ESTUDIOS DE FORMACIÓN PERMANENTE

BIG DATA Y BUSINESS ANALYTICS

MODELO DE IDENTIFICACIÓN DEL AHORRO A PARTIR DE LOS INGRESOS Y EGRESOS VISUALIZADOS EN LOS ESTADOS DE CUENTA BANCARIOS, PARA PREDECIR RESULTADOS FUTUROS Y PROYECTAR ESCENARIOS.

MENDIZABAL ARCAS, XABIER

MADRID, 12 DE SEPTIEMBRE DE 2024

RESUMEN

Este Trabajo de Fin de Máster se centra en el diseño y desarrollo de un modelo de machine learning aplicado al análisis de estados de cuenta bancarios. El proceso comienza con la extracción de datasets, que posteriormente, se limpian y preparan para su análisis.

Buscando la consolidación de ingresos, gastos y ahorro, los datos se actualizan mensualmente. Seguidamente, se analizan para calcular diversas métricas y se exploran las relaciones entre las variables, a través de coeficientes de correlación, histogramas, gráficos por pares y boxplots. Tras identificar y eliminar valores extremos, se genera una nueva correlación que permite un análisis más preciso.

El núcleo del proyecto es la etapa de modelado. Se construyen modelos que relacionan el ahorro con los ingresos, así como modelos que analizan los gastos básicos en función de los ingresos. Por otra parte, se predicen nuevos valores, y se realiza una visualización de los resultados por comunidad autónoma.

SUMMARY

This Master's Thesis focuses on the design and development of a machine learning model applied to the analysis of bank account statements. The process begins with the extraction of datasets, which are then cleaned and prepared for analysis.

Aiming to consolidate income, expenses, and savings, the data is updated monthly. Next, the data is analyzed to calculate various metrics and explore the relationships between variables through correlation coefficients, histograms, pair plots, and boxplots. After identifying and eliminating outliers, a new correlation is generated, allowing for more accurate analysis.

The core of the project is the modeling stage. Models are constructed to relate savings to income, as well as models that analyze basic expenses based on income. Additionally, new values are predicted, and the results are visualized by autonomous community.

ÍNDICE

|  |  |
| --- | --- |
| 1.Introducción ……………………………………………………………………………….………… | 5 |
| *1.1 Objetivos* …………………………………………………………………………….……….……. | 6 |
| *1.1.1. Objetivos generales* …………………………………………………………………….…. | 6 |
| *1.1.2. Objetivos específicos* ……………………………………………………………………… | 6 |
| *1.2. Enfoque* …………………………………………….………………………………………..……… | 6 |
| *1.3. Metodología* …………………………………………………………………………………….… | 6 |
| 2. Diseño de un modelo *machine learning* ………………………………….…………… | 7 |
| *2.1. Etapa de extracción de los datasets* ……………………. …………………………… | 7 |
| 2.2. *Etapa de transformación del dataset de estados de cuenta bancaria….* | 8 |
| *2.3. Etapa de consolidación (totalización de datos)* ……………………………….. | 9 |
| *2.4. Etapa de análisis, métricas y gráficos* ……………………………………………… | 10 |
| *2.4.1. Métricas* …………………………………………………………………………………….… | 10 |
| *2.4.2. Coeficiente de correlación* ……………………………………………………………. | 10 |
| *2.4.3. Histogramas* ……………………………………………………………………………..…. | 11 |
| *2.4.4. Visualización gráfica por pares de ingresos, gastos y ahorro* ………… | 11 |
| *2.4.5. Gráficos de boxplot* ……………………………………………………………………… | 12 |
| *2.5. Eliminación de valores extremos y generación de nueva correlación* .... | 12 |
| *2.6. Etapa de modelado de regresión lineal…*…………………………………………… | 13 |
| *2.6.1* *Modelo de ahorro familiar en función de ingresos* ………………………….. | 13 |
| *2.6.2. Modelo de gastos básicos en función de ingresos* …………………………… | 13 |
| *2.6.3. Transformación del dataframe de salario neto por comunidad* ……… | 13 |
| *2.6.4. Predicción de nuevos valores* …………………………………………………………. | 14 |
| *2.6.5. Visualización de los resultados por comunidad autónoma* ……………..  *2.7. Modelo de regresión no lineal (polinomial)* ………………………………. | 16  17 |
| 3. Resultados …………………………………………………………………………………………… | 18 |
| 4. Conclusiones ………………………………………………………………………................... | 18 |
| 5. Fuentes bibliográficas y documentales …...………………………………………….. | 19 |
| 6. Anexos ………………………………………………………………………………………………… | 20 |

ÍNDICE DE FIGURAS

|  |  |  |
| --- | --- | --- |
| Figura 1: *Estado de cuenta 2020-2024* |  | 7 |
| Figura 2: *Salario neto promedio por comunidad autónoma*…………………… |  | 8 |
| Figura 3: *Conversión del importe a valor absoluto* ………………………………... |  | 8 |
| Figura 4: *Categorías del dataset*……………………………………………………………. |  | 8 |
| Figura 5: *Categorías del dataset después de la conversión* …………..……… |  | 8 |
| Figura 6: *Formato de las categorías* …………………………………………………… |  | 9 |
| Figura 7: *Categorías con nombres ajustados* ……………………………………….. |  | 9 |
| Figura 8: *Totalización de variables por mes*  …………………………………………. |  | 9 |
| Figura 9: *Consolidación de variables por mes* ……………………………………. |  | 9 |
| Figura 10: *Data de estadísticos por variable ...* …………………………………….. |  | 10 |
| Figura 11: *Coeficiente de correlación entre variables* …………………………… |  | 10 |
| Figura 12: *Histogramas* ……………………………………………………………………….. |  | 11 |
| Figura 13: *Visualización gráfica por pares de ingresos, gastos y ahorro* .. |  | 11 |
| Figura 14: G*ráficos de boxplot* …………………………………………………………….. |  | 12 |
| Figura 15: *Datos estadísticos por variables* ………………………………………….. |  | 12 |
| Figura 16: *Coeficiente de correlación entre variables normalizadas* ……… |  | 13 |
| Figura 17: *Coeficientes entre ahorro e ingresos* ……………………………………. |  | 13 |
| Figura 18: *Coeficientes entre gastos básicos e ingresos* ………………………… |  | 14 |
| Figura 19: *Salario neto promedio y salario neto promedio por grupo familiar* |  | 15 |
| Figura 20: *Ingresos, gastos básicos y ahorro por comunidades* ……………… |  | 16 |
| Figura 21: *Comparativa familiar por comunidad* …………………………………… |  | 17 |
| Figura 22: *Comparativa gastos e ingresos con rangos..*…………………………… |  | 18 |

1. INTRODUCCIÓN

El complejo e inestable entorno económico actual ha cobrado una importancia fundamental en la economía personal de los ciudadanos, quienes encaran el reto de analizar con precisión sus estados de cuenta bancarios, que, a pesar de su nivel de detalle, resultan complicados para identificar patrones evidentes de ahorro y gasto. Esta dificultad, junto a la falta de herramientas sencillas para visualizar y proyectar escenarios financieros a largo plazo, limita la capacidad de las personas y sus asesores financieros para tomar las mejores decisiones. (Espiga, 2022).

