**Trabajo Práctico**

**Matemática III**

**Curso:**

2° Cuatrimestre 2024 - Turno Noche

Profesores:

BOMPENSIERI Josefina

PRUDENTE Tomás

Alumnos:

GARCIA Nahuel (nahuelgar567@gmail.com)

MANQUEZ Víctor ([vmanquez37@gmail.com](mailto:vmanquez37@gmail.com))

Fecha de Entrega:

01/11/2024

**Informe**

**Parte 1: Análisis de la Base de Datos**

**1. Describe cada columna del conjunto de datos:**

**¿Qué representa?**

1. **id**(int): Identificador único del paciente.
2. **diagnosis**(string): Diagnóstico del cáncer, donde M representa maligno y B benigno.
3. **radius\_mean** (float): Promedio del radio de las células tumorales.
4. **texture\_mean** (float): Promedio de la textura de las células tumorales.
5. **perimeter\_mean** (float): Promedio del perímetro de las células tumorales.
6. **area\_mean** (float): Promedio del área ocupada por las células tumorales.
7. **smoothness\_mean** (float): Promedio de la suavidad de la superficie de las células tumorales.
8. **compactness\_mean** (float): Promedio de la compacidad de las células tumorales.
9. **concavity\_mean** (float): Promedio de la concavidad de las células tumorales.
10. **concave points\_mean** (float): Promedio de los puntos cóncavos en las células tumorales.
11. **symmetry\_mean** (float): Promedio de la simetría de las células tumorales.
12. **fractal\_dimension\_mean** (float): Promedio de la dimensión fractal de las células tumorales.
13. **radius\_se** (float): Error estándar del radio de las células tumorales.
14. **texture\_se** (float): Error estándar de la textura de las células tumorales.
15. **perimeter\_se** (float): Error estándar del perímetro de las células tumorales.
16. **area\_se** (float): Error estándar del área ocupada por las células tumorales.
17. **smoothness\_se** (float): Error estándar de la suavidad de las células tumorales.
18. **compactness\_se** (float): Error estándar de la compacidad de las células tumorales.
19. **concavity\_se** (float): Error estándar de la concavidad de las células tumorales.
20. **concave points\_se** (float): Error estándar de los puntos cóncavos en las células tumorales.
21. **symmetry\_se** (float): Error estándar de la simetría de las células tumorales.
22. **fractal\_dimension\_se** (float): Error estándar de la dimensión fractal de las células tumorales.
23. **radius\_worst** (float): Peor radio registrado de las células tumorales.
24. **texture\_worst** (float): Peor textura registrada de las células tumorales.
25. **perimeter\_worst** (float): Peor perímetro registrado de las células tumorales.
26. **area\_worst** (float): Peor área ocupada por las células tumorales.
27. **smoothness\_worst** (float): Peor suavidad registrada de las células tumorales.
28. **compactness\_worst** (float): Peor compacidad registrada de las células tumorales.
29. **concavity\_worst** (float): Peor concavidad registrada de las células tumorales.
30. **concave points\_worst** (float): Peor cantidad de puntos cóncavos en las células tumorales.
31. **symmetry\_worst** (float): Peor simetría registrada de las células tumorales.
32. **fractal\_dimension\_worst** (float): Peor dimensión fractal de las células tumorales.

**¿Es una variable categórica, continua, discreta, etc.?**

* **Categórica:** diagnosis.
* **Continuas:** Todas las demás variables (medias, errores estándar y peores medidas).

**2. Análisis de Correlaciones:**

**Usa correlaciones para evaluar si las características son relevantes para la tarea de clasificación. Identifica las relaciones entre las características y la columna objetivo. ¿Cuáles parecen ser las más influyentes?**

Al analizar la matriz de correlación, se observa que las variables relacionadas con las características celulares, como el radio promedio y el perímetro promedio, presentan una fuerte correlación positiva con la variable de diagnóstico (diagnosis). Por otro lado, variables como el área de la célula y la suavidad muestran una correlación inversa significativa.

Cabe aclarar que en este análisis no se tuvo en cuenta la columna ID, ya que se considera irrelevante para la evaluación de las características predictivas.

**3. Análisis de Factibilidad:**

**¿Es esta base de datos adecuada para entrenar una red neuronal de clasificación? Justifica tu respuesta en base a los datos.**

Sí, esta base de datos es adecuada para una red neuronal de clasificación. Contiene suficiente cantidad de datos y características relevantes que están bien correlacionadas con la variable objetivo de diagnóstico, lo que ayuda a la red a aprender patrones útiles para clasificar entre casos benignos y malignos.

