INSITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE COMPUTO

ING. INTELIGENCIA ARTIFICIAL

APRENDIZAJE DE MÁQUINA

HERNÁNDEZ JIMÉNEZ ERICK YAEL

PATIÑO FLORES SAMUEL

ROBERT GARAYZAR ARTURO

LABORATORIO 5

GRUPO 5BV1

21 DE OCTUBRE DE 2024

Introducción

En esta práctica se implementaron funciones sin el uso de bibliotecas externas para calcular medidas de desempeño de modelos de clasificación binaria, tales como: Accuracy, Error, Precisión, Recall, Tasa de Verdaderos Positivos (TPR), Tasa de Verdaderos Negativos (TNR), Tasa de Falsos Positivos (FPR), Tasa de Falsos Negativos (FNR) y el F1-Score. Adicionalmente, se realizó una comparación utilizando una biblioteca de Python para validar los resultados obtenidos.

Parte I: Implementación Manual de las Métricas

1. Accuracy

La precisión, o Accuracy, es la medida de cuántas predicciones fueron correctas en relación con el total de predicciones realizadas. Se define como la suma de verdaderos positivos (TP) y verdaderos negativos (TN) dividida por el total de observaciones:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Donde:

- **TP**: Verdaderos positivos (predicciones correctas de la clase positiva)
- **TN**: Verdaderos negativos (predicciones correctas de la clase negativa)
- **FP**: Falsos positivos (predicciones incorrectas de la clase positiva)
- **FN**: Falsos negativos (predicciones incorrectas de la clase negativa)

La función implementada calcula esta métrica a partir de los valores mencionados, proporcionando una visión global del desempeño del modelo.

2. Error

El Error mide el porcentaje de predicciones incorrectas. Es el complemento de la precisión, definido como:

$$\text{Error} = 1 - \text{Accuracy}$$

Esta métrica es útil para entender la tasa de fallos de un modelo y puede ser crítica en aplicaciones donde los errores tienen un alto costo.

3. Matriz de Confusión

La matriz de confusión es una tabla que describe el desempeño de un modelo de clasificación, mostrando el número de predicciones correctas e incorrectas clasificadas por cada clase. Se divide en:

- **TP (True Positives)**: Predicciones correctas de la clase positiva.
- **TN (True Negatives)**: Predicciones correctas de la clase negativa.
- **FP (False Positives)**: Predicciones incorrectas de la clase positiva (falsos positivos).
- **FN (False Negatives)**: Predicciones incorrectas de la clase negativa (falsos negativos).

4. Precision (Precisión)

La precisión evalúa la proporción de predicciones positivas que realmente son correctas. Se calcula como:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Es particularmente útil en contextos donde los falsos positivos tienen un alto costo, como en la detección de fraudes o enfermedades.

5. Recall (Sensibilidad o Tasa de Verdaderos Positivos)

La sensibilidad o recall mide la proporción de verdaderos positivos correctamente identificados. Se define como:

$$\text{Recall} = \frac{TP}{TP + FN}$$

Es una métrica importante cuando es crítico identificar todos los casos positivos, como en la detección de enfermedades graves.

6. True Positive Rate (Tasa de Verdaderos Positivos)

Este valor es equivalente al Recall, y mide la capacidad del modelo para identificar correctamente los positivos.

7. True Negative Rate (Tasa de Verdaderos Negativos)

La Tasa de Verdaderos Negativos mide la proporción de predicciones negativas correctas:

$$\text{True Negative Rate} = \frac{TN}{TN + FP}$$

Esta métrica es útil en escenarios donde también es importante que el modelo clasifique correctamente los negativos.

8. False Positive Rate (Tasa de Falsos Positivos)

Esta tasa mide la proporción de predicciones incorrectamente clasificadas como positivas entre todas las negativas. Un modelo con una alta tasa de falsos positivos podría ser problemático en aplicaciones como la detección de fraudes.