Este Trabajo de Fin de Máster tiene como objetivo principal desarrollar, a partir de herramientas de machine learning, un modelo para identificar patrones de ahorro basado en los ingresos y egresos registrados en los estados de cuenta bancarios, y evaluar la viabilidad de diferentes decisiones económicas estratégicas.

Se acude a un enfoque práctico orientado a soluciones, el cual contempla desde la importación y transformación del dataset de estados de cuenta bancarios, hasta la aplicación de métodos de machine learning, específicamente la regresión lineal y regresión no lineal (polinomial), para la predicción de datos futuros.

Este trabajo se estructura en seis fases: inicialmente, se realiza la importación del dataset con los estados de cuenta bancarios de un grupo familiar; seguidamente, se transforma el dataset mediante la limpieza de datos, eliminación de datos extremos, consolidación de categorías y totalización mensual. En una tercera fase, se calculan las métricas, gráficas y se analizan los datos; a continuación, en la cuarta fase, se modelan los datos mediante machine learning. En una quinta fase, se predicen los datos futuros con datos de entrenamiento de las comunidades autónomas (CCAA). En la sexta, se generan los gráficos.

Además de contribuir en el campo del Big Data y el Business Analytics, se ofrece una herramienta útil para la gestión financiera personal y de asesoría financiera, que permite a los usuarios identificar oportunidades de ahorro y tomar decisiones informadas sobre su futuro financiero.

1.1. Objetivos

1.1.1. Objetivo general: Desarrollar un modelo que permita identificar el ahorro a partir de los ingresos y egresos visualizados en estados de cuenta bancarios, para predecir resultados futuros y proyectar escenarios.

1.1.2. Objetivos específicos:

- Importar y transformar el dataset de estados de cuenta bancarios, incluyendo la limpieza de datos, eliminación de valores extremos, consolidación y totalización de categorías por mes.

- Calcular métricas, analizar los datos y generar visualizaciones para interpretar los resultados de manera efectiva.

- Modelar los datos mediante técnicas de machine learning, específicamente utilizando el método de regresión lineal y regresión no lineal polinomial, para predecir escenarios financieros futuros.

1.2. Enfoque

El enfoque del presente TFM es práctico orientado a soluciones. En este, se identifica un problema real para abordarlo con los modelos y técnicas de Big Data que se adapten a la naturaleza del problema.

Se desarrolla una herramienta funcional que resuelve de forma efectiva y eficiente el problema. Se combinan diferentes metodologías y modelos hasta dar con aquellos que produzcan mejores resultados en los datos a analizar. Se utilizan técnicas métricas que permiten evaluar el rendimiento de cada modelo y justificar su elección, evaluando sus fortalezas y limitaciones. Se utilizan herramientas de visualización de datos, que permiten la interpretación de los resultados del análisis.

1.3. Metodología

Para el desarrollo de este trabajo, se utiliza la metodología ETL (extracción, transformación y carga), fuentes de datos y herramientas tecnológicas con el fin de construir un modelo robusto, que permita identificar patrones de ahorro y proyectar posibles escenarios de inversión, a partir de los estados de cuenta bancarios de un grupo familiar.

Se utilizan dos fuentes principales de datos: los estados de cuenta bancaria (2020-2024) de un grupo familiar, compuesto por tres miembros, dos de los cuales aportan ingresos mensuales y la base de datos de salario neto promedio por comunidad autónoma del Instituto Nacional de Estadística (INE).

Para la implementación y análisis de los datos, se emplean las siguientes herramientas colaborativas: Git y repositorio público de GitHub, los entornos de desarrollo RStudio y RMarkdown; librerías en R: knitr, rio, dlyr, tidyverse, lubridate, e1071, ggplot2 y patchwork y openxlsx.

El proceso de análisis de datos incluye gráficos de dispersión por pares, boxplots, histogramas, y gráficos de barras. para predecir patrones de ahorro y proyectar escenarios financieros, se acudió a los modelos de regresión lineal y regresión no lineal (polinomial) de machine learning.

Finalmente, como apoyo en la búsqueda de información y codificación, se acudió a Gemini.

1. DISEÑO DE UN MODELO DE *MACHINE LEARNING*

2.1. Etapa de extracción de los datasets

La extracción de datos es el primer proceso en el análisis de datos, que permite la obtención de datos desde diferentes fuentes.

Inicialmente, se **instalan las librerías; esto es** descargar e instalar las herramientas para acceder a los datos y manipularlos:

library(knitr), library(rio), library(dplyr), library(tidyverse), library(lubridate), library(e1071), library(ggplot2), library(patchwork), library(openxlsx)

Seguidamente, se importan los dataset necesarios: estados de cuenta bancarios (2020-2024) y salario neto por comunidad autónoma del INE (base =2022).

EdoCtaConsol2020\_2024 <- import("<https://raw.githubusercontent.com/Xabiertxu/TFMUned/master/EdoCtaConsol2020-2024.xlsx>")

Se obtienen 6271 filas y 11 columnas:

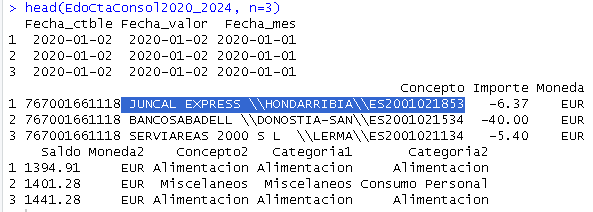


Figura 1: *Estado de cuenta 2020-2024* Fuente: Fuente: RStudio

SalarioNeto\_Comunidad <-import("<https://raw.githubusercontent.com/Xabiertxu/TFMUned/master/Salario_mensual_Comunidad_2022.xlsx>")

Se obtienen 18 filas y 6 columnas:

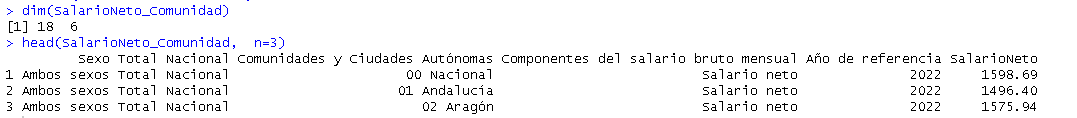


Figura 2: *Salario neto promedio por comunidad autónoma* Fuente: Fuente: RStudio

2.2. Etapa de transformación del dataframe de estados de cuenta bancaria

En esta etapa se limpia el dataframe, se convierten variables y se consolida la data. Como primer paso, se elimina la columna "Concepto" para anonimizar el dataframe. Seguidamente, se cambia el signo negativo (-) de las partidas de gastos y egresos en la variable Importe, y se crea una nueva variable “Importe\_abs”

