Logotipo, nombre de la empresa

Descripción generada automáticamente

Universidad de Montevideo – Introducción a la Ciencia de Datos

19 de Noviembre, 2024

**1. Como interpretaría los coeficientes que resultan de entrenar un modelo de Regresión Logística? Interprete para el caso de un coeficiente negativo, positivo y cero. Cómo es que estos afectan la probabilidad del resultado?**

Los coeficientes de un modelo de Regresión Logística representan el efecto de las variables independientes (predictoras) sobre la probabilidad del resultado en la variable dependiente, la cual es categórica y binaria (por ejemplo, éxito o fracaso). Los coeficientes afectan la probabilidad del resultado a través de su influencia en el logit (log-odds), que es la función lineal subyacente al modelo.

La relación de los coeficientes y la probabilidad se define por la función logística:

Interpretación de los coeficientes:

A) Coeficiente positivo:

* Indica que a medida que aumenta el valor de la variable independiente , también aumenta el logit (log-odds) de que ocurra el evento positivo (resultado = 1). Cuando alfa es muy positivo, tiende a cero, por lo que la fracción tiende a 1.
* Esto implica que la probabilidad del evento aumenta.

B) Coeficiente negativo:

* Indica que a medida que aumenta el valor de la variable independiente , disminuye el logit (log-odds) de que ocurra el evento positivo. Cuando alfa es muy negativo, tiende a infinito, por lo que la fracción tiende a 0.
* Esto implica que la probabilidad del evento disminuye.

C) Coeficiente igual a cero:

* Indica que la variable independiente no tiene efecto sobre el logit ni sobre las odds.
* La probabilidad del resultado no cambia cuando esta variable varía.

**2. Considere los dos siguientes escenarios:**

* 1. **Un científico de datos entrena una regresión lineal para capturar una relación parabólica.**
  2. **Un científico de datos entrena un algoritmo Random Forest con una gran profundidad y un gran número de árboles.**

**Cual escenario es probable que resulte en underfitting y cual en overfitting? Explique su respuesta.**

* **Escenario 1:** Este escenario es un caso clásico de **underfitting**. La regresión lineal solo puede modelar relaciones lineales entre las variables independientes y la variable dependiente. Si la verdadera relación es parabólica (no lineal), la regresión lineal no podrá ajustarse adecuadamente a los datos, independientemente de los parámetros o la cantidad de datos disponibles. Como resultado, el modelo tendrá alta grado de error (es decir, grandes diferencias entre las predicciones del modelo y los valores reales) tanto en los datos de entrenamiento como en los de prueba. Esto se traduce en un modelo que no captura la complejidad inherente de los datos.
* **Escenario 2:** Este escenario es un caso típico de **overfitting**. Los Random Forests tienden a ser robustos frente al overfitting cuando tienen un número limitado de árboles y una profundidad razonable. Sin embargo, cuando se configuran con gran profundidad, los árboles individuales pueden volverse muy específicos para los datos de entrenamiento (capturando incluso el ruido). Si además el modelo tiene un gran número de árboles, el conjunto agregado puede ajustarse excesivamente a las particularidades de los datos de entrenamiento. Esto resulta en un modelo que funciona muy bien en los datos de entrenamiento pero tiene un desempeño pobre en los datos de prueba debido a su incapacidad para generalizar.

**3. Considere el siguiente corpus, compuesto por 3 textos:**

# List of sentences

sentences = [

"The cat sat on the mat.",

"The dog barked at the cat.",

"The dog sat on the mat."

]

**Calcule la representación TF-IDF para las palabras “dog” y “cat”.**

Para calcular el TF-IDF para las palabras “cat” y “dog” debemos calcular primero el term frequency y luego el inverse document frequency.

