Apuntes 19 de marzo, 2024

*Semana 7, I Semestre

1st Gustavo Pérez Badilla
Escuela de Computación
Instituto Tecnológico de Costa Rica
Cartago, Costa Rica
gperezb2002@estudiantec.cr
2020084832

I. Preguntas y respuestas del 3er quiz

A. 1ra Pregunta: Describa la técnica y fórmula de Normalización y Estandarización. (35pts)

Normalización: Transforma los datos del Dataset a una escala con distribución normal facilitando el procesamiento de los datos, generalmente se convierten en valores en un intervalo de [0,1], dependiendo del problema estos pueden ajustarse a [-1,1].

$$x_{normalizado} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Estandarización: Transforma los datos en una escala con una media de 0 y una desviación estándar 1. Facilitando el entrenamiento y reduciendo el tiempo de procesamiento del Dataset.

$$Z = \frac{X - \mu}{\sigma}$$

Donde X corresponde al valor original, μ es la media y σ es la desviación estándar.

B. 2da Pregunta: Describa qué es overfitting y underfitting. (25pts)

Overfitting: Corresponde a la sobrealimentación de datos hacia el modelo, de forma que aprende demasiado de los estos al punto de que no es capaz de generalizar bien los nuevos datos.

Underfitting: Corresponde a la falta de aprendizaje del modelo debido a la carencia de datos, de forma que este no puede encontrar buenas relaciones entre las características otorgadas durante el entrenamiento.

C. 3ra Pregunta: Nombre y explique el proceso de convertir features contínuos a múltiples features binarios. (20pts)

Se denomina **Binning** y este busca un rango o intervalo entre los valores contínuos y los reduce a un solo valor, de forma que se permita la categorización de estos datos, en este caso corresponde a un valor binario.

D. 4ta Pregunta: Defina qué es un Epoch. (20pts)

Un **Epoch** corresponde a una iteración completa del modelo durante el proceso de entrenamiento, donde encuentra las relaciones entre los datos y las características que pueden ser reevaluadas posteriormente en otra iteración.

II. Anotaciones de la Tarea No.3

El objetivo de la tarea está en aplicar los diferentes filtros y compararlos para obtener los resultados de emplear cada uno, el caso del Chatbot (Natural Language Processing, NLP) corresponde a puntos adicionales, por lo que es importante darle prioridad a la implementación de los filtros.

Tomando en cuenta que las imagenes son matrices, los filtros corresponden a las modificaciones de los valores de dicha matriz, de forma que se obtiene la misma imagen con cambios uniformes. Para su implementación se puede emplear la biblioteca OpenCV.

Los filtros deben aplicarse a todas las imágenes para observar cuáles elementos se diferencian entre las clases así como cuales son visibles dependiendo del filtro aplicado. Se recomienda aplicar cada uno de los filtros en X y en Y independientemente para ver cada efecto, y posteriormente aplicar ambos para ver su comportamiento.

En cuanto a la sección del Chatbot que emplea NLP es importante entender el concepto de **Embedding**, el cual corresponde a una representación vectorial que mapea los textos, de forma que estos se representen como números y el modelo sea capaz de interpretarlos. Para emplear este procesamiento se debe buscar un modelo que emplee este concepto.

Este modelo permite realizar cálculos entre los valores de los vectores creados por embeddings, donde los resultados de estos pueden ser muy cercanos a otra característica, con esto se puede establecer una relación y por ende, una conclusión.

III. Feature Scaling

Corresponde a una técnica empleada en el aprendizaje de modelos que busca transformar los Features de un conjunto de datos (Dataset) en una misma escala uniforme. Para lograr esta escalabilidad, se emplean dos técnicas conocidas como **Estandarización** y **Normalización**.

Normalización: Consiste en la transformación de las variables a una escala común, mejorando la comparabilidad de los datos y acelera la convergencia de los modelos a los que se les otorga el conjunto de datos. Existen varios tipos de normalización dentro de las cuales destaca la normalización por Min-Max, donde escala las variables en un intrevalo de [0,1] principalmente.

Estandarización: Corresponde a un tipo de normalización que busca acomodar los valores de las características en una distribución normal, con media 0 y desviación estándar 1, se puede decir que sigue una distribución normal, sin embargo, cabe recalcar que este no acota los valores en el sentido de que no establece un límite mínimo o máximo, de forma que los valores atípicos se alejan más de la media.

Valor	Feature
10	A
20	B
30	C
40	D
50	E
60	F
70	G
80	H
90	I
100	J

Tabla I: Datos sin estandarizar

Valor estandarizado	Feature
-0.877	A
-0.577	B
-0.277	C
0.023	D
0.323	E
0.623	F
0.923	G
1.223	H
1.523	I
1.823	J

Tabla II: Datos estandarizados

La escalabilidad de los datos tiene sus ventajas, dentro de las cuales resaltan las siguientes:

- Es requerido por algoritmos que requieren cálculos de gradiente, ya que permite una convergencia más rápida, una mejor precisión al encontrar el mínimo global.
- Empleada principalmente en regresión logística y redes neuronales, en el caso de los algoritmos donde los calculos se basan en la distancia entre los valores (KNN, Árboles de decisión, entre otros) no es necesario realizarlo.
- Omitirlo propicia a que los datos se encuentren en diferentes escalas, por lo que pueden tener diferentes tamaños para los steps de entrenamiento.

En cuanto a cuál emplear depende de cuál se ajusta mejor al modelo, algunos consejos pueden ser:

- Si no se quiere asumir la distribución, lo preferible es emplear normalización.
- La estandarización es mejor emplearla cuando los datos siguen una distribución Gausiana, o si existen outliers (valores atípicos).