Tabla

Descripción generada automáticamente

Figura 3: *Conversión del importe a valor absoluto* Fuente: Fuente: RStudio

A continuación, se seleccionan las cinco variables más representativas:



Figura 4: *Categorias del dataset inicial* Fuente: Fuente: RStudio

Se transforman los elementos de la variable "Categoria2" en nuevas variables utilizando la función pivot\_wider:

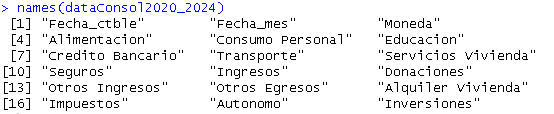


Figura 5: *Categorías del dataset después de la conversión* Fuente: Fuente: RStudio

Se sustituyen los valores nulos (NA), por ceros para poder totalizar las variables. Las variables “Fecha” se convierten en formato Date y el formato “yy-mm-dd” se lleva al formato “yy-mm”:

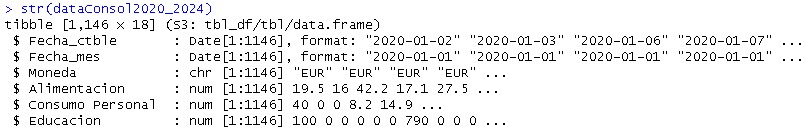


Figura 6:  *Formato de las categorías* Fuente: Fuente: RStudio

Por norma, los nombres de las categorías no pueden llevar espacios en blanco. En este caso, se unen las palabras.

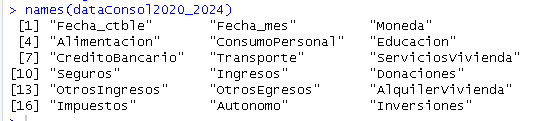


Figura 7: *Categorías con nombres ajustados* Fuente: Fuente: RStudio

2.3. Etapa de consolidación (totalización de datos).

La etapa de consolidación o totalización de datos consiste en agrupar y resumir los datos para obtener un panorama general.

Se totaliza cada variable numérica del dataframe por mes, utilizando las funciones: group\_by, summarise y sum. En la data seleccionada, se obtiene un dataframe de 54 filas y 16 columnas.

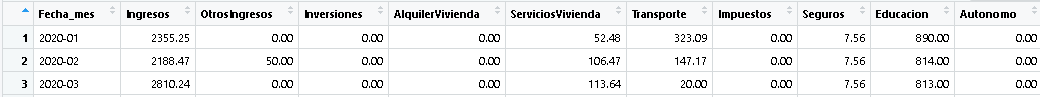


Figura 8: *Totalización de variables por mes* Fuente: Elaboración propia

Posteriormente, seleccionamos solo ingresos y gastos básicos recurrentes. Los ingresos están conformados por ingresos y otros ingresos. Los gastos básicos están conformados solo por servicios, transporte, seguros, educación, alimentación y consumo personal. Se crea la columna de ahorro por diferencia entre ingresos y gastos básicos. Como resultado se obtiene un dataframe de 54 filas y 4 columnas.

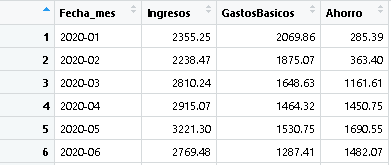


Figura 9: *Consolidación de variables por mes Fu*ente: Elaboración propia

2.4. Etapa de análisis, métricas y gráficos

Proceso en el que se analizan los datos estadísticamente con la finalidad detectar tendencias y relaciones, que se visualizan gráficamente (Dormido, 2024b)

2.4.1. Métricas

Se calculan media, mediana, Q1, Q3, mínimos y máximos del dataframe.

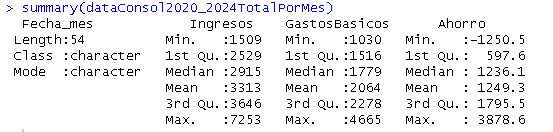


Figura 10: *Datos estadísticos por variable* Fuente: RStudio

Las tres categorías, muestran una gran dispersión de los datos si se comparan los mínimos y máximos con Q1-Q3. Esto es debido a que los ingresos y los gastos atienden a temporalidades. La media del ingreso familiar está en 3313€, los gastos en 2064€ y el ahorro resultante en 1249.3€. Por su parte, los gastos representan el 62.54% y el ahorro un 37.46%, lo que indica que gran parte de los ingresos se destinan a gastos. Si se analizan Q1-Q3, podemos observar que el 50% de los ingresos está distribuido entre 2529€ - 3646€, los gastos entre 1516€ - 2278€ y el ahorro entre 597.6€ - 1795.5€

2.4.2. Coeficiente de correlación

Se determina el coeficiente de correlación entre las variables ingresos, gastos básicos y el ahorro.

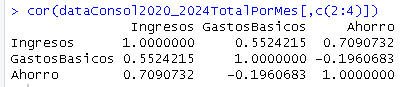


Figura 11:  *Coeficiente de correlación entre variables* Fuente: RStudio

Existe una correlación de 70.91 % entre el ingreso y el ahorro, lo que indica, que el modelo a elegir está relacionado con el ahorro en función de los ingresos. Hay una correlación moderada del 55.24 % entre los gastos básicos y el ahorro. se debe a que los gastos básicos, no guardan relación directa con los ingresos.

2.4.3. Histogramas de ingresos, gastos básicos y ahorro

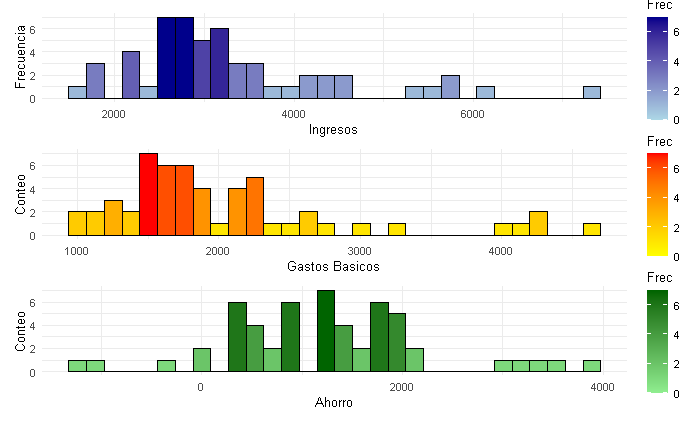


Figura 12: *Histogramas de ingresos, gastos básicos y ahorro* Fuente: Elaboración propia

Las gráficas de Ingresos y gastos básicos representan una distribución normal leptocúrtica con sesgo hacia la derecha, mostrando valores muy extremos como máximos. Los ingresos, se concentran principalmente entre 2529€ y 3646€ y la mediana en 2915€ y de media 3313€, es decir, que durante el período 2020-2024 hemos tenido períodos donde los ingresos han sido superiores a la mediana, tal como se aprecia en el histograma de ingresos.

