

Clasificación de ingreso anual mayor a 50k

José Pablo Martínez Valdivia

7 de septiembre de 2024

Abstract

This paper presents a predictive model based on the Random Forest algorithm to classify income levels using the Census Income (50k) dataset. The objective of the study is to predict whether an individual's income exceeds \$50,000 per year based on demographic and work-related attributes. Furthermore, I discuss the various preprocessing decisions made, such as handling encoding categorical variables, and the choice of hyperparameters to optimize model performance.

1. Introducción

El dataset "Census Income" fue proveido por el "UCI Machine Learning Repository". Contiene 14 variables y 48842 instancias de datos provenientes de un censo realizado en 1994 para determinar si un adulto genera un ingreso anual mayor a \$50k. Emplearemos el modelo de bosque aleatorio (random forest) para emplear una predicción categórica y disminuir el overfit en el modelo.

2. ETL

2.1. Extracción

Los datos fueron descargados por medio de la librería de "ucimlrepo". Estos proveen datos demográficos como edad, estado civil, educación, ocupación, raza, sexo, país de origen, etc.

Los datos se encuentran completos en su mayoría, pero hay datos faltantes en las columnas de tipo de trabajo, ocupación y país de origen como se puede ver en 1. Se tomó la decisión de deshacerse de las filas con datos faltantes las cuales representan 1221 instancias, lo cual es el 2.4 % de los datos; sobrándonos 47621 instancias. Esta decisión se tomó considerando que algún tipo de imputación podría introducir errores para la tarea de clasificación.

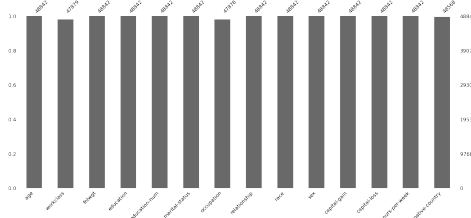


Figura 1: Valores faltantes.

2.2. Transformación

Para poder general el bosque aleatorio necesitamos convertir las columnas categóricas en valores numéricos. Para esto haremos uso de "one-hot encoding", esta técnica consta de tomar todas las categorías en una columna y crear una columna por categoría, dejando un uno en la categoría que presenta la instancia y ceros en el resto.

Colour		Green	Red	Blue
Green		0	1	1
Red		1	1	1
Blue		1	0	1
		0	0	0
		0	1	0

Figura 2: Método de one-hot encoding.

Solo la columna de *Estado* contiene variables categóricas, pero se decidió añadir al modelo, ya que se espera que diferentes estados tengan influencia en la tasa de crímenes. Esto se hizo por medio de codificación buscando las siglas únicas de cada estado y asignándoles un número dependiendo de su orden (de 0 a n estados).

Las variables numéricas se encontraban en distintos rangos de valores por lo cual se implementó un escalador usando la técnica de *z-scaling* donde se toma la media y desviación estándar de la distribución y se usan para escalar las variables transformando la distribución a una con media en 0 y desviación estándar de 1.

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

3. Regresión

3.1. Modelo lineal

Para este modelo tomamos la suposición de una relación lineal entre las variables más un sesgo.

$$Y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p \quad (2)$$

Lo cual podemos representar de forma matricial para poder implementar las predicciones de todos los datos como:

$$\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1p} \\ 1 & x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (3)$$

3.2. Función de costo

Se usó la función de *Mean square Errors (MSE)* la cual representa el promedio de las divergencias de la predicción y la variable esperada al cuadrado

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y})^2 \quad (4)$$

Usamos esta función para encontrar el error en los pesos y poder ajustar respectivamente en un proceso iterativo.

3.3. Descenso Gradiente

Evaluando nuestra función de costo llámese $J(\theta)$, con respecto a cada uno de los pesos θ y multiplicando por un valor de aprendizaje α podemos ajustar cada uno de los pesos de la siguiente manera.

$$\theta := \theta - \alpha \frac{\partial}{\partial \theta} J(\theta) \quad (5)$$

Realizando esto en un proceso iterativo, podemos llegar a un mínimo local de la ecuación y disminuir el error.

3.4. Coeficiente de Determinación

La R^2 es una métrica que mide la variabilidad de un factor con respecto a otro. Este lo obtenemos tomando 1 menos el cociente de la suma de las divergencias de la variable Y con sus predicciones \hat{Y} al cuadrado, sobre la suma de sus diferencias con la media \bar{Y} al cuadrado.

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} \quad (6)$$

Este valor entre más cercano a uno nos indica que hay mayor probabilidad de que la predicción explique a la variable objetivo. Esta métrica será usada más adelante para evaluar el modelo.

3.5. Entrenamiento

Las filas del dataset fueron aleatorizadas y separadas en tres conjuntos; entrenamiento, validación y prueba, estos constan del 60 %, 20 % y 20 % de los datos respectivamente. Esta separación se hizo para garantizar la eficiencia del modelo ante datos nuevos y castigar el sobre ajuste que pueda tener al trabajar solo con los datos provistos.

Posteriormente se corrió la regresión lineal con el descenso gradiente hasta que no hubiera cambio en los coeficientes o un número fijo de epochs se cumpliera. Cada epoch se calculo el *MSE* tomando la pérdida con el conjunto de validación y ajustando los coeficientes en magnitud del valor de aprendizaje.

4. Resultados

Después de 40,000 epochs, el modelo converge a un *MSE* de 0.042171 en el conjunto de validación y finalmente podemos ver las R^2 para cada conjunto en la tabla 1 comparando con el conjunto de prueba; esto representa un alto ajuste y poder predictivo del modelo con los datos reales de asesinatos en Estados Unidos. Podemos notar un grado bajo de overfitting para los datos de entrenamiento (1 %) al momento de comparar las R^2 de entrenamiento y validación el cual en esta magnitud no presenta un problema considerable para predecir datos nuevos. Igualmente podemos ver poca diferencia en la reducción del error 3 para los conjuntos de validación y entrenamiento. En general el modelo tiene un buen poder predictivo y no cae en overfitting por lo cual podemos considerar su grado de sesgo como bajo.

R^2 train	0.9554
R^2 validation	0.9335
R^2 test	0.9462

Cuadro 1: Desglose de R^2 .

La figura 3 muestra la gráfica del error de la función de pérdida a través de las epochs. Podemos ver que el modelo converge relativamente rápido y las 40000 están de más para el ajuste.

Finalmente podemos observar en la figura 4 como se comparan las predicciones con los valores reales del conjunto de prueba. Se nota inicialmente

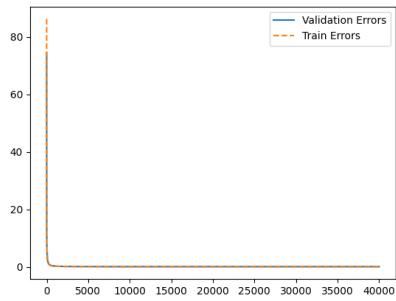


Figura 3: Función de pérdida.

que la predicción se mantiene consistente para valores cercanos a 0 pero comienza a divergir con valores de asesinatos más altos.

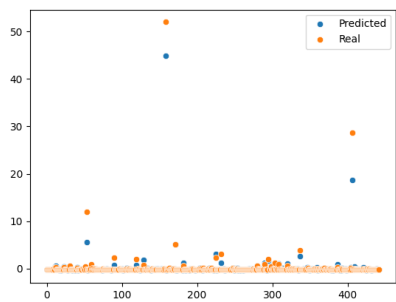


Figura 4: Resultado de la regresión.

Referencias

Redmond, Michael. (2011). Communities and Crime Unnormalized. UCI Machine Learning Repository. <https://doi.org/10.24432/C5PC8X>.