9. False Negative Rate (Tasa de Falsos Negativos)

De manera similar, la tasa de falsos negativos mide la proporción de positivos incorrectamente clasificados como negativos.

10. F1-Score

El F1-Score es una métrica que combina la precisión y el recall en un solo valor, especialmente útil cuando existe un desequilibrio entre clases:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Es una métrica adecuada cuando se quiere equilibrar la importancia de falsos positivos y falsos negativos.

Parte II: Uso de Bibliotecas de Python

Las métricas mencionadas anteriormente están disponibles en bibliotecas de Python, como scikit-learn. Esta biblioteca proporciona funciones predefinidas que permiten calcular fácilmente el rendimiento de un modelo con solo importar y aplicar las funciones correspondientes.

Por ejemplo, las funciones accuracy_score, precision_score, recall_score, y f1_score permiten calcular estas métricas a partir de las etiquetas verdaderas y las predicciones realizadas por el modelo. Además, la función confusion_matrix genera automáticamente la matriz de confusión.

En esta práctica se utilizó scikit-learn para procesar un conjunto de predicciones y etiquetas verdaderas, lo que produjo una matriz de confusión y las siguientes métricas:

- **Matriz de confusión**: Una tabla que mostró la distribución de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos.
- **Accuracy**: 0.6667, lo que indica que el modelo predijo correctamente el 66.67% de los ejemplos.
- **Precision**: 0.5, lo que significa que el 50% de las predicciones positivas fueron correctas.
- **Recall**: 1.0, indicando que todos los verdaderos positivos fueron correctamente identificados.
- **F1-Score**: 0.6667, que es un balance entre la precisión y la sensibilidad del modelo.

Conclusiones

Erick Yael Hernández Jiménez:

Los parámetros de evaluación como Accuracy, Precision, Recall, y F1-Score son esenciales para comprender el rendimiento de un modelo de clasificación, ya que proporcionan una visión detallada de su capacidad para hacer predicciones correctas, evitar errores y balancear correctamente los positivos y negativos en diferentes contextos. Utilizar librerías como scikit-learn para calcular estas métricas no solo simplifica el proceso, sino que también mejora la escalabilidad al permitir el manejo eficiente de datasets más complejos y de mayor tamaño. Estas herramientas están optimizadas para trabajar con grandes volúmenes de datos, lo que facilita la evaluación de modelos en tiempo real o en sistemas de producción, permitiendo a los analistas identificar rápidamente patrones, ajustar modelos y tomar decisiones informadas sobre la efectividad de sus algoritmos de manera más rápida y precisa.

Samuel Patiño Flores:

La programación de estas métricas nos ayuda a comprender cómo las diferentes métricas pueden priorizar aspectos distintos del rendimiento. Por ejemplo, en problemas de clasificación donde los falsos negativos pueden ser más críticos que los falsos positivos, el uso de métricas como el recall puede ser más adecuado que simplemente depender del accuracy. El cálculo manual también resalta la importancia de elegir la métrica correcta para cada caso de uso particular.

El uso de bibliotecas externas no solo ahorra tiempo, sino que también ayuda a garantizar la precisión en los cálculos, dado que estas funciones han sido ampliamente probadas en la comunidad de machine learning. A modo de ejemplo, un código en Python utilizando scikit-learn puede calcular la matriz de confusión y sus métricas asociadas en unas pocas líneas, lo que resulta altamente productivo para proyectos donde la eficiencia y la rapidez en la evaluación son importantes.

Arturo Robert Garayzar:

Esta práctica permitió desarrollar una comprensión profunda de las métricas de rendimiento en aprendizaje automático, primero implementándolas desde cero y luego utilizando una biblioteca estándar de Python. El uso de métricas como la precisión, el recall y el F1-Score es fundamental para evaluar el desempeño de modelos de clasificación, especialmente en casos donde la clasificación errónea

puede tener un impacto significativo. Las funciones implementadas manualmente se alinearon con los resultados obtenidos mediante scikit-learn, confirmando la validez del enfoque.