Los gastos básicos están concentrados entre 1516€ y 2278€, mediana 1779€ y media 2064€, lo que nos indica, que durante los períodos 2020-2024 hemos tenido gastos superiores a la mediana, tal como se observa en histograma de gastos básicos.

El ahorro, representa una curva más normal con mediana (1236.1€) y media (1249.3€) y leptocúrtica. El ahorro está balanceado, sin embargo, tenemos puntos extremos muy alejados, tal como se ve en el histograma de ahorro.

2.4.4. Visualización gráfica por pares de ingresos, gastos y ahorro

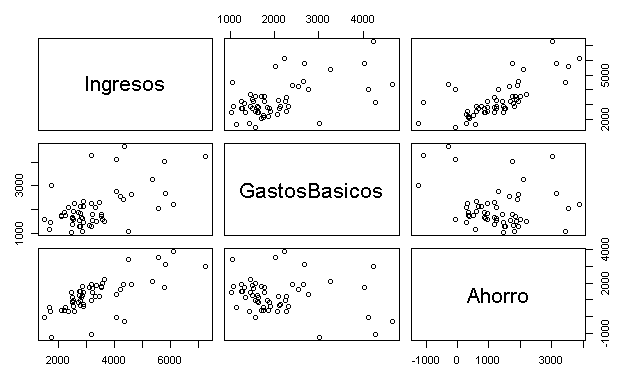


Figura 13: *Visualización gráfica por pares de ingresos, gastos y ahorro* Fuente: RStudio

Para determinar las relaciones entre pares de variables, se representa una matriz de diagramas de dispersión utilizando el comando en R: pairs() (Mendoza, s.f.).

Se observa una relación lineal fuerte entre los ingresos y el ahorro, mientras que los gastos básicos no guardan relación lineal con los ingresos y el ahorro, ya que los gastos, tienen un fuerte componente fijo.

2.4.5 Graficación de boxplot

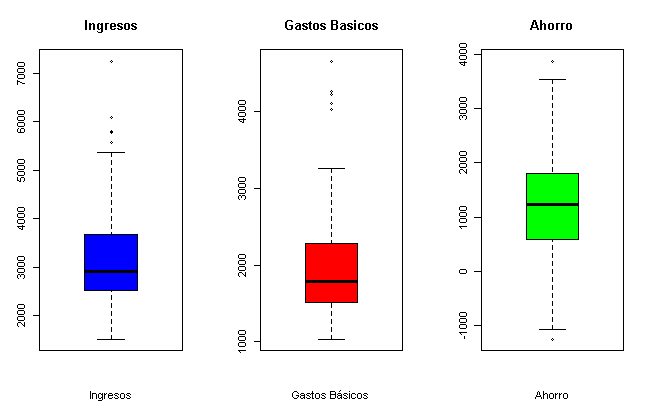


Figura 14: *Gráficos de boxplot* Fuente: RStudio

Se observa que tanto en los ingresos como en los gastos básicos existen valores mínimos y máximos muy extremos que requieren ser normalizados.

2.5. Eliminación de valores extremos y generación de la nueva correlación

En la eliminación de valores extremos o remoción de outliers, se identifican y eliminan valores alejados del rango “normal", ya que se consideran y podrían distorsionar el análisis (Dormido, 2024b).

Para eliminar valores extremos, se crea la función que los identifique y se aplica al dataframe. Seguidamente, se procede a eliminar las filas de estos valores; se verifica que hayan sido eliminados correctamente, cerciorándose de la integridad de los datos para el análisis.

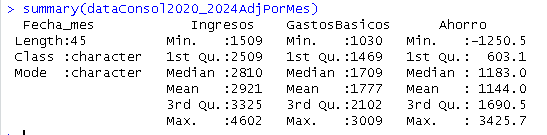


Figura 15: *Datos estadísticos por variables* Fuente: Fuente: RStudio

Los ingresos y gastos básicos se normalizan. La mediana y la media están más cercanas entre sí para las tres variables, por tanto están dentro de una curva normal.

Seguidamente, se verifica el grado de correlación entre las variables después de normalizar la data.

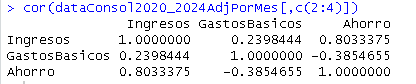


Figura 16: *Coeficiente de correlación entre variables normalizadas* Fuente: RStudio

En este caso, la correlación entre ahorro e ingresos se ha incrementado al 80.33% vs 70.90%, después de la normalización y en contrapartida, los gastos básicos se reducen de 55.24% al 23.98%, por lo que su grado de dispersión es mayor con respecto a los ingresos.

2.6. Etapa de modelado de regresión lineal

Con la finalidad de transformar los datos en herramientas predictivas, se construyen modelos matemáticos para representar los patrones de los datos (Dormido, 2024a)

2.6.1. Modelo de ahorro familiar en función de ingresos

Se aplica el modelo de regresión lineal (lm), que determina la relación entre la variable dependiente (ahorro) con la variable independiente (ingresos) (Villarino, 2024).

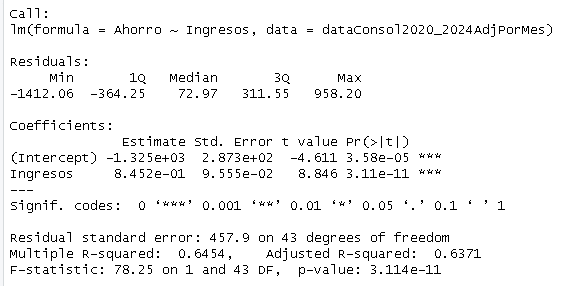


Figura 17: *Coeficientes entre ahorro e ingresos* Fuente: RStudio

La ecuación resultante: *Ahorro = -1325.06 + 0.8452 \* Ingresos*

Existe una relación positiva y significativa (p-value: 3.114e-11) entre los ingresos y el ahorro. El R-square (0.6454) indica que el 64.54 % de la variabilidad en el ahorro depende de la variabilidad en los ingresos, lo cual garantiza el modelado y la predicción de valores. El Residual standard error: 457.9 indica que por debajo de este valor el modelo predice con mayor precisión.

2.6.2. Modelo de gastos básicos familiar en función de ingresos.

Se aplica el modelo de regresión lineal (lm), que determina la relación entre la variable dependiente (gastos básicos) con la variable independiente (ingresos) (Dormido, 2024a).