IV. Redes Neuronales

Corresponden a un modelo de Machine Learning que se basa en el cerebro humano y buscan imitar su comportamiento, recreando los vínculos entre las neuronas. Está constituida por capas donde cada una cuenta con una cantidad deteminada de elementos (neuronas), y cada una de estas contiene un peso y un umbral, que determinan si debe de enviarse una u otra señal a las siguientes capas de neuronas. Cuenta con una capa inicial que cuenta con los valores de entrada, posteriormente siguen una cantidad establecida de capas ocultas las cuales se encargan de encontrar las relaciones entre los valores ingresados y enviarlas a las siguientes, por último se encuentra una capa de salida de datos la cual retorna los resultados obtenidos del algoritmo.

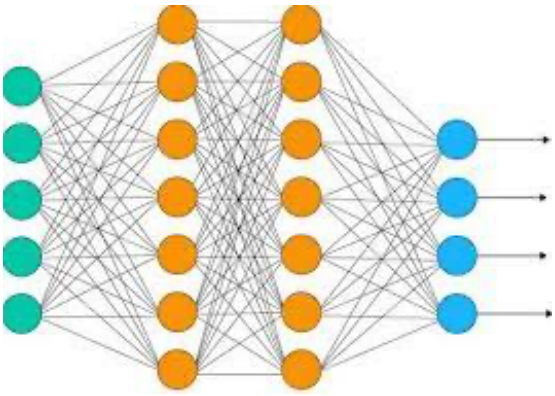

Fig. 1: Capas de una red neuronal.

Dentro del procesamiento de los datos en una red neuronal existe la capacidad de retroalimentar las neuronas de capas anteriores enviando los errores para corregir los resultados, a este proceso se le conoce como **Back Propagation**, en la Fig. 2 se pueden apreciar dos flechas en los extremos superior e inferior que muestran una vinculación entre una capa posterior a otra anterior, haciendo posible la visualización del concepto.

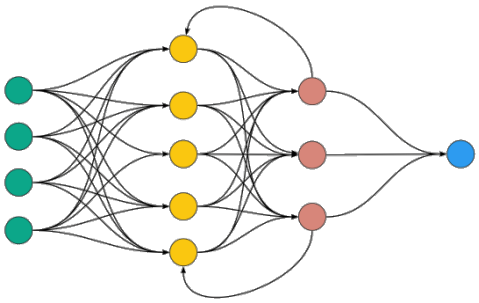

Fig. 2: Representación del Back Propagation.

A. *MNIST Dataset*

Consiste en un subconjunto de imágenes con dígitos escritos a mano pertenecientes a una base de datos mayor conocida como NIST. Cuenta con 70000 de los cuales 60000 apuntan a ser empleados para el entrenamiento del modelo, mientras que los 10000 sobrantes se usan para las pruebas. Fue creado para emplearlos en los primero algoritmos de inteligencia artificial visual.

Dichas imágenes tienen una dimensión de 28x28 píxeles y están en escala de grises con el propósito de que el algoritmo sea capaz de detectar patrones entre los dígitos para determinar a cuál corresponde realmente cada uno. En la Fig. 3 podemos apreciar un ejemplar encontrado dentro del Dataset.

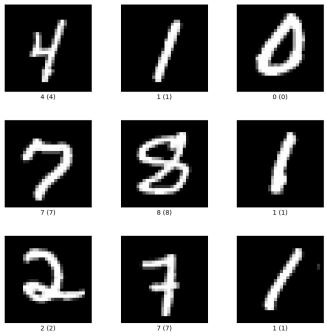

Fig. 3: Ejemplares presentes en el MNIST Dataset.

Lo mas cercano que se tiene para determinar a qué dígito corresponde una imágen es identificar los patrones presentes en cada número para poder compararlos y no los valores específicamente. Para este caso se requiere hacer una clasificación multiclase, de forma que el algoritmo sea capaz de otorgar una respuesta de entre 10 posibles, aquí es donde se introduce el **One-Hot vector** como un vector que cuenta con una cantidad de elementos igual a la cantidad de categorias establecidas, donde todos estos tienen un valor de 0 a excepción de solo un elemento el cual tendrá un valor de 1.

Color	Rojo	Azul	Verde
Vector One-Hot	[1, 0, 0]	[0, 1, 0]	[0, 0, 1]

Tabla III: Ejemplo de vector One-Hot para el color de una camiseta

En este caso cada neurona se encarga de cada uno de los dígitos, si más de una resultandar un valor de 1, se escoge aquella que tenga mayor probabilidad de ocurrencia. Entre mayor sea la cantidad de capas mejor, ya que las primeras se encargan de recopilar los detalles más generales de cada dígito, y las capas más profundas son las que determinan los detalles más específicos.

Para optimizar el proceso podemos cambiar los vectores por matrices, con esto logramos pasar de realizar N cálcules en uno solo, donde N corresponde al tamaño de la siguiente capa. De esta forma realizamos la regresión lineal en la matriz resultante

REFERENCES

[1] "Importance of feature scaling," scikit, https://scikit-learn.org/stable/auto_examples/preprocessing/plot_scaling_importance.html (accessed Mar. 30, 2024).

[2] "¿Qué son las redes neuronales?," IBM, https://www.ibm.com/es-es/topics/neural-networks (accessed Mar. 30, 2024).

[3] Y. LeCun, C. Cortes, and C. J. C. Burges, "The mnist database," MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges, http://yann.lecun.com/exdb/mnist/ (accessed Mar. 30, 2024).