diferentes valores de K y diferentes distancias.**

## I. Introducción

Este documento analiza la exactitud del algoritmo de K Vecinos mas Cercanos[1], el cual es un método de clasificación supervisada, usando la combinación de diferentes hiper-parámetros, como el K y las distancias de Chebyshev, Manhattan y Levenshtein, para analizar el comportamiento.

## II. Metodología

Se pretende comparar la exactitud del algoritmo de K Vecinos mas Cercanos con los diferente hiper-parámetro realizando una serie de pruebas. Esto con el propósito de conocer como los hiper-parámetros alteran el resultado del algoritmo.

Para estas pruebas se usa NumPy, una librería desarrollada para Python de la cual se aprovecha las ventajas que ofrece como el rendimiento de operaciones sobre arreglos multi-dimensionales. El set de datos que se usa es CIFAR-10, el cual contiene 60 mil imágenes de 32x32 píxeles donde 50 mil son usadas para entrenamiento y 10 mil para pruebas.

### *II-A. Definición del algoritmo kNN [1]*

*II-A1. Algoritmo de entrenamiento:* Para cada ejemplo $< x, f(x) >$, donde $x \in X$ , agregar el ejemplo a la estructura representando el aprendizaje.

*II-A2. Algoritmo de clasificación:* Dado un ejemplar $x_q$ que debe ser clasificado, sean $x_1, ..., x_k$ los $k$ vecinos más cercanos a $x_q$ en los ejemplos de aprendizaje, regresar

$$\hat{f}(x) \leftarrow argmax_{v \in V} \sum_{i=1}^{k} \delta(v, f(x_i))$$

donde $\delta(a, b) = 1$ si $a = b$; y 0 en cualquier otro caso.

el valor $\hat{f}(x_q)$ devuelto por el algoritmo como un estimador de $f(x_q)$ es solo el valor más común de $f$ entre los $k$ vecinos más cercanos a $x_q$. Si elegimos $k = 1$; entonces el vecino más cercano a $x_i$ determina su valor.

### *II-B. Exactitud*

La exactitud se mide usando la siguiente fórmula:

$$\text{exactitud} = \frac{\text{categorías acertadas}}{\text{total de pruebas}} * 100$$

### *II-C. Hiper-parámetro K*

En el algoritmo de K Vecinos más Cercanos la K representa cuantos son vecinos más cercanos que se tomarán en cuenta para decidir la clasificación a la que pertenece la muestra. Este análisis se hará usando K = 1, K = 2 y K = 3

### *II-D. Hiper-parámetro de distancias*

Para calcular cuales son los vecinos más cercanos a una muestra es necesario aplicar algoritmos para medir la distancia entre los datos y la muestra.

*II-D1. Distancia Chebyshev:* La distancia de Chebyshev[2] está dada por la siguiente función:

$$D_{\text{Chebyshev}}(x, y) = max(|x_i - y_i|), i = 0, 1, ..., n$$

*II-D2. Distancia Manhattan:* La distancia de Manhattan[3] está dada por la siguiente función:

$$D_{\text{Manhattan}}(x, y) = \sum(|x_i - y_i|)$$

*II-D3. Distancia Levenshtein:* Dado dos hileras a, b, la distancia de Levenshtein[4] está dada por la función $D_{\text{Leven}}(|a|,|b|)$ que se define a continuación:

$$D_{\text{Leven}}(i, j) = \begin{cases} max(i, j) & \text{si min(i,j) = 0} \\ min(i, j) \begin{cases} D_{\text{Leven}}(i-1, j) + 1 \\ D_{\text{Leven}}(i, j-1) + 1 \\ D_{\text{Leven}}(i-1, j-1) + 1 \end{cases} & \text{en otro caso.} \end{cases}$$

## III. Experimentos

El experimento consiste en cargar los 50 mil datos con sus respectivos resultados y usar las 10 mil fotos para clasificarlas en base a los hiper-parámetros y conocer a cual categoría se aproxima más la muestra. Para lograr esto, 9 pruebas se van a ejecutar:

- Distancia Chebyshev con K = 1.
- Distancia Manhattan con K = 1.
- Distancia Levenshtein con K = 1.
- Distancia Chebyshev con K = 2.
- Distancia Manhattan con K = 2.
- Distancia Levenshtein con K = 2.
- Distancia Chebyshev con K = 3.

- Distancia Manhattan con K = 3.
- Distancia Levenshtein con K = 3.

## IV. Resultados

A continuación se muestra una tabla con los diferentes hiper-parámetros y su respectiva exactitud:

Cuadro I
TABLA DE EXACTITUD

| K | *Chebyshev* | *Manhattan* | *Levenshtein* |
|---|---|---|---|
| 1 | 9.2299 | 24.92 | 27.43 |
| 2 | 9.2299 | 24.92 | 27.43 |
| 3 | 9.06 | 23.39 | 24.34 |

## V. Conclusiones

- La similitud de valores en K = 1 y K = 2 puede radicar en que si en K = 2 hay un empate entre dos vecinos, se toma el más cercano a la muestra, haciendo la elección igual a la de K = 1.
- Aumentar la cantidad de vecinos que se toman como referencia para decidir a que clase pertenece, no necesariamente significa que se van a obtener mejores resultados.
- Para este experimento, el algoritmo de Levenshtein, con K = 1 o K = 2, presentó la mejor exactitud, con el 27.43 de la prueba categorizada con correctamente.

## Referencias


[1] En.wikipedia.org. (2018). k-nearest neighbors algorithm. [Online] Disponible en: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm [Accesado el 22 Feb. 2018].

[2] En.wikipedia.org. (2018). Chebyshev distance. [Online] Disponible en: https://en.wikipedia.org/wiki/Chebyshev_distance [Accesado el 22 Feb. 2018].

[3] En.wikipedia.org. (2018). Taxicab geometry. [Online] Disponible en: https://en.wikipedia.org/wiki/Taxicab_geometry [Accesado el 22 Feb. 2018].

[4] En.wikipedia.org. (2018). Levenshtein distance. [Online] Disponible en: https://en.wikipedia.org/wiki/Levenshtein_distance [Accesado el 22 Feb. 2018].