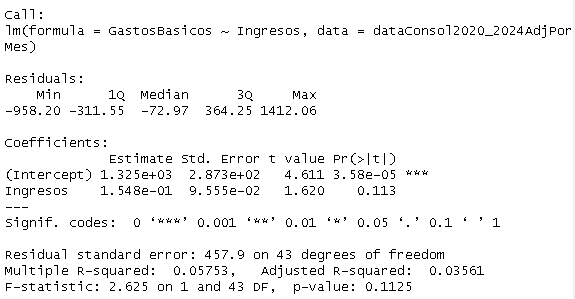


Figura 18:  *Coeficientes entre gastos básicos e ingresos* Fuente: Fuente: RStudio

Nuestra ecuación resultante: *GastosBasicos = 1325.06 + 0.1548 \* Ingresos*

No existe una relación significativa (p-value: 0.1125) entre los ingresos y los gastos. R-cuadrado: 0.05753: indica que solo 5.75% de la variabilidad de los gastos depende de la variabilidad en los ingresos, pero nos permite predecir con cierto grado de variabilidad. Residual estándar error: 457.9, indica que, para valores por debajo, el modelo puede predecir con mayor precisión.

2.6.3. Transformación del dataframe de salario neto por comunidad

Transformación del dataframe de salario neto por comunidad se realiza para usar los valores como entrenamiento en el modelo de regresión lineal de los puntos 2.6.1. y 2.6.2, y predecir nuevos valores. A continuación, se cambian los nombres de las comunidades de la variable "Comunidades y Ciudades Autónoma" para que, posteriormente, facilite la lectura de los gráficos. Se genera una nueva columna de salario neto familiar (factor: nº de miembros generadores de ingresos), que sería el resultado de multiplicar el salario neto por el número de miembros que generan ingresos en el grupo familiar.

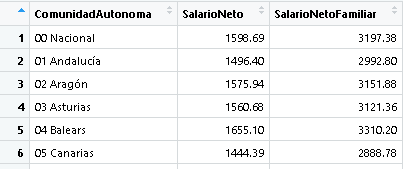


Figura 19: Salario neto promedio y salario neto promedio por grupo familiarFuente: Elaboración propia

2.6.4. Predicción de nuevos valores

En la predicción de nuevos valores, se utiliza un modelo estadístico con datos de entrenamiento para proyectar a futuro los valores. Se toma en cuenta como datos de entrenamiento la base de datos de salario neto promedio familiar por comunidad, ya que se trata de una data confiable, donde el salario neto promedio se considera como ingreso. El ingreso familiar es igual al salario neto familiar.

De los modelos desarrollados, se predicen el ahorro y los gastos básicos en función de los ingresos familiares. A partir de esta predicción, se crea un nuevo dataframe por comunidad autónoma y se incorporan las columnas de porcentaje de gastos y de ahorro, para visualizar las variaciones entre comunidades.

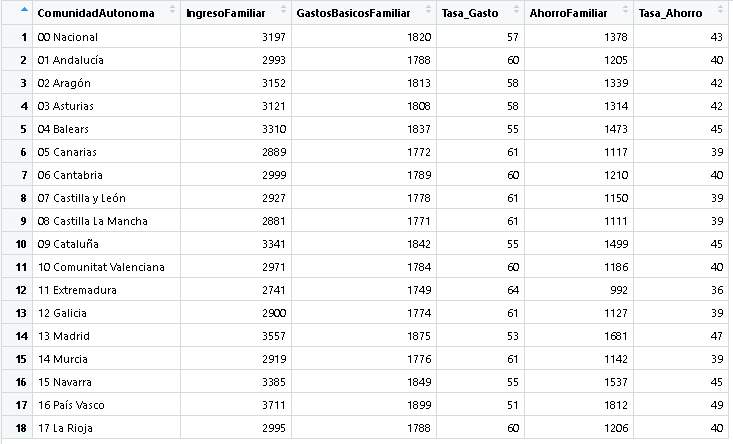


Figura 20:  *Ingresos, gastos básicos y ahorro por comunidades*  Fuente: Elaboración propia

El modelo predice con precisión los valores entrenados por comunidad.

Según la predicción, País Vasco genera el mayor ingreso neto promedio 3711€ y un ahorro 1812€, que representa el 49% de los ingresos. Con el 36 %, Extremadura genera el menor ingreso neto promedio 2741€ y un ahorro neto 992€. La media nacional, corresponde a un ingreso neto de 3197€ y un ahorro neto de 1378€, que corresponde al 43% de los ingresos.

Si se compara la media de ingresos y el ahorro del grupo familiar del estudio, se tiene una media de 2921€ y un ahorro de 1144€ (42%) , situándose a niveles de la comunidad de Castilla y León en ingresos y ahorro, sin embargo, ofrecen un margen de oportunidades en la medida que se incrementen los ingresos por miembro.

Desde el punto de vista del mercado financiero, los ahorros generados por comunidades aportan un incentivo adicional, para ofertas de productos financieros: inversiones de capital (planes de retiro, colocaciones) y préstamos bancarios (hipotecas, tarjetas de crédito, financiamiento) de acuerdo a la comunidad autónoma. Tal es el caso de País Vasco, Madrid, Navarra, Baleares y Cataluña, que generaron un ahorro sostenido con mejores oportunidades de inversión. También se pueden utilizar estos valores para otras estimaciones relacionadas con grupos familiares, tales como datos demográficos, análisis financieros (inversiones de capital, inversiones inmobiliarias, WACC (Coste Promedio de Capital y presupuestos).

2.6.5. Visualización de los resultados por comunidad autónoma.

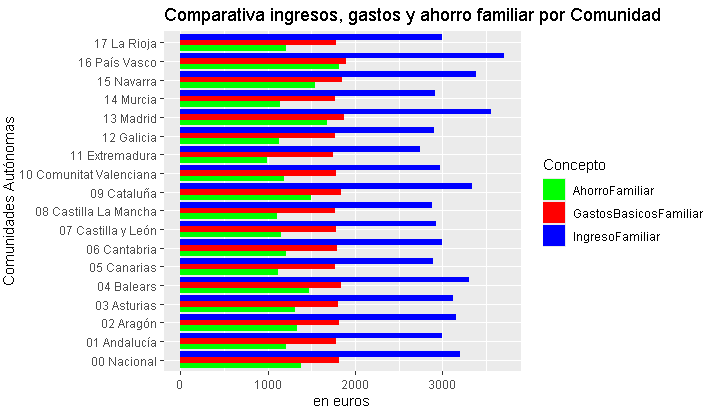


Figura 21: *Comparativa familiar por comunidad* Fuente: Elaboración propia

Los gastos básicos se comportan de forma muy similar para todas las comunidades, por su poca variabilidad en el modelo de regresión lineal.

El ahorro, por su parte, muestra variaciones en las diferentes comunidades en función de los ingresos.

Sólo en las comunidades de País Vasco, Madrid, Navarra y Balears podríamos conseguir un nivel de ahorro más alto, por tanto, tendremos mayores oportunidades para ofertar productos financieros o contrataciones.

2.7. Modelado de regresión no lineal (polinomial)

Debido a que los gastos básicos no guardan una variabilidad significativa respecto a los ingresos, se crea un modelo de regresión no lineal (Polinomial) aplicando la metodología del modelo anterior (regresión lineal) y utilizando los mismos valores de entrenamiento. Este permite modelar curvas y capturar relaciones (patrones) más complejas que el modelo lineal.

A continuación, se presenta el resultado del modelo polinomial en comparación con los resultados del modelo de regresión lineal, utilizando los mismos datos de entrenamiento por comunidad.



Figura 22: *Comparativa ingresos, gastos y ahorro por rangos y por comunidad* Fuente: Elaboración propia

Los gastos básicos (GtosRegresion) predichos en el modelo de regresión lineal se acercan más a el límite superior (GtosSuperior) de los gastos básicos predichos en el modelo de regresión polinomial para cada valor de ingresos entrenados por comunidades.

Por otro lado, el ahorro predicho (AhorroRegresion) en el modelo de regresión lineal se acerca al límite inferior(AhorroInferior) del ahorro predicho en el modelo polinomial.

En ambos casos, nuestros valores predichos se encuentran dentro de los valores inferiores y superiores del modelo polinomial sin que existan valores que salgan de esos rangos en cada una de las comunidades.

Como ejemplo tenemos que nivel Nacional tenemos un Ingreso promedio familiar de 3197€, con ahorro que oscila entre 1273€ y 1628$ y una fuerte tendencia hacia los 1378€. A nivel de gastos tenemos un gasto oscilante entre 1569€ y 1924€ con una fuerte tendencia hacia los 1820€.  
Nuevamente, País Vasco, Madrid, Navarra, Cataluña y Balears mantienen un ahorro por encima de la media nacional.

Se concluye que el modelo de regresión lineal predice de forma significativa el grado de precisión (64,54 %) el ahorro; además, los gastos básicos estimados del modelo no lineal aportan un rango de variabilidad adicional en el análisis por comunidades.

1. RESULTADOS

La relación positiva y significativa entre los ingresos y el ahorro y su posterior modelado, refleja un vínculo moderadamente alto entre ambas variables, lo que lo convierte en una herramienta confiable para anticipar comportamientos futuros.

El modelo de regresión lineal predice con precisión el ahorro con valores entrenados por comunidades y adicionalmente, el modelo no lineal polinomial proporciona una variabilidad de ahorros y gastos que complementa el modelo de regresión lineal.

Un asesor financiero podría analizar los ingresos actuales de sus clientes a través del modelo, y predeciría su capacidad de ahorro. Asimismo, en situaciones especiales (crecimiento del grupo familiar, jubilación, adquisición de inmueble…), el modelo le ayudaría a proyectar el ahorro futuro del cliente a partir del comportamiento de gastos e ingresos; en escenarios negativos, el asesor podría preparar al cliente para asegurar que su capacidad de ahorro no se vea completamente comprometida.

1. CONCLUSIONES

* Gracias a la correcta importación y transformación de la data, se organizan y limpian los datos, permitiendo que sean precisos y fiables. Así mismo, la eliminación de valores extremos y la consolidación de categorías por mes facilitan un análisis conciso, que proporciona una base sólida para el modelado posterior.
* A través del cálculo de métricas y la creación de visualizaciones, se identifican las relaciones entre ahorro, gastos e ingresos. Con la finalidad de apoyar la decisión de un modelo de regresión lineal, se normalizan los valores, que permiten facilitar la identificación de patrones y variaciones regionales .
* El modelo de regresión lineal adoptado demuestra su efectividad para identificar patrones de ahorro e ingresos, entrenados con datos de comunidades, confirmando su utilidad como herramienta para la planificación financiera y la toma de decisiones informadas sobre estrategias de ahorro y posibles inversiones.
* Se diseña el modelo de regresión no lineal (polinomial) que permite determinar la variabilidad de los gastos básicos y el ingreso en una relación no lineal, lo que aporta nuevos rangos de variabilidad de los gastos básicos por comunidad autonómica.
* Los modelos demuestran eficacia en la predicción de ahorro y gastos, lo cual es aplicable a una variedad de escenarios futuros, como inversión inmobiliaria, capacidad de generar ahorro, inversiones de capital o datos demográficos. Esta capacidad predictiva resulta especialmente útil en aplicaciones empresariales, donde puede optimizar la toma de decisiones financieras y estratégicas, mejorando la planificación.

1. FUENTES BIBLIOGRÁFICAS

Dormido Canto, R. (2024, enero). *Módulo 4: Introducción al Machine Learning con R. Parte II: Introducción a la estadística descriptiva con R*. Universidad Nacional de Educación a Distancia (UNED).

Dormido Canto, R. (2024, enero). *Módulo 4: Introducción al Machine Learning con R. Parte III: Aprendizaje con R*. Universidad Nacional de Educación a Distancia (UNED).

Espiga Fernández, F. (2022, junio). *Uso de machine learning para la toma de decisiones financieras*. Ciencia de datos. Net. <https://cienciadedatos.net/uso-de-machine-learning-para-la-toma-de-decisiones-financieras>

Google. (2024). *Google Gemini* (versión 1.0). [Herramienta de inteligencia artificial].<https://gemini.google.com>

Mendoza Vega, J.B. (n.d.). *R para principiantes*.<https://bookdown.org/jboscomendoza/r-principiantes4/>

Villarino, G. (2024, febrero). *Estadística parte II: Análisis de varianza y análisis de regresión*. Universidad Nacional de Educación a Distancia (UNED).

1. ANEXOS

Se ha incluido la codificación de cada etapa del proceso de extracción, transformación y modelado en el entorno de RStudio. La nomenclatura utilizada en Word es la misma que se ha utilizado en Rstudio.

#### # 2.1.Etapa de Extracción del dataset

#### # Instalar las librerías

>install.packages(“knitr”)

>library(knitr)

>install.packages(“rio”)

>library(rio)

>install.packages(“dplyr”)

>library(dplyr)

>install.packages(“tidyverse”)

>library(tidyverse)

>install.packages(“lubridate”)

>library(lubridate)

>install.packages(“e1071”)

>library(e1071)

>install.packages(“ggplot2”)

>library(ggplot2)

>library(patchwork)

>install.packages(“openxlsx”)

>library(openxlsx)

#### # Se importa el dataset “EdoCtaConsol2020-2024.xlsx” y se crear un dataframe de como: “EdoCtaConsol2020\_2024”

>EdoCtaConsol2020\_2024 <- import(“<https://raw.githubusercontent.com/Xabiertxu/TFMUned/master/EdoCtaConsol2020-2024.xlsx>”)

>attach(EdoCtaConsol2020\_2024)

>dim(EdoCtaConsol2020\_2024)

>head(EdoCtaConsol2020\_2024, n=3)

#### # Se importan los datos de salario neto por comunidad obtenidos del INE Base = 2022 y se crear el dataframe como: “SalarioNeto\_Comunidad”.