**Explica el propósito de entrenar la red. ¿Qué intentará predecir? ¿Cuál es el objetivo del modelo?**

El propósito de entrenar esta red neuronal es predecir si un tumor es benigno o maligno, con base a características celulares obtenidas. El modelo tiene como objetivo identificar patrones en los datos que permitan clasificar los tumores con precisión.

**4. Datos Atípicos y Limpieza de Datos:**

**¿Es necesario limpiarlos? Si decides hacerlo, justifica tu decisión y describe cómo lo harías. Si no, explica por qué preferís dejarlos.**

Sí, es necesario. Al eliminar o ajustar los datos atípicos ayuda a mejorar el funcionamiento de la red neuronal. Al reducir la influencia de valores extremos, se evita que estos desvíen el aprendizaje del modelo y provoquen errores o predicciones menos precisas.

Después de limpiar la base de datos, generé un nuevo archivo llamado cancer\_mama\_limpio.csv. Luego, analicé la correlación de las variables y seleccioné las 10 características con mayor correlación respecto a la variable de salida. Con estas variables, creé un mapa de correlación y guardé los datos resultantes en un archivo adicional llamado cancer\_mama\_limpio\_top10.csv. Esta selección optimiza la entrada para el modelo de red neuronal al centrarse en las características más relevantes para la clasificación.

**5. Transformaciones Preliminares:**

**Explica por qué estas transformaciones son necesarias**

Las transformaciones preliminares, como la *normalización* y la *conversión de categorías a valores numéricos*, son esenciales para mejorar el rendimiento de la red neuronal. *Normalizar* asegura que todas las características tengan una escala similar, evitando que valores más grandes dominen el aprendizaje. Convertir *categorías a valores numéricos* permite que la red procese los datos categóricos, ya que las redes neuronales solo trabajan con valores numéricos.

**Parte 2: Desarrollo de la Red Neuronal**

**Arquitectura de la Red: Dibuja la arquitectura de la red neuronal.**

**¿Cuántas capas tendrá la red?**

La red tendrá dos capas: una capa ocultas y una capa de salida

**¿Cuántas neuronas habrá en cada capa?**

* **Capa de entrada*:*** 9 neuronas
* **Capa oculta**: 3 neuronas
* **Capa de salida**: 1 neurona

**¿Qué función de activación utilizará cada capa? Explica por qué elegiste esas funciones de activación.**

Las dos capas ocultas utilizaran la función ReLu y la capa de salida sigmoide.

X si X > 0 0

ReLu: deja igual los valores de entrada positivos y = 0 y a los negativos

0 si X < 0

**Sigmoide**: transforma la salida en un valor entre 0 y 1, ideal para clasificación binaria.

**4. Análisis de Overfitting:**

**Analiza si tu modelo presenta overfitting.**

**¿Cómo se puede identificar el punto en el que el modelo empieza a sobreajustarse a los datos de entrenamiento? ¿Cómo lo manejarías?**

El gráfico muestra una ligera brecha entre la precisión de entrenamiento y la de prueba que indica un grado de sobreajuste, aproximadamente a partir de las 2.000 iteraciones. Lo manejaría deteniendo el entrenamiento alrededor de las 10,000-15,000 iteraciones o simplificar el modelo podrían ser soluciones efectivas para mejorar la generalización.

**Parte 3: Comparación con scikit-learn**

**2. Comparación de Rendimiento: Compara el rendimiento de la red neuronal implementada en numpy con la que implementaste en scikit-learn.**

**¿En qué aspectos los resultados son similares?**

La red neuronal manual y la de scikit-learn mostraron un rendimiento muy similar, con una precisión del 95-96% en ambos conjuntos de Test y Train.

**¿En qué aspectos son diferentes? Compara las curvas de entrenamiento y validación, el tiempo de ejecución, y cualquier otro aspecto relevante.**

**Parte 4: Conclusión Final**

Desarrollar una red neuronal desde cero nos permitió entender en profundidad cómo funcionan las redes neuronales en cada paso del proceso, desde la inicialización de pesos y sesgos hasta el forward y el backward. Al construir manualmente cada función de activación y sus derivadas, adquirí una comprensión mucho más detallada de cómo los gradientes afectan el ajuste de los pesos y de cómo se optimizan los modelos de aprendizaje.

Comparado con el uso la librería de scikit-learn, crear una red neuronal manualmente ofrece ventajas significativas en términos de aprendizaje. Trabajar desde cero ayuda a entender mejor las matemáticas y la lógica detrás del entrenamiento de los modelos. La principal desventaja de este enfoque es que resulta mucho más laborioso y es muy propenso a errores, ya que requiere gestionar manualmente cada cálculo y ajuste de parámetros.