Paso 1: TF

A screenshot of a table

Description automatically generated

Para calcular el IDF usamos la formula vista en clase:

A screenshot of a table

Description automatically generated

**4. Calcule la correlación de Spearman para la siguiente tabla:**

Ver el ejemplo de la presentación de pruebas estadísticas. Al calcular los rangos (ranks) de las columnas vemos que los rangos coinciden en todas las filas. Por lo tanto los cuadrados de las diferencias serán 0 y la correlación dará perfecta, es decir igual a 1. Formula:

**5. Comente los errores que pueden apreciarse en los gráficos del anexo y como modificaría los mismos.**

Todas las graficas presentan algún error en el eje-y. Veamos a continuación:

Como vimos en clase, la primer grafica presenta el típico error cuando se quiere mostrar “lo bajo es bueno” (*How Can I Show That Down Is Good?*). El error consiste en representar de manera decreciente en lugar de modificar el eje-y de manera inversa y mostrar un aumento creciente.

La segunda grafica presente el claro error de mostrar iconos que no son proporcionales. La diferencia de altura debe calcularse respecto al cero y no respecto al valor mínimo del eje-y. El eje debe representarse respecto al cero absoluto.

Nuevamente, las diferencias no son proporcionales a las alturas. No se provee ningún tipo de eje o escala. Visto de la manera que se presenta, el grafico parece sesgado hacia ciertos meses cuando en realidad la distribución es uniforme. La realidad es que la diferencia de nacimientos entre los distintos meses no es significativa, pero al mostrarlo de la manera que se muestra sin el eje-y parece que hay meses donde hay significativamente mas nacimientos.

**6. Una compañía de marketing desea enviar promociones a sus clientes en base sus interacciones en su plataforma de streaming. Su cliente le ofrece un dataset con los patrones de visualizaciones de sus clientes. Además, le informa que la empresa por su conocimiento del rubro estima segmentos de clientes entre 5 y 20 grupos. ¿A qué tipo de problema cree que se está enfrentando? ¿Qué algoritmo utilizaría para resolver este problema y como determinaría la cantidad optima de segmentos?**

Este problema es un problema no supervisado de clustering donde queremos encontrar segmentos de clientes para enviar promociones personalizadas a cada segmento. La letra aporta que por conocimiento del rubro se estima que la cantidad segmentos ronda entre 5 y 20. Como sabemos de antemano la estimación de la cantidad de segmentos podemos ajustar un modelo de Clustering como K-Means (u otros) variando la cantidad de clusters. Podemos elegir una *metrica* para medir la calidad de los clusters resultantes como el coeficiente de Silhouette (u otras) y tomar la combinación de hiperparametros que maximice/minimice la metrica en cuestión bajo la restricción que la cantidad de segmentos este en el rango estimado. Otro mecanismo para determinar el numero optimo de segmentos puede ser mediante el “método del codo” (elbow method).

**7. Como se descompone una serie de tiempo? Que modelos utilizaría para capturar las distintas componentes?**

En clase vimos que una serie de tiempo se descompone en 4 componentes (modelo aditivo):

Respectivamente, componente tendencial, estacional, cíclica y error irreductible.

Como vimos en clase, la componente tendencial se captura con un modelo lineal como la Regresion Lineal. La componente estacional y cíclica se suelen detectar con modelos armónicos o series de Fourier y el error irreductible mediante modelos como el ruido blanco o zero mean model.

**8. Suponga que usted quiere utilizar un algoritmo no supervisado para encontrar clusters en su conjunto de datos. Sin embargo, sus datos son categóricos (no numéricos). Que feature engineering debe aplicarse a estas columnas categóricas y por qué?**

Cuando se trabaja con datos categóricos y se quiere aplicar un algoritmo de clustering no supervisado, es necesario realizar una transformación adecuada de las columnas categóricas en valores numéricos para que los algoritmos de clustering puedan procesarlas. Esto se debe a que la mayoría de los algoritmos de clustering, como K-means y DBSCAN, requieren datos numéricos para calcular distancias entre los puntos de datos y asignarlos a clusters. Los procesamientos de columnas categóricas dependen de la naturaleza misma de la columna. Si la columna es categorica nominal se debe aplicar un One Hot Encoding ya que no existe un orden subyacente. Si la columna es categorica ordinal debe aplicarse un Ordinal Encoding ya que existe un orden subyacente. No se debe aplicar Label Encoding en caso de que exista un orden ya que este no captura el orden de los datos. Y si la columna es nominal y se aplica Label Encoding se introduce un orden inexistente, por ejemplo, podría asignar: Azul = 0, Rojo = 2 y Verde = 3, y esto no significa que Verde > Rojo > Azul.

**9. En qué casos puede ser una mala decisión imputar valores faltantes utilizando la media y en cuales la moda?**

Imputar valores faltantes con la media o la moda es una técnica común de imputación, pero no siempre es la más adecuada. El uso de estas técnicas depende del tipo de variable y de la distribución de los datos.

**Media:** Si los datos tienen una distribución sesgada (por ejemplo, si hay valores extremos o outliers), la media no representará bien el valor central de los datos. Imputar con la media en este caso puede introducir sesgo y distorsionar la estructura de los datos. Asimismo, si la columna tiene una alta variabilidad o dispersión (gran diferencia entre los valores), la media puede no reflejar adecuadamente la tendencia central.

**Moda:** Si los datos son relativamente equilibrados (distribución uniforme, es decir, no hay una categoría que sea mucho más frecuente que las demás), imputar con la moda puede introducir un sesgo, ya que el valor más frecuente podría no ser representativo de las observaciones faltantes. Al igual que en la imputación con la media, si la distribución de las categorías tiene varios picos o modas (multimodal), imputar con la moda podría hacer que los datos faltantes sean reemplazados por una categoría que no refleja la diversidad de las observaciones originales.

**10. Que ventajas presenta realizar una validación K-fold (con 5 folds) vs una validación donde se realiza un train/test Split de 80/20?**

**Mejor estimación de la generalización:** K-fold proporciona una estimación más robusta de la capacidad del modelo para generalizar, ya que el modelo se entrena y evalúa múltiples veces en diferentes particiones del conjunto de datos. En cada iteración, se utiliza una partición diferente para la evaluación, lo que permite que el modelo sea evaluado en diferentes subconjuntos de datos. En el caso del train/test split 80/20, el modelo solo se entrena en un subconjunto específico (el 80%) y se evalúa en el otro subconjunto (el 20%). Esto puede ser muy sensible a la elección de los datos de entrenamiento y prueba, lo que puede dar una evaluación menos confiable, especialmente si el conjunto de datos es pequeño o si los datos son sesgados de alguna manera.

**Reducción de la varianza en la estimación del rendimiento:** Con K-fold, se realiza un promedio de las métricas de rendimiento de cada uno de los 5 folds. Esto ayuda a mitigar la varianza que puede surgir al depender de un único subconjunto de prueba, como ocurre en el train/test split 80/20. Con un solo split, el rendimiento puede ser afectado por el azar del conjunto de prueba seleccionado. K-fold, al hacer uso de todos los datos en diferentes momentos para pruebas y entrenamientos, da una evaluación más equilibrada y confiable del rendimiento del modelo.

Otras razones:

* **Mayor robustez** en la evaluación del modelo, al promediar múltiples particiones.
* **Menor varianza** en las métricas de rendimiento, lo que da una mejor idea del rendimiento real del modelo.
* **Mejor uso de los datos**, especialmente cuando se tienen pocos datos.
* **Más adecuado para modelos con alta varianza**, que requieren una evaluación más precisa.
* **Mejor para la selección de hiperparámetros** y la optimización de modelos.
* **Mayor capacidad para detectar sobreajuste y subajuste** a través de las distintas particiones.

A graph with red lines and numbers

Description automatically generated