>SalarioNeto\_Comunidad <-import(“<https://raw.githubusercontent.com/Xabiertxu/TFMUned/master/Salario_mensual_Comunidad_2022.xlsx>”)

>attach(SalarioNeto\_Comunidad)

>dim(SalarioNeto\_Comunidad)

>head(SalarioNeto\_Comunidad, n=3)

#### # 2.2. Etapa de transformación del dataset.

#### # Se elimina la columna “concepto” para anonimizar el dataset

>EdoCtaConsol2020\_2024cl <-subset(EdoCtaConsol2020\_2024, select = -c(Concepto))

#### # Se cambia el signo negativo (-) de las partidas de gastos y egresos en la variable Importe y se crea una nueva variable importe\_abs.

>Importe\_abs <- abs(EdoCtaConsol2020\_2024cl$Importe)

EdoCtaConsol2020\_2024cl <- data.frame(EdoCtaConsol2020\_2024cl,Importe\_abs)

>View(EdoCtaConsol2020\_2024cl)

#### # Se seleccionan del dataframe las 5 variables más representativas.

>dataConsol2020\_2024 <- subset(EdoCtaConsol2020\_2024cl, select = c(“Fecha\_ctble”, “Fecha\_mes”, “Importe\_abs”, “Moneda”,“Categoria2”))

>head(dataConsol2020\_2024, n=5)

#### # Se transforman los elementos de la variable “Categoria2” en nuevas variables utilizando la función pivot\_wider

>dataConsol2020\_2024 <- pivot\_wider(dataConsol2020\_2024, names\_from = Categoria2, values\_from = Importe\_abs, values\_fn = sum)

>str(dataConsol2020\_2024) names(dataConsol2020\_2024)

#### # Se sustituyen los valores NA por ceros

>dataConsol2020\_2024 <- dataConsol2020\_2024 %>% mutate(across(where(is.numeric), ~replace(., is.na(.), 0)))

>View(dataConsol2020\_2024) str(dataConsol2020\_2024)

#### # Se convierten las variables Fecha a formato Date.

>dataConsol2020\_2024[,“Fecha\_ctble”] <- as.Date(dataConsol2020\_2024$Fecha\_ctble)

>dataConsol2020\_2024[,“Fecha\_mes”] <- as.Date(dataConsol2020\_2024$Fecha\_mes)

#### # Se convierte el formato yy-mm-dd a formato yy-mm de la variable Fecha\_mes.

>dataConsol2020\_2024[,“Fecha\_mes”] <- format(dataConsol2020\_2024$Fecha\_mes, “%Y-%m”)

>View(dataConsol2020\_2024) str(dataConsol2020\_2024)

#### # Se modifican los nombres de algunas variables recien creadas, que tienen espacios en blanco.

>names(dataConsol2020\_2024)[names(dataConsol2020\_2024) == “Alquiler Vivienda”] <- “AlquilerVivienda”

> names(dataConsol2020\_2024)[names(dataConsol2020\_2024) == “Consumo Personal”] <- “ConsumoPersonal” >names(dataConsol2020\_2024)[names(dataConsol2020\_2024) == “Credito Bancario”] <- “CreditoBancario”

>names(dataConsol2020\_2024)[names(dataConsol2020\_2024) == “Otros Egresos”] <- “OtrosEgresos”

>names(dataConsol2020\_2024)[names(dataConsol2020\_2024) == “Otros Ingresos”] <- “OtrosIngresos”

>names(dataConsol2020\_2024)[names(dataConsol2020\_2024) == “Servicios Vivienda”] <- “ServiciosVivienda”

>names(dataConsol2020\_2024)

#### # 2.3. Etapa de consolidación (totalización de datos).

#### # Se totalizan las variables por mes.

>dataConsol2020\_2024PorMes <- dataConsol2020\_2024 %>% group\_by(Fecha\_mes) %>% summarise(Ingresos = sum(Ingresos), OtrosIngresos = sum(OtrosIngresos), Inversiones = sum(Inversiones), AlquilerVivienda = sum(AlquilerVivienda), ServiciosVivienda = sum(ServiciosVivienda), Transporte = sum(Transporte), Impuestos = sum(Impuestos), Seguros = sum(Seguros), Educacion = sum(Educacion), Autonomo = sum(Autonomo), Alimentacion = sum(Alimentacion), ConsumoPersonal =sum(ConsumoPersonal), CreditoBancario = sum(CreditoBancario), OtrosEgresos = sum(OtrosEgresos), Donaciones =sum(Donaciones))

>View(dataConsol2020\_2024PorMes)

>dim(dataConsol2020\_2024PorMes)

#### # Se consolidan los ingresos y gastos.

>dataConsol2020\_2024TotalPorMes <- dataConsol2020\_2024PorMes %>% mutate( Ingresos = Ingresos + OtrosIngresos, GastosBasicos = ServiciosVivienda + Transporte + Seguros + Educacion + Alimentacion + ConsumoPersonal ) %>% select(Fecha\_mes, Ingresos, GastosBasicos)

>View(dataConsol2020\_2024TotalPorMes) dim(dataConsol2020\_2024TotalPorMes)

#### # Se incluye la variable ahorro y cálcular el ahorro mensual.

>dataConsol2020\_2024TotalPorMes <- dataConsol2020\_2024TotalPorMes %>% mutate( Ahorro = Ingresos - GastosBasicos) %>% select(Fecha\_mes, Ingresos, GastosBasicos, Ahorro)

>View(dataConsol2020\_2024TotalPorMes)

#### # 2.4. Etapa de análisis, métricas y gráficos.

#### # 2.4.1. Métricas

>summary(dataConsol2020\_2024TotalPorMes)

#### #2.4.2. Coeficiente de correlación.

>cor(dataConsol2020\_2024TotalPorMes[,c(2:4)])

#### # 2.4.3. Histogramas por totales: ingresos, gastos básicos y ahorro.

>library(“patchwork”)

>p1 <- ggplot(data = dataConsol2020\_2024TotalPorMes) + geom\_histogram( mapping = aes(x = Ingresos, fill = ..count..), color = “black” ) + scale\_fill\_gradient(low = “lightblue”, high = “darkblue”, name = “Frec”) + labs(x = “Ingresos”, y = “Frecuencia”) + theme\_minimal()

>p2 <- ggplot(data = dataConsol2020\_2024TotalPorMes) + geom\_histogram( mapping = aes(x = GastosBasicos, fill = ..count..), color = “black” ) + scale\_fill\_gradient(low = “yellow”, high = “red”, name = “Frec”) + labs(x = “Gastos Basicos”, y = “Conteo”) + theme\_minimal()

>p3 <- ggplot(data = dataConsol2020\_2024TotalPorMes) + geom\_histogram( mapping = aes(x = Ahorro, fill = ..count..), color = “black” ) + scale\_fill\_gradient(low = “lightgreen”, high = “Darkgreen”, name = “Frec”) + labs(x = “Ahorro”, y = “Conteo”) + theme\_minimal()

>combined\_plot <- p1 / p2 / p3

>combined\_plot

#### # 2.4.4. Visualización gráfica por pares de ingresos, gastos y ahorro

>pairs(dataConsol2020\_2024TotalPorMes[, c(2:4)])

#### # 2.4.5 Graficación de boxplot

>par(mfrow = c(1, 3)) boxplot(dataConsol2020\_2024TotalPorMes[,“Ingresos”], xlab=“Ingresos”, main = “Ingresos”, col=“blue”) boxplot(dataConsol2020\_2024TotalPorMes[,“GastosBasicos”], xlab=“Gastos Basicos”, main = “GastosBasicos”, col=“red”) boxplot(dataConsol2020\_2024TotalPorMes[,“Ahorro”], xlab=“Ahorro”, main = “Ahorro”, col=“green”)

#### # 2.5. Eliminación de valores extremos y generación de la nueva correlación

#### #Se crea una función de identificación de los valores extremos

>find\_outliers <- function(x) { Q1 <- quantile(x, 0.25) Q3 <- quantile(x, 0.75) IQR <- Q3 - Q1 lower\_bound <- Q1 - 1.5 \* IQR upper\_bound <- Q3 + 1.5 \* IQR return(which(x < lower\_bound | x > upper\_bound)) }

#### # 2.5.2. Aplicación de la función al datset

>outliers <- Reduce(union, lapply(dataConsol2020\_2024TotalPorMes[, c(“Ingresos”,“GastosBasicos”)], find\_outliers))

#### # 2.5.3. Eliminación las filas con valores extremos

>dataConsol2020\_2024AdjPorMes <- dataConsol2020\_2024TotalPorMes[-outliers, ]

#### # 2.5.4. Verificación de eliminación de valores extremos.

>summary(dataConsol2020\_2024AdjPorMes) View(dataConsol2020\_2024AdjPorMes)

.

#### # Se verifica nuevamente el grado de correlación entre las variables.

>cor(dataConsol2020\_2024AdjPorMes[,c(2:4)])

#### # 2.6. Etapa de modelado usando el dataframe normalizado

#### # 2.6.1. Modelo de ahorro familiar en función de ingresos

>Ahorro\_IngresosFamiliarRegress <- lm(Ahorro ~ Ingresos, data = dataConsol2020\_2024AdjPorMes) summary(Ahorro\_IngresosFamiliarRegress)

#### # 2.6.2. Modelo de gastos básicos familiar en función de ingresos

>GastosBasicos\_IngresosFamiliarRegress <- lm(GastosBasicos ~ Ingresos, data = dataConsol2020\_2024AdjPorMes) summary(GastosBasicos\_IngresosFamiliarRegress)

#### # 2.6.3. Transformación del dataframe de salario neto por comunidad, para usar los valores como entrenamiento en el modelo de regresión lineal de los puntos 2.6.1. y 2.6.2.

# A continuación, se cambian los nombres de las comunidades de la variable “Comunidades y Ciudades Autónoma” para que, posteriormente, facilite la lectura de los gráficos y el mapa. Se genera una nueva columna de salario neto familiar (factor: nº de miembros generadores de ingresos), que sería el resultado de multiplicar el salario neto por el número de miembros que generan ingresos en el grupo familiar.

>names(SalarioNeto\_Comunidad)[names(SalarioNeto\_Comunidad) == “Comunidades y Ciudades Autónomas”] <- “ComunidadAutonoma”

#### # Se incluye una nueva columna de salario neto familiar, usando como factor el número de miembros generadores de ingresos

>NroGeneradorIngresos <- 2 # 2 miembros generadores de ingresos

>SalarioNeto\_Comunidad <- SalarioNeto\_Comunidad %>% mutate( SalarioNetoFamiliar = SalarioNeto \* NroGeneradorIngresos ) %>% select(ComunidadAutonoma, SalarioNeto, SalarioNetoFamiliar )

>View(SalarioNeto\_Comunidad)

#### # Se modifican los nombres de la comunidades autónomas (CCAA)

>SalarioNeto\_Comunidad <- SalarioNeto\_Comunidad %>% mutate(ComunidadAutonoma = case\_when(ComunidadAutonoma == “03 Asturias, Principado de” ~ “03 Asturias”, ComunidadAutonoma == “04 Balears, Illes” ~ “04 Balears”, ComunidadAutonoma == “08 Castilla - La Mancha” ~ “08 Castilla La Mancha”, ComunidadAutonoma == “17 Rioja” ~ “17 La Rioja”, ComunidadAutonoma == “13 Madrid, Comunidad de” ~ “13 Madrid”, ComunidadAutonoma == “14 Murcia, Región de” ~ “14 Murcia”, ComunidadAutonoma == “15 Navarra, Comunidad Foral de” ~ “15 Navarra”, ComunidadAutonoma == “17 Rioja, La” ~ “17 La Rioja”, TRUE ~ ComunidadAutonoma ))

>View(SalarioNeto\_Comunidad)

#### # 2.6.4. Predicción de nuevos valores

>IngresoFamiliarEst <- data.frame(Ingresos = SalarioNeto\_Comunidad$SalarioNetoFamiliar)

>predict(Ahorro\_IngresosFamiliarRegress,IngresoFamiliarEst) AhorroFamiliarPred <- data.frame(predict(Ahorro\_IngresosFamiliarRegress,IngresoFamiliarEst))

>Ahorro\_IngresosFamiliarEstim <-data.frame(IngresoFamiliarEst,AhorroFamiliarPred)

>colnames(Ahorro\_IngresosFamiliarEstim)

>Ahorro\_IngresosFamiliarEstim <- Ahorro\_IngresosFamiliarEstim %>% rename(AhorroFamiliarEst = predict.Ahorro\_IngresosFamiliarRegress..IngresoFamiliarEst.)

>View(Ahorro\_IngresosFamiliarEstim)

#### # Se predicen los gastos básicos en función de los ingresos familiares por comunidad

Se iguala el salario neto familiar por comunidad al ingreso familiar usado.

>IngresoFamiliarEst <- data.frame(Ingresos = SalarioNeto\_Comunidad$SalarioNetoFamiliar)

>predict(GastosBasicos\_IngresosFamiliarRegress,IngresoFamiliarEst) >GastosBasicosFamiliarPred <- data.frame(predict(GastosBasicos\_IngresosFamiliarRegress,IngresoFamiliarEst))

>GastosBasicos\_IngresosFamiliarEstim <-data.frame(IngresoFamiliarEst,GastosBasicosFamiliarPred)

>colnames(GastosBasicos\_IngresosFamiliarEstim)

>GastosBasicos\_IngresosFamiliarEstim <- GastosBasicos\_IngresosFamiliarEstim %>% rename(GastosBasicosFamiliarEst = predict.GastosBasicos\_IngresosFamiliarRