Aprendizaje Supervisado
Clasificación

Profesor: Juan Gamarra Moreno

Algunos Modelos de Clasificación

Regresión Logística

Concepto General:

- La **regresión logística** es un modelo de clasificación utilizado principalmente para resolver problemas de clasificación binaria (dos clases). Es una extensión de la regresión lineal, pero en lugar de predecir un valor continuo, predice una probabilidad que luego se puede convertir en una clase.

Regresión Logística

Función Sigmoide:

- La regresión logística usa una **función sigmoide** para transformar la salida lineal en una probabilidad. La fórmula de la sigmoide es:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

- Donde z es la salida del modelo lineal $z = w^T x + b$. Esta transformación asegura que la salida esté en el rango de [0, 1], lo cual es ideal para problemas de clasificación binaria.

Regresión Logística

Decisión de Clasificación:

- Una vez que se obtiene la probabilidad, se puede clasificar utilizando un umbral (generalmente 0.5). Si la probabilidad es mayor o igual a 0.5, la clase es 1, de lo contrario, es 0.

Ventajas y Uso Común:

- La regresión logística es simple y efectiva para problemas donde las clases son linealmente separables y donde se requiere interpretabilidad de las probabilidades.

K-Nearest Neighbors (KNN)

Concepto General:

- **KNN** es un **algoritmo de clasificación basado en instancias**, es decir, almacena todos los ejemplos de entrenamiento y para clasificar un nuevo punto, simplemente encuentra los **K** puntos más cercanos en los datos de entrenamiento y asigna la clase más común (mayoría) entre estos vecinos.

K-Nearest Neighbors (KNN)

Distancia y Vecinos:

- El algoritmo se basa en una métrica de distancia, típicamente la distancia euclidiana. Para un nuevo punto, se calcula la distancia a todos los puntos del conjunto de entrenamiento y se seleccionan los **K vecinos más cercanos**.

Valor de K:

- La elección del valor de **K** es crucial. Si **K** es muy pequeño, el modelo puede ser muy sensible al ruido, y si **K** es muy grande, puede perder detalles locales. Se recomienda ajustar **K** utilizando validación cruzada.

K-Nearest Neighbors (KNN)

Ventajas y Limitaciones:

- **Ventajas**: Es un algoritmo muy intuitivo y no requiere suposiciones fuertes sobre la distribución de los datos.
- **Limitaciones**: Puede volverse ineficiente para conjuntos de datos grandes, ya que requiere comparar con cada punto en el conjunto de entrenamiento. Además, puede tener dificultades en datos con dimensiones muy altas

Support Vector Machines (SVM)

Concepto General:

- **SVM** es un modelo supervisado de clasificación que intenta encontrar el **hiperplano** que mejor separa las diferentes clases. En problemas de dos clases, el SVM busca el **margen máximo**, es decir, el hiperplano que está más lejos de los puntos de ambas clases, de modo que la separación sea lo más clara posible.

Support Vector Machines (SVM)

Margen Máximo y Soportes:

- Los puntos de datos que están más cerca del hiperplano se llaman **vectores de soporte**. Son los puntos más críticos porque definen la posición y orientación del hiperplano. El objetivo de SVM es maximizar el margen, que es la distancia entre los vectores de soporte y el hiperplano.

Support Vector Machines (SVM)

Kernels para Clasificación no Lineal:

- Si las clases no son linealmente separables en su espacio original, SVM puede usar **kernels** para transformar los datos a un espacio de mayor dimensión donde la separación sea posible. El **kernel de base radial (RBF)** y el **polinomial** son los más comunes para datos no lineales.

Support Vector Machines (SVM)

Ventajas y Uso:

- **Ventajas**: SVM es muy efectivo en espacios de alta dimensión y sigue siendo robusto frente al sobreajuste si se ajustan bien los parámetros. Funciona muy bien con márgenes claros entre las clases.
- **Limitaciones**: En casos donde los datos no están claramente separados o cuando hay mucho ruido, el ajuste de los hiperparámetros y el uso de kernels puede ser complicado y computacionalmente costoso.

Métricas de Evaluación en Clasificación

Precision (Precisión)

La **precisión** mide la proporción de ejemplos clasificados como positivos que son verdaderamente positivos. Indica cuán precisa es la clasificación positiva del modelo.

- **Fórmula**:

$$Precision = \frac{Verdaderos\ Positivos\ (TP)}{Verdaderos\ Positivos\ (TP) + Falsos\ Positivos\ (FP)}$$

Precision (Precisión)

Interpretación:

- Alta precisión significa que el modelo tiene pocos **falsos positivos**, lo que es importante en situaciones donde las predicciones incorrectas como positivas tienen un alto costo (por ejemplo, clasificar a una persona como enferma cuando no lo está).

Ejemplo:

- Si el modelo clasifica 100 ejemplos como positivos, y de esos, 90 son realmente positivos, la precisión sería 90%.

Recall (Sensibilidad o Tasa de Verdaderos Positivos)

Definición: El **recall** mide la proporción de ejemplos verdaderamente positivos que el modelo fue capaz de identificar. Indica la capacidad del modelo para encontrar todos los ejemplos positivos en el conjunto de datos.

- **Fórmula**:

$$Recall = \frac{Verdaderos\ Positivos\ (TP)}{Verdaderos\ Positivos\ (TP) + Falsos\ Negativos\ (FP)}$$

Recall (Sensibilidad o Tasa de Verdaderos Positivos)

Interpretación:

- Alto recall significa que el modelo tiene pocos **falsos negativos**, lo cual es crucial en casos donde es vital identificar todos los positivos, como en la detección de cáncer, donde es preferible detectar todos los posibles casos.

Ejemplo:

- Si de 100 personas que tienen una enfermedad, el modelo identifica correctamente a 90, el recall es del 90%.

F1-Score

Definición: El **F1-Score** es la **media armónica** entre la precisión y la sensibilidad. Es especialmente útil en casos donde hay un **desequilibrio** en las clases, es decir, cuando una clase es mucho más común que la otra.

- **Fórmula**:

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

F1-Score

Interpretación:

- El **F1-Score** es útil cuando se necesita un equilibrio entre precisión y recall, y se desea evitar que una métrica sea significativamente mayor que la otra.

Ejemplo: Si un modelo tiene una precisión de 0.80 y un recall de 0.60, el F1-Score será:

$$F1 = 2 \times \frac{0.80 \times 0.60}{0.80 + 0.60} = 0.6857$$

Accuracy (Exactitud)

Definición: El **accuracy** mide la proporción total de predicciones correctas, tanto para las clases positivas como para las negativas. Es la métrica más intuitiva y es útil cuando las clases están **balanceadas**.

- **Fórmula**:

$$Recall = \frac{Verdaderos\ Positivos\ (TP) + Verdaderos\ Negativos\ (TP)}{Total\ de\ Ejemplos}$$

Accuracy (Exactitud)

Interpretación:

- Una alta **accuracy** significa que el modelo realiza un buen trabajo clasificando correctamente la mayoría de los ejemplos, tanto positivos como negativos.
- Sin embargo, en **conjuntos de datos desbalanceados** (donde una clase domina sobre la otra), la **accuracy** puede ser engañosa. Por ejemplo, en un conjunto de datos donde el 95% de los ejemplos son negativos, un modelo que siempre predice la clase negativa tendrá una alta exactitud (95%), pero no será un buen clasificador.

Accuracy (Exactitud)

Ejemplo:

- Si un modelo clasifica correctamente 90 de 100 ejemplos, su accuracy es 90%. Sin embargo, si en un conjunto de datos desbalanceado (95% de ejemplos negativos), un modelo predice todo como negativo, puede tener un 95% de accuracy pero será ineficaz para identificar los positivos.

Relación y Uso de estas Métricas

- **Precisión vs. Recall**: Estas métricas tienden a tener una relación inversa en muchos casos. Incrementar la precisión puede disminuir el recall y viceversa. Por ello, es importante elegir la métrica adecuada según el problema. Por ejemplo, si es más costoso predecir un falso positivo que un falso negativo, la precisión es más importante. Si el costo de perder un positivo es más alto (como en la detección de enfermedades), el recall será más relevante.
- **F1-Score en Casos Desbalanceados**: En escenarios donde una clase domina sobre la otra (por ejemplo, en la detección de fraudes donde solo el 1% de las transacciones son fraudulentas), el **F1-Score** es la métrica preferida, ya que proporciona un equilibrio entre precisión y recall.

Resumen de Cuándo Usar Cada Métrica

- **Precisión**: Útil cuando los **falsos positivos** son costosos.
- **Recall**: Importante cuando los **falsos negativos** son inaceptables.
- **F1-Score**: Preferido cuando las clases están desbalanceadas y se necesita un equilibrio entre precisión y recall.
- **Accuracy**: Adecuada en **clases balanceadas**, pero puede ser engañosa en conjuntos de datos desbalanceados.

Curva ROC y AUC (Area Under the Curve)

Definición: La **Curva ROC (Receiver Operating Characteristic)** es una representación gráfica de la relación entre el **Recall (Tasa de Verdaderos Positivos)** y la **Tasa de Falsos Positivos (FPR)** a diferentes umbrales de clasificación.

Tasa de Falsos Positivos (FPR):

$$FPR = \frac{Falsos\ Positivos\ (FP)}{Falsos\ Positivos\ (FP) + Verdaderos\ Negativos\ (TN)}$$

Curva ROC y AUC (Area Under the Curve)

Interpretación de la Curva ROC:

- En el gráfico ROC, el eje **Y** representa el **Recall (TPR)** y el eje **X** representa la **Tasa de Falsos Positivos (FPR)**. Cada punto en la curva corresponde a un umbral de decisión diferente.
- Un modelo perfecto tendría una curva ROC que pasa por la esquina superior izquierda (donde el **TPR** es 1 y el **FPR** es 0), mientras que un modelo aleatorio sigue la línea diagonal (45°) del gráfico.

Curva ROC y AUC (Area Under the Curve)

Uso:

- La Curva ROC y el AUC son útiles en problemas de clasificación donde el **desbalance de clases** es significativo. Ayuda a visualizar cómo el rendimiento del modelo cambia con diferentes umbrales y proporciona una evaluación más completa del rendimiento del modelo, en lugar de depender de una única métrica como el accuracy.

Curva ROC y AUC (Area Under the Curve)

Relación entre las Métricas y la Curva ROC:

- **Precisión y Recall**: Aunque la precisión y el recall son útiles, dependen de un umbral específico para clasificar los ejemplos como positivos o negativos. La curva ROC, en cambio, evalúa el rendimiento del modelo en **diferentes umbrales**.
- **F1-Score y ROC**: Si se necesita un equilibrio entre precisión y recall, el F1-Score es útil, pero la curva ROC proporciona una visión más amplia del rendimiento general del modelo.
- **Accuracy y ROC**: La accuracy puede ser engañosa en datos desbalanceados, mientras que la curva ROC refleja mejor el rendimiento en estos casos.

Curva ROC y AUC (Area Under the Curve)

Ejemplo de la Curva ROC:

- Supongamos que un modelo de clasificación binaria tiene diferentes umbrales para predecir si una transacción es fraudulenta o no. A medida que ajustamos el umbral, obtenemos diferentes tasas de verdaderos positivos y falsos positivos. La curva ROC nos muestra cómo varían estas tasas a lo largo de los umbrales, permitiendo seleccionar el umbral más adecuado según nuestras necesidades.