CAMBIO DE TRABAJO DE LOS CIENTIFICOS DE DATOS

Ronald Luis Tapia Flores

Universidad Mayor de "San Andrés"

La Paz, Bolivia

rltapia3@umsa.bo

Resumen – El estudio del Dataframe para determinar que candidatos están interesados en cambiar sus empleos actuales para ser parte de una empresa activa en Big Data y Data Science mediante el uso de algoritmos de inteligencia artificial. Uso de Clasificadores Tree y Logistic Regression

Palabras Clave – Clasificacion por Arboles, Regresión Logistica.Dataframe, PCA

Abstract - The Dataframe study to determine which candidates are interested in changing their current jobs to be part of a company active in Big Data and Data Science through the use of artificial intelligence algorithms. Using Tree Classifiers and Logistic Regression

***Keywords* - Tree Classifiers, Logistic Regression, Dataframe. PCA**

I. INTRODUCCIÓN

Una empresa activa en Big Data y Data Science quiere contratar científicos de datos entre las personas que superan con éxito algunos cursos que imparte la empresa. Mucha gente se apunta a su formación. La empresa quiere saber cuáles de estos candidatos realmente quieren trabajar para la empresa después de formarse o buscar un nuevo empleo porque ayuda a reducir el costo y el tiempo, así como la calidad de la formación o la planificación de los cursos y la categorización de los candidatos. La información relacionada con la demografía, la educación y la experiencia está en manos de la inscripción y la inscripción de los candidatos. Este conjunto de datos está diseñado para comprender los factores que llevan a una persona a dejar su trabajo actual para dedicarse también a investigaciones de recursos humanos. Por modelo (s) que utiliza las credenciales actuales, datos demográficos y datos de experiencia, podrá predecir la probabilidad de que un candidato busque un nuevo trabajo o trabaje para la empresa, además de interpretar los factores afectados en la decisión del empleado.

Para el correspondiente estudio se utilizará el lenguaje de programación Python pues es el más versátil en la aplicación del estudio de Dataframes con Inteligencia Artificial

II. OBJETIVO

Predecir la probabilidad de que un candidato trabaje para la empresa

III. VARIABLES

enrollee_id: ID único del candidato.

city: Código de ciudad.

city_ development _index: Índice de desarrollo de la ciudad (escalado).

gender: Género del candidato.

relevent_experience: Experiencia relevante del candidato.

enrolled_university: Tipo de curso universitario inscrito, si lo hubiera.

education_level: Nivel de educación del candidato.

major_discipline: Educación disciplina principal del candidato.

experience: Experiencia total del candidato en años.

company_size: Número de empleados en la empresa del empleador actual.

company_type: Tipo de compañía actual.

lastnewjob: Diferencia en años entre el trabajo anterior y el trabajo actual.

training_hours: Horas de formación completadas.

target: 0 - No busca un cambio de trabajo, 1 - Busca un cambio de trabajo.

IV. . PROCEDIMIENTO

A. IMPORTAR LOS DATOS

El estudio se inicializará con la importación de las principales librerías de Python como es el caso de numpy para el manejo numérico, pandas para el manejo de dataframes y matplotlib que facilita el uso de gráficos estadísticos.

Ya importadas las librerías se lee el dataframe denominado "aug_train.csv" en una variable que se llamara "data".

Como ya contamos con los datos del dataframe proseguimos con el siguiente paso que es analizarlo

B. ANALISIS DEL DATAFRAME

Se comenzará conociendo el número de datos y variables que tiene el "data" para lo cual se utilizará el comando "data.shape".

Comprobamos que los datos se visualizan correctamente con el comando "data.head()".

Posteriormente obtenemos información del "data" para visualizar si existen columnas nulas, los tipos de datos que almacenan las columnas, si existe equilibrio en el numero de datos de las columnas, y otros.

Notamos que en nuestro ejemplo el número de datos esta desequilibrado por lo que debemos solucionar este percance. Una de las técnicas es obtener el porcentaje de los datos faltantes para poder complementarlos con el dato que este de moda en la columna.

Siguiendo con el análisis, se verificará el número de valores únicos que tiene cada columna. En nuestro ejemplo notamos que la columna

"enrolee_id" tiene un valor muy alto y este altera la tendencia y también no es un dato muy significativo para nuestro estudio.

C. PREPROCESAMIENTO

Iniciamos eliminando el dato que altera la tendencia. Para ello utilizamos el comando data.drop("enrollee_id", axis=1, inplace=True).

Continuamos analizando las columnas y notamos que la columna city contiene datos mixtos entre caracteres y números por lo que para tratarlo creamos un diccionario de datos el cual almacenará el valor correspondiente al porcentaje de ocurrencia de las distintas ciudades dentro el dataframe. De esta forma al igualar la columna "city" con el diccionario de datos "d{}" transformamos los datos en numéricos.

En una observación dentro la etapa de análisis se notó que existía muchos valores faltantes por lo que reemplazamos los valores faltates o "NaN" por la moda de los valores que son validos dentro las diferentes columnas. Para ello utilizamos un ciclo como sigue:

```
for col in data.columns:
    data[col].fillna(data[col].mode()[0], inplace=True)
```

Posteriormente se comprueba si aun existes valores faltantes en las columnas con el comando data.isna().sum()

Una vez ya listos los datos ya podemos utilizar algoritmos de preprocesamiento de la librería sklearn.

El primero que se utilizará será el método "LabelEncoder" el cual codificará los datos no numéricos y los convertirá en numéricos

Se puede comprobar el data con el comando "data.head()" en cual nos mostrará que ya todos los datos son numéricos.

Posteriormente se dividirá el data en un arreglo "x" que contendrá las variables independientes y el arreglo "y" que almacenará el target.

Continuando con el preprocesamiento utilizando la librería sklearn se utilizará el método SimpleImputer que cambiará los valores NaN del arreglo "x" por los valores de las medias de cada columna.

Se procederá con la división de los datos para entrenamiento y para testeo para lo cual se utilizará el método "train_test_split" y las variables "xtrain, xtest, ytrain, ytest".

Ya obtenidas las variables de testeo y entrenamiento se estandarizarán estas para que tengan valores entre 0 y 1 que son los valores que generalmente trabajan los clasificadores tree y LogisticRegression. Para ello se utilizará el método StandardScaler de la librería sklearn.

D. SELECCIÓN DEL CLASIFICADOR

Para abordar de forma adecuada el dataframe se seleccionó clasificadores supervisados ya que se cuenta con el target el cual indica los resultados a los que el clasificador debe intentar llegar. Para este caso se pueden utilizar diferentes clasificadores supervisados se detallarán el clasificador Tree y el clasificador Logistic Regression.

1) CLASIFICATION TREE

El método del árbol de clasificación para respuestas binarias proporciona un árbol gráfico que describe el proceso de decisión que conduce a las predicciones de respuesta. Este método formaliza un proceso de decisión que utiliza un conjunto secuencial de preguntas sobre los valores de x para inducir una creación de los valores que producen cada predicción de clasificación para y. En comparación con la regresión logística y el análisis discriminante lineal, este método de clasificación es mucho menos restrictivo en la forma del límite de decisión del predictor. El árbol de clasificación resume las divisiones binarias de las variables en varias etapas para determinar la predicción. El conjunto de valores de x para los que $\hat{y} = 1$ consta de un conjunto de regiones rectangulares que se pueden resumir fácilmente sin un predictor lineal u otra fórmula. [1]

El método para construir un árbol de clasificación binaria utiliza un algoritmo de partición recursivo para determinar (1) cómo elegir la variable de división en cada nodo, (2) cómo dividir un nodo en la variable elegida y (3) cómo declarar un nodo ser terminal. Sin entrar en detalles, ahora esbozamos las ideas principales. Primero, se utilizan divisiones binarias en lugar de divisiones de múltiples vías para que los datos no se fragmenten demasiado rápido. En cualquier caso, las divisiones de múltiples vías pueden resultar de una serie de divisiones binarias.

El método del árbol de clasificación comienza en el nodo raíz con toda la muestra y primero selecciona el mejor predictor binario de la variable de respuesta. Esto produce dos nuevos nodos, cada uno de los cuales es candidato para una mayor división binaria. Para una variable ordinal o una variable cuantitativa como el ancho, la división toma la forma de valores que caen por encima o por debajo de un nivel particular. Para una variable nominal, la división se basa en ordenar las categorías por las proporciones de la muestra que caen en la categoría de respuesta de interés y luego usar el mismo criterio para seleccionar un punto de corte para separarlas en dos conjuntos de categorías. Para encontrar la primera división binaria, el algoritmo forma una tabla de clasificación para cada posible división binaria para cada variable explicativa. La división elegida satisface algún criterio de optimalidad, como maximizar la diferencia entre la desviación basada en la verosimilitud binomial para el modelo con una probabilidad común para todas las observaciones y el modelo que permite dos regiones disjuntas de valores x, cada una con una probabilidad común. A continuación, se utiliza el mismo procedimiento con cada nuevo nodo.

2) THE LOGISTIC REGRESSION MODEL

El modelo de regresión logística tiene una forma lineal para el logit de la probabilidad de éxito, es decir, el logaritmo de las probabilidades,

$$\text{logit}[\pi(x)] = \log\left(\frac{\pi(x)}{1-\pi(x)}\right) = \alpha + \beta x$$

Para x cuantitativo, la fórmula implica que $\pi(x)$ cambia como una función en forma de S de x,. La regresión logística tiene una fórmula correspondiente para $\pi(x)$, usando la función exponencial $\exp(\alpha + \beta x) = e\alpha + \beta x$

$$\pi(x) = \frac{e^{\alpha+\beta x}}{1 + e^{\alpha+\beta x}}$$

El parámetro de efecto β determina la tasa de aumento o disminución de la curva en forma de S para $\pi(x)$. El signo de β indica si la curva asciende ($\beta > 0$) o desciende ($\beta < 0$).

La tasa de cambio aumenta cuando $|\beta|$ aumenta. Cuando $\beta = 0$, la curva se aplana a una línea recta horizontal. La variable de respuesta binaria es entonces independiente de la variable explicativa. [1]

La fórmula de regresión logística indica que el logit aumenta en β por cada aumento de 1 unidad en x. La mayoría de nosotros no pensamos de forma natural en una escala logit, por lo que a continuación sugerimos interpretaciones alternativas.

Al exponenciar ambos lados de la ecuación de regresión logística, obtenemos una interpretación que utiliza las probabilidades y la razón de probabilidades. Las probabilidades de éxito son

$$\frac{\pi(x)}{1-\pi(x)} = \exp(\alpha + \beta x) = e^{\alpha}\left(e^{\beta}\right)^{x}.$$

Por lo tanto, las probabilidades se multiplican por eβ por cada aumento de 1 unidad en x. Es decir, las probabilidades en el nivel x + 1 son iguales a las probabilidades en x multiplicadas por eβ. Cuando β = 0, eβ = 1, y las probabilidades no cambian a medida que cambia x.

Una interpretación más simple se refiere a la probabilidad π (x) en sí. La Figura 4.1 muestra la apariencia en forma de S del modelo para π (x), como se ajusta para el siguiente ejemplo. Dado que es una curva en lugar de una línea recta, la tasa de cambio en π (x) por 1 unidad de aumento en x depende del valor de x. Una línea recta trazada tangente a la curva en un Valor x particular, como se muestra en la Figura 4.1, describe la tasa de cambio en ese punto. Para el parámetro de regresión logística β, esa línea tiene una pendiente igual a βπ (x) [1 - π (x)]. Por ejemplo, la línea tangente a la curva en x para la cual π (x) = 0.50 tiene pendiente β (0.50) (0.50) = 0.25β; por el contrario, cuando π (x) = 0.90 o 0.10, tiene pendiente 0.09β. La pendiente se acerca a 0 cuando π (x) se acerca a 1.0 o 0. La pendiente más pronunciada ocurre cuando π (x) = 0.50. Ese valor de x se relaciona con los parámetros de regresión logística por 2 x = −α / β. Este valor de x a veces se denomina nivel medio efectivo. Representa el punto en el que cada resultado tiene un 50% de probabilidad [1]

E. APLICACIÓN DEL CLASIFICADOR

Una vez elegido el algoritmo Clasification Tree, implementamos este método utilizando la librería Sklearn y el método DecisionTreeClassifier.

Para la comprobación de los resultados hallamos la matriz de Confusión, la Precisión y la Exactitud.

Finalmente utilizamos un método de entrenamiento y testeo también de la librería sklearn que se denomina StratifiedShuffleSplit para 100 splits con un test size del 25%.

V. ANÁLISIS DE LOS COMPONENTES PRINCIPALES (PCA)

En estadística, el análisis de componentes principales (en español ACP, en inglés, PCA) es una técnica utilizada para describir un conjunto de datos en términos de nuevas variables («componentes») no correlacionadas. Los componentes se ordenan por la cantidad de varianza original que describen, por lo que la técnica es útil para reducir la dimensionalidad de un conjunto de datos.

Técnicamente, el ACP busca la proyección según la cual los datos queden mejor representados en términos de mínimos cuadrados. Esta convierte un conjunto de observaciones de variables posiblemente correlacionadas en un conjunto de valores de variables sin correlación lineal llamadas componentes principales.

El ACP se emplea sobre todo en análisis exploratorio de datos y para construir modelos predictivos. El ACP comporta el cálculo de la descomposición en autovalores de la matriz de covarianza, normalmente tras centrar los datos en la media de cada atributo.

Debe diferenciarse del análisis factorial con el que tiene similitudes formales y en el cual puede ser utilizado como un método de aproximación para la extracción de factores. [2]

A. FUNDAMENTO

El ACP construye una transformación lineal que escoge un nuevo sistema de coordenadas para el conjunto original de datos en el cual la varianza de mayor tamaño del conjunto de datos es capturada en el primer eje (llamado el Primer Componente Principal), la segunda varianza más grande es el segundo eje, y así sucesivamente. Para construir esta transformación lineal debe construirse primero la matriz de covarianza o matriz de coeficientes de correlación. Debido a la simetría de esta matriz existe una base completa de vectores propios de la misma. La transformación que lleva de las antiguas coordenadas a las coordenadas de la nueva base es precisamente la transformación lineal necesaria para reducir la dimensionalidad de datos. Además, las coordenadas en la nueva base dan la composición en factores subyacentes de los datos iniciales.

El ACP es particularmente útil para reducir la dimensionalidad de un grupo de datos. Los primeros componentes principales describen la mayor parte de la varianza de los datos (más cuanto más correlacionadas estuvieran las variables originales). Estos componentes de bajo orden a veces contienen el aspecto "más importante" de la información, y los demás componentes se pueden ignorar. Existen diferentes técnicas para estimar el número de componentes principales que son relevantes; la técnica más apropiada dependerá de la estructura de correlaciones en los datos originales. [2]

B. MATEMATICAS DEL PCA

Supongamos que existe una muestra con n individuos para cada uno de los cuales se han medido m variables (aleatorias) El PCA permite encontrar un número de factores subyacentes p < m que explican aproximadamente el valor de las m variables para cada individuo. El hecho de que existan estos p factores subyacentes puede interpretarse como una reducción de la dimensionalidad de los datos: donde antes necesitábamos m valores para caracterizar a cada individuo ahora nos bastan p valores. Cada uno de los p encontrados se llama componente principal, de ahí el nombre del método.

Existen dos formas básicas de aplicar el ACP:

Método basado en la matriz de correlación, cuando los datos no son dimensionalmente homogéneos o el orden de magnitud de las variables aleatorias medidas no es el mismo.

Método basado en la matriz de covarianzas, que se usa cuando los datos son dimensionalmente homogéneos y presentan valores medios similares. [2]

C. Método basado en correlaciones

El método parte de la matriz de correlaciones, consideremos el valor de cada una de las m variables aleatorias. Para cada uno de los n individuos tomemos el valor de estas variables y escribamos el conjunto de datos en forma de matriz:

$$(F_j^\beta)_{j=1,\dots,m}^{\beta=1,\dots,n}.$$

Obsérvese que cada conjunto:

$$\mathcal{M}_j = \{F_j^\beta | \beta = 1, \dots, n\}$$

puede considerarse una muestra aleatoria para la variable A partir de datos correspondientes a las m variables aleatorias, puede construirse la matriz de correlación muestral, que viene definida por:

$$\mathbf{R} = [r_{ij}] \in M_{m\times m}\,, \qquad \text{donde} \qquad r_{ij} = \frac{\operatorname{cov}(F_i, F_j)}{\sqrt{\operatorname{var}(F_i)\operatorname{var}(F_j)}}\,.$$

Puesto que la matriz de correlaciones es simétrica entonces resulta diagonalizable y sus valores propios verifican:

$$\sum_{i=1}^{m} \lambda_i = m$$

Debido a la propiedad anterior estos m valores propios reciben el nombre de pesos de cada uno de los m componentes principales. Los factores principales identificados matemáticamente se representan por la base de vectores propios de la matriz. Está claro que cada una de las variables puede ser expresada como combinación lineal de los vectores propios o componentes principales. [2]

D. Método basado en las covarianzas

El objetivo es transformar un conjunto dado de datos X de dimensión n x m a otro conjunto de datos Y de menor dimensión n x l con la menor perdida de información útil posible utilizando para ello la matriz de covarianza.

Se parte de un conjunto n de muestras cada una de las cuales tiene m variables que las describen y el objetivo es que, cada una de esas muestras, se describa con solo I variables, donde l < m. Además, el número de componentes principales l tiene que ser inferior a la menor de las dimensiones de X.

$$l \leq \min\{n, m\}$$

Los datos para el análisis tienen que estar centrados a media 0 (restándoles la media de cada columna) y/o autoescalados(centrados a media 0 y dividiendo cada columna por su desviación estándar).

$$\mathbf{X} = \sum_{a=1}^{l} \mathbf{t}_a \mathbf{p}_a^T + \mathbf{E}$$

Los vectores son conocidos como scores y contienen la información de cómo las muestras están relacionadas unas con otras además, tienen la propiedad de ser ortogonales. Los vectores se llaman loadings e informan de la relación existente entre las variables y tienen la cualidad de ser ortonormales. Al coger menos componentes principales que variables y debido al error de ajuste del modelo con los datos, se produce un error que se acumula en la matriz

El PCA se basa en la descomposición en vectores propios de la matriz de covarianza. La cual se calcula con la siguiente ecuación:

$$\operatorname{cov}(X) = \frac{X^T X}{n-1}$$
$$\operatorname{cov}(X)\, \mathbf{p}_a = \lambda_a\, \mathbf{p}_a$$
$$\sum_{a=1}^{m} \lambda_a = 1$$

Donde es el valor propio asociado al vector propio. Por último,

$$\mathbf{t}_a = X\, \mathbf{p}_a$$

Esta ecuación la podemos entender como que, donde los valores propios miden la cantidad de varianza capturada, es decir, la información que representan cada uno de los componentes principales. La cantidad de información que captura cada componente principal va disminuyendo según su número, es decir, el componente principal número uno representa más información que el dos y así sucesivamente. [2]

VI. CONCLUSIONES

El estudio del interés en cambiar de trabajo de los científicos de datos para pertenecer a la empresa de Big Data y Data Science depende de muchas variables que al interactuar entre ellas determinaran si el empleado está o no interesado en formar parte de este nuevo equipo para lo cual se utiliza la inteligencia artificial para identificar que empleados están realmente interesados en ser parte de este equipo.

La reducción de la matriz de datos utilizando PCA nos brinda resultados similares al estudio realizado sin la reducción de la matriz de datos.

REFERENCIAS

[1] A. AGRESTI, AN INTRODUCTION TO CATEGORICAL DATA ANALYSIS, Hoboken: Wiley, 2019.

[2] Comunidad, «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/An%C3%A1lisis_de_componentes_principales.