Universidad Autónoma del Estado de México

Facultad de Ingeniería

Ciencia de los Datos

Prof. Luis Enrique Ledezma Fuentes

Modelo de predicción GBM para Salarios

Proyecto para Primer Examen Parcial

Jorge de Jesús Jiménez Servín | 2121168

Erick López González | 1710498

Grupo O2

Octubre 2024

Contenido

Introducción 3
Desarrollo 4
Extracción De Datos 4
Preparación De Los Datos 5
EDA: Exploración Y Visualización De Los Datos 8
Modelo Predictivo 11
Semilla de aleatoriedad (seed = 6699): 11
Control del entrenamiento (trainControl): 12
Creación del modelo GBM: 12
Predicción del salario: 13
Validación Del Modelo 13
Conclusión 15
Despliegue 16
Bibliografía 17

Introducción

El salario es un indicador económico clave que refleja la remuneración que recibe un individuo por su trabajo y está influenciado por diversos factores como la educación, la experiencia y la industria en la que se desempeña. En el contexto del análisis predictivo, comprender los determinantes del salario es esencial tanto para los profesionales de recursos humanos como para los economistas y responsables de políticas laborales, ya que permite realizar estimaciones sobre la equidad salarial, tendencias de empleo y políticas de compensación.

El presente proyecto de ciencia de datos tiene el objetivo de predecir el salario esperado de un individuo basado en un conjunto de características altamente descriptivas como lo son la edad, el género, el grado de educación, el título de profesión y los años de experiencia.

La metodología consiste en aplicar una preparación de los datos, lo que implica una limpieza y tratamiento de estos para que sean adecuados para el análisis; la exploración y visualización de los datos mediante EDA's basado en un modelo de regresión lineal simple, la generación de un modelo predictivo de gradiente estocástico, validar el modelo y desplegar el modelo para poder entregar conclusiones.

Dentro de la preparación de los datos se contempla el uso de componentes principales para determinar que variables son las que estadísticamente aportaran más al generar un modelo de regresión lineal simple para el EDA.

Posteriormente, se construye un modelo predictivo utilizando el paquete Caret, el cual facilita la implementación de una variedad de métodos complejos de clasificación y regresión. En este caso, se utiliza el algoritmo GBM (Stochastic Gradient Boosting), que combina múltiples modelos para mejorar la precisión en las predicciones. Este enfoque es adecuado tanto para problemas de regresión como de clasificación, y ha demostrado su capacidad para captar relaciones no lineales entre las variables predictoras y el resultado esperado.

Desarrollo

En este proyecto de Ciencia de Datos, seguiremos un esquema que corresponde a los pasos típicos de un Análisis de Datos. Estos pasos nos permitirán comprender la estructura y las características de nuestro conjunto de datos, así como identificar patrones, tendencias y relaciones significativas entre variables.

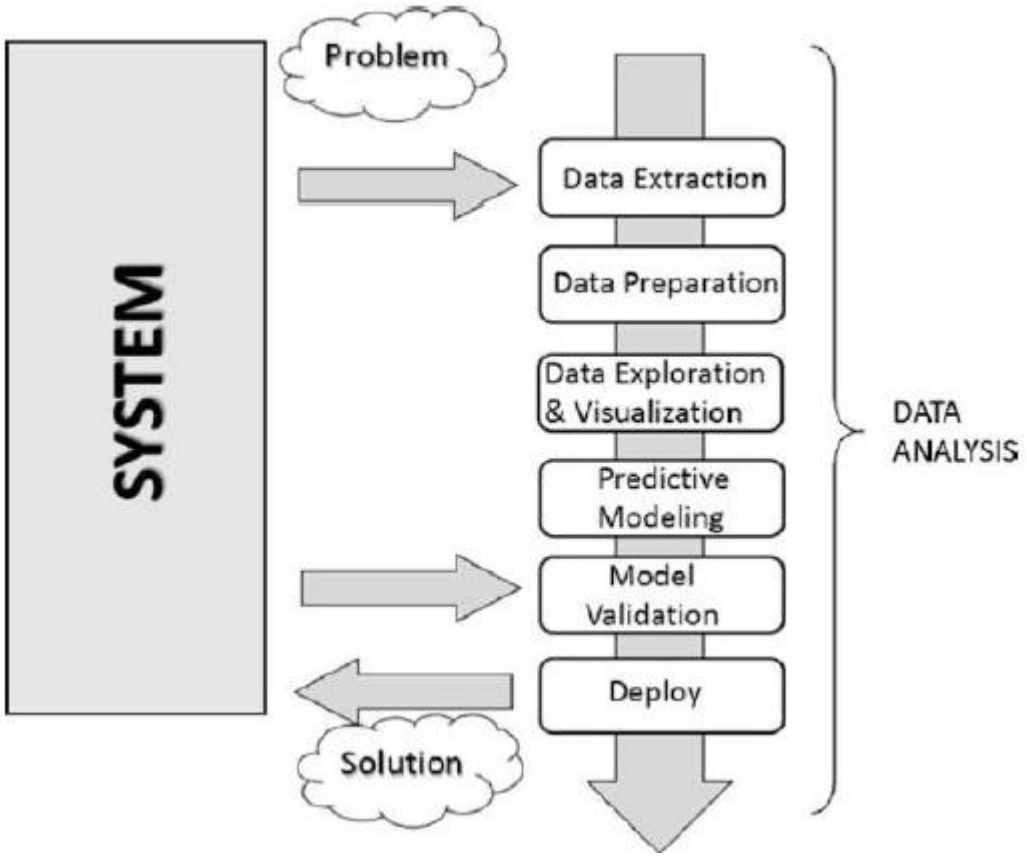

Imagen 1: Metodología de un proyecto de análisis de datos en Ciencia de Datos

Extracción De Datos

Los datos con los que se trabajara para realizar el modelo predictivo fueron dispuestos por el profesor responsable de asignatura, de este modo, la metodología de extracción esta implícita y se supone la tabulación de los datos en un archivo .csv como los datos finales resultado de algún debido proceso de extracción.

```
#Obteniendo los datos
training_data <- read.csv("Salary_Data.csv")
```

Imagen 2: Línea de código que obtiene los datos desde el .csv

Preparación De Los Datos

Para la preparación de los datos se realiza una ***limpieza*** de aquellos registros que carecen de alguna característica (Edad, Género, Nivel de educación, Cargo, Años de Experiencia o Salario).

- Se encontraron 5 registros con alguno(s) de los campos nulos. Se eliminan del Data Frame.

```
# Revisando que renglones contienen datos nulos
training_data[!complete.cases(training_data), ]

# Quitando datos nulos
no_null_dt <- na.omit(training_data)
no_null_dt[!complete.cases(no_null_dt), ]
```

Imagen 3: Líneas responsables de la limpieza de datos

Posteriormente se realizó una ***transformación*** de la variable categórica género a una variable numérica:

- 1: Hombre
- 0: Mujer

```
# Transformando la variable genero categorica a numerica
no_null_dt$Gender <- ifelse(no_null_dt$Gender== "Male", 1, 0)
str(no_null_dt)
```

Imagen 4: Líneas transformando y mostrando resultado de variable categórica a numérica

Se implemento ***PCA*** (Principal Component Analysis – Análisis de Componentes Principales) para detectar cuales son las variables que sustancialmente aportan más al modelo y así destinar la generación del modelo predictivo bajo esas variables evitando utilizar las que no aportan la variabilidad suficiente y ahorrando procesamiento sobrante.

Se utilizaron solo las variables numéricas para el PCA:

- Edad
- Genero [0,1]
- Años de Experiencia
- Salario

```r
# 3: Análisis de Componentes Principales (PCA)

# Seleccionar solo las variables numéricas para el PCA
numeric_vars <- no_null_dt %>% select(Age, Salary, Years.of.Experience, Gender)

# Estandarización de las variables
scaled_data <- scale(numeric_vars)

# Realizar PCA
pca_result <- PCA(scaled_data, graph = FALSE)

# Mostrar resumen del PCA (contribución de cada componente)
summary(pca_result)

# Visualización de la varianza explicada por cada componente
fviz_eig(pca_result, addlabels = TRUE, ylim = c(0, 100))

# Visualización de las variables en el plano de los primeros dos componentes principales
fviz_pca_var(pca_result, col.var = "contrib",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE) # Evita superposición de etiquetas

# Visualización de las observaciones en el plano de los primeros dos componentes principales
fviz_pca_ind(pca_result, col.ind = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE) # Evita superposición de etiquetas
```

Imagen 5: Líneas responsables del PCA

Resultados del PCA

Ordenados del de más impacto al menor.

1) Genero
2) Años de Experiencia
3) Edad
4) Salario: No aplica

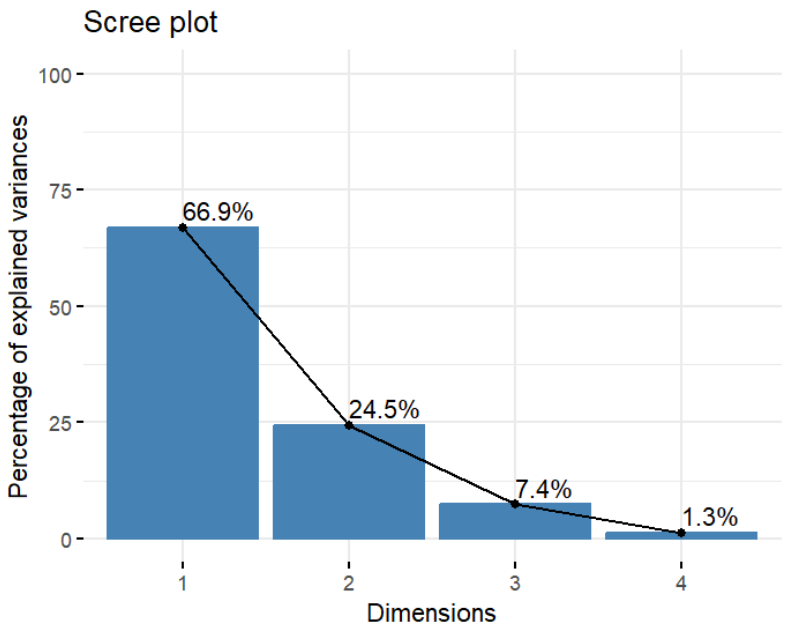

Imagen 6: Visualización de la varianza explicada por cada componente

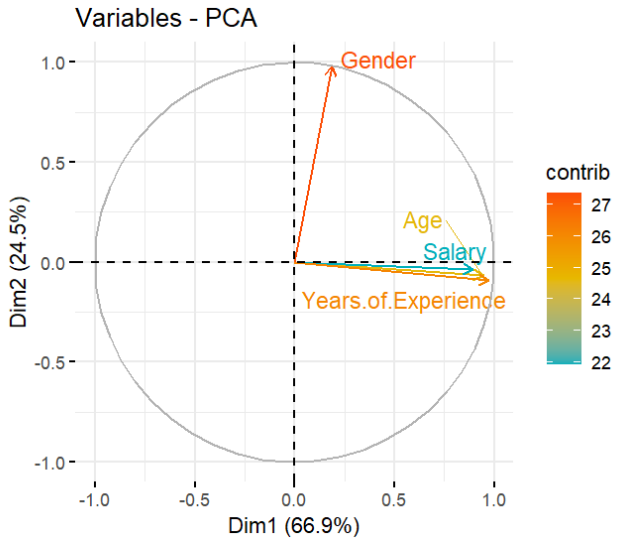

Imagen 7: Visualización de las variables en el plano de los primeros dos componentes principales

Ahora bien, se concluyó que no puede elegirse la variable género como variable independiente para un futuro modelo de regresión lineal que nos sirva para realizar un EDA, ya que, aunque el resultado del PCA es contundente, de manera actual, el modelo podría tener cierto sesgo al presentársele datos actuales para usar el modelo debiéndose principalmente a que la diferencia del salario entre géneros se ha visto

difuminada por las regulaciones modernas en términos de igualdad de género en el aspecto laboral.

Por tanto, la variable elegida será para realizar el modelo de EDA es: **Años de experiencia.**

EDA: Exploración Y Visualización De Los Datos

En este análisis exploratorio de datos (EDA) se realiza los siguientes pasos claves para entender la relación entre las variables de interés y explorar las características principales de los datos:

1. **Resumen estadístico de las variables salario y años de experiencia**

   Se genera un resumen estadístico de las variables en la base de datos, lo cual permite observar los valores mínimos, máximos, medias, medianas y otros percentiles importantes. Esto nos proporciona un panorama general de la distribución de los salarios y la experiencia en el conjunto de datos.

```
# Resumen estadístico de las variables de interés
summary(no_null_dt$Salary)
summary(no_null_dt$Years.of.Experience)
```

Imagen 8: Líneas de código correspondiente a la impresión del resumen en consola

```
> # Resumen estadístico de las variables de interés
> summary(no_null_dt$Salary)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    350   70000  115000  115327  160000  250000
> summary(no_null_dt$Years.of.Experience)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   3.000   7.000   8.095  12.000  34.000
```

Imagen 9: Resultad del resumen estadístico de las variables salario y años de experiencia

2. **Grafica de dispersión con línea de tendencia**

   Se generó una gráfica de dispersión que muestra la relación entre el salario y los años de experiencia. Agregando una línea de tendencia ajustada por un modelo

de regresión lineal simple permitiéndonos visualizar si existe una correlación positiva o negativa entre los años de experiencia y el salario.

```
ggplot(no_null_dt, aes(x = Years.of.Experience, y = Salary)) +
  geom_point(color = "blue", size = 2) +  # puntos de dispersión
  geom_smooth(method = "lm", se = FALSE, color = "red") +  # línea de tendencia
  labs(title = "Scatter Plot of Salary vs Years of Experience",
       x = "Years of Experience",
       y = "Salary") +
  theme_minimal()
```

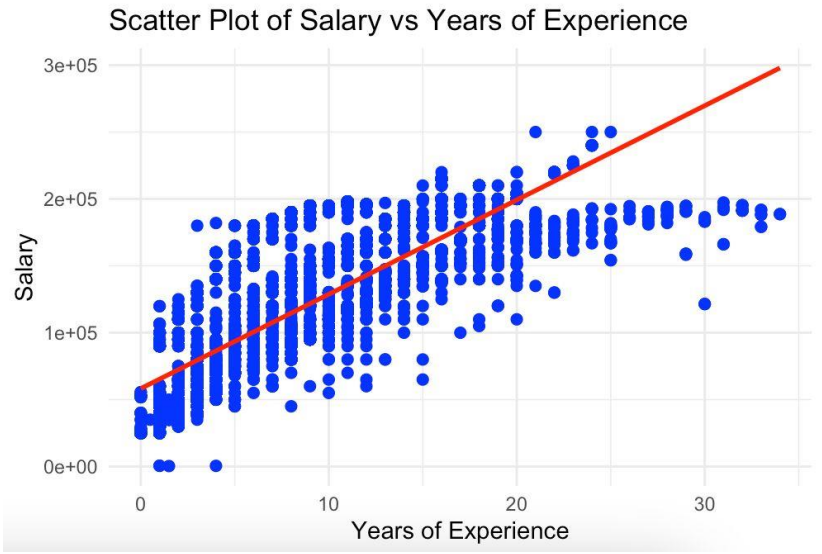

Imagen 3: Gráfica de dispersión

3. Cálculo del coeficiente de determinación

El modelo fue ajustado para calcular el coeficiente de determinación R^2. Un valor de R^2 más cercano a 1 indica que el modelo ajusta mejor los datos.

```
# Ajuste del modelo lineal simple
linear_model <- lm(Salary ~ Years.of.Experience, data = no_null_dt)

# Mostrar el resumen del modelo lineal para ver R²
summary(linear_model)

# Extraer el valor de R² del modelo lineal
r_squared <- summary(linear_model)$r.squared
cat("R² del modelo lineal:", r_squared, "\n")
```

```
> # Mostrar el resumen del modelo lineal para ver R²
> summary(linear_model)

Call:
lm(formula = Salary ~ Years.of.Experience, data = no_null_dt)

Residuals:
    Min      1Q  Median      3Q     Max
-148236  -22377   -5564   21015  100576

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         58283.28     632.71   92.12   <2e-16 ***
Years.of.Experience  7046.77      62.57  112.62   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 31030 on 6697 degrees of freedom
Multiple R-squared:  0.6544,    Adjusted R-squared:  0.6544
F-statistic: 1.268e+04 on 1 and 6697 DF,  p-value: < 2.2e-16

>
> # Extraer el valor de R² del modelo lineal
> r_squared <- summary(linear_model)$r.squared
> cat("R² del modelo lineal:", r_squared, "\n")
R² del modelo lineal: 0.6544307
```

La gráfica de dispersión muestra una relación positiva entre los años de experiencia y el salario, lo que indica que a medida que los años de experiencia aumentan, el salario también tiende a incrementarse.

Por otro lado, el coeficiente de determinación $R^2 = 0.6$ implica que el 60% de la variabilidad en el salario puede explicarse por los años de experiencia. Este valor es moderado lo que significa que, aunque hay una relación entre la experiencia y el salario, existen otros factores que también influyen como es el caso de la edad, o el género.

Modelo Predictivo

Se construye un modelo predictivo (veasé sección 6 del código en R) utilizando un modelo GBM (Gradient Boosting Machine) para predecir los salarios basándose en diferentes variables del conjunto de datos, como:

- Edad
- Genero
- Nivel de educación
- Profesion
- Años de experiencia.

A continuación, se describe de manera más detallada cómo funciona esta sección y su configuración.

```r
#6: Creación de un Modelo Predictivo

set.seed(6699)

# Control del entrenamiento: validación cruzada
trCtrl <- trainControl(method = "cv", number = 10)

# Modelo GBM (Gradient Boosting Machine) para predicción de Salary
gbm_model <- train(Salary ~ Age + Gender + Education.Level + Job.Title +
                     Years.of.Experience, trControl = trCtrl,
                   method = "gbm", data = no_null_dt, verbose = FALSE)

# Mostrar el resumen del modelo
print(gbm_model)
```

Imagen 10: Líneas responsables de la generación del modelo predictivo gbm

Semilla de aleatoriedad (seed = 6699):

La línea `set.seed(6699)` se utiliza para asegurar que los resultados del modelo sean reproducibles. La semilla 6699 es un valor numérico que garantiza que los procesos aleatorios dentro del modelo (como la división de datos en la validación cruzada) se realicen de la misma manera cada vez que se ejecute el código. Esto es útil en entornos de desarrollo y evaluación de modelos para mantener coherencia en los resultados.

Control del entrenamiento (trainControl):

La función `trainControl(method = "cv", number = 10)` especifica el tipo de validación que se utilizará para el entrenamiento del modelo. En este caso:

- `cv (cross-validation)`: Se está utilizando validación cruzada.
- `number = 10`: Se realizarán 10 iteraciones, es decir, se divide el conjunto de datos en 10 subconjuntos (folds). El modelo se entrena con 9 de ellos y se valida con el subconjunto restante, repitiendo este proceso hasta que todos los subconjuntos se utilicen como conjunto de validación. Este enfoque ayuda a evitar el sobreajuste y proporciona una estimación más fiable del rendimiento del modelo.

Creación del modelo GBM:

El modelo `gbm_model` se crea usando la función `train` de la librería `caret`, que realiza el entrenamiento de un modelo de **Boosting de Gradiente (GBM)**. Este modelo es un tipo de ensamblaje que combina múltiples árboles de decisión débiles (submodelos) para mejorar la precisión de las predicciones. El código siguiente define las variables predictoras y la variable objetivo:

```
# Modelo GBM (Gradient Boosting Machine) para predicción de Salary
gbm_model <- train(Salary ~ Age + Gender + Education.Level + Job.Title +
                     Years.of.Experience, trControl = trCtrl,
                   method = "gbm", data = no_null_dt, verbose = FALSE)
```

Imagen 11: Definiendo las variables predictoras (después del ~) y la variable objetivo Salary (antes del ~)

Variables predictoras:

- Edad (Age): Representa la edad de los empleados.
- Género (Gender): Variable codificada como 0 (femenino) o 1 (masculino).
- Nivel de educación (Education.Level): El nivel de educación de los empleados.
- Profesión (Job.Title): La profesión del empleado.
- Años de experiencia (Years.of.Experience): Los años de experiencia laboral del empleado.

Variable objetivo:

- Salary: El salario que se desea predecir.

Predicción del salario:

Una vez entrenado el modelo GBM, se utilizan los datos de entrenamiento (`no_null_dt`) para realizar predicciones de salarios:

```
# Realizar predicciones sobre los datos de entrenamiento
salary_predictions <- predict(gbm_model, no_null_dt)
```

Imagen 12: Realizando predicción

Estas predicciones permiten comparar los **valores reales** de los salarios con los **valores predichos** por el modelo.

Visualizando un resumen de las primeras predicciones.

```
> head(data.frame(Real = no_null_dt$Salary, Predicted = salary_predictions))
    Real Predicted
1  90000 152717.81
2  65000 105589.66
3 150000 164182.23
4  60000  92746.05
5 200000 180825.54
6  55000  67132.19
>
```

Imagen 13: Visualización de las primeras predicciones real vs predicho

Validación Del Modelo

Para evaluar el rendimiento del modelo se utilizan dos métricas principales:

- RMSE (Root Mean Squared Error): Se calcula para medir la diferencia promedio entre los valores reales y los predichos. Un RMSE más bajo indica un mejor ajuste del modelo.

```
# Cálculo de RMSE (Root Mean Squared Error)
rmse_value <- sqrt(mean((no_null_dt$Salary - salary_predictions)^2))
cat("RMSE del modelo: ", rmse_value, "\n")
```

- R^2 (Coeficiente de determinación): Mide la proporción de la variabilidad de la variable dependiente que es explicada por las variables independientes. Un

valor de R^2 cercano a 1 indica un buen ajuste, es decir, indica la capacidad del modelo para predecir el resultado.

```
# Cálculo de R² para ver la proporción de varianza explicada por el modelo
r_squared <- cor(no_null_dt$Salary, salary_predictions)^2
cat("R² del modelo: ", r_squared, "\n")
```

En definitiva, el modelo de regresión-clasificación gbm muestra una mejor predicción de los salarios utilizando la totalidad de variables independientes.

Se puede observar que la media cuadrada del error medio en la predicción de los datos es de $\pm\$13820.27$. Lo que nos indica que es un error razonable. RMSE se comporta de una manera especial considerando el rango de valores del salario. Esta métrica es sensible a datos muy grandes (por elevarse al cuadrado) pero es funcional estadísticamente. Por ejemplo, en este casi se está prediciendo salarios, donde estos oscilan entre $350 y $250,000, un RMSE de $5000 a $13,900 podría considerarse razonable. Sin embargo, si los salarios variaran entre $20,000 y $30,000, un RMSE de $5000 sería elevado.

Ahora bien, el valor de $\boldsymbol{R^2}$ **implica que el 93%** de la variabilidad en el salario puede explicarse por la ***edad, el género, nivel de educación, profesión y años de experiencia***. Este valor es alto lo que significa que hay una relación muy fuerte entre las variables independientes y el salario, por lo que es mejor modelo que el modelo lineal simple presentando para el desarrollo del EDA.

```
>
>
>
> summary(no_null_dt$Salary)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    350   70000  115000  115327  160000  250000
```

Imagen 14: Rango de salarios

```
>
>
>
> # Cálculo de RMSE (Root Mean Squared Error)
> rmse_value <- sqrt(mean((no_null_dt$Salary - salary_predictions)^2))
> cat("RMSE del modelo: ", rmse_value, "\n")
RMSE del modelo:  13820.27
>
> # Cálculo de R² para ver la proporción de varianza explicada por el modelo
> r_squared <- cor(no_null_dt$Salary, salary_predictions)^2
> cat("R² del modelo: ", r_squared, "\n")
R² del modelo:  0.9330135
>
```

Imagen 15: Resultado de las métricas de R^2 y de RMSE

Conclusión

El proyecto de ciencia de datos desarrollado tiene como objetivo predecir el salario esperado de un individuo considerando variables clave como la edad, género, nivel de educación, profesión y años de experiencia. A lo largo del proceso, se llevó a cabo una preparación y análisis exhaustivo de los datos, empleando técnicas como el Análisis de Componentes Principales (PCA) y la Exploración de Datos (EDA). Estas herramientas fueron fundamentales para identificar las variables más influyentes y descartar posibles sesgos o redundancias en los datos.

El modelo predictivo se construyó utilizando el algoritmo Gradient Boosting Machine (GBM), lo que permitió alcanzar una precisión elevada, logrando explicar el 93% de la variabilidad en los salarios. Este resultado evidencia una fuerte correlación entre las variables seleccionadas y el salario predicho, superando en rendimiento a modelos más sencillos como el de regresión lineal.

Además, la elección del GBM subraya la importancia de utilizar algoritmos avanzados que capturen mejor la complejidad de los datos y las interacciones entre las variables. En conjunto, este proyecto muestra cómo una adecuada selección de características y el uso de técnicas robustas de machine learning pueden mejorar significativamente la capacidad predictiva en escenarios reales.

Despliegue

El despliegue de este modelo predictivo se realiza mediante la publicación del código fuente y todos los recursos asociados en un repositorio público de GitHub. Esto incluye tanto el script en R que ejecuta el modelo predictivo como los archivos de datos necesarios, como el archivo `Salary_Data.csv` utilizado para entrenar y validar el modelo.

El repositorio en GitHub permitirá que otros usuarios y colaboradores puedan:

- Acceder al código completo del modelo.
- Descargar los datos de entrenamiento para realizar sus propias pruebas o análisis.
- Ejecutar el código en sus propios entornos y verificar los resultados.
- Modificar y mejorar el modelo a partir de los recursos compartidos.

Este tipo de despliegue garantiza la transparencia, accesibilidad y replicabilidad del modelo, facilitando su uso y evolución en futuras iteraciones.

Bibliografía

RPubs - Funciones Principales de la Libreria Caret. (n.d.). https://rpubs.com/chzelada/279724

Kuhn, M. (2019, March 27). 7 train Models By Tag | The caret Package. https://topepo.github.io/caret/train-models-by-tag.html#accepts-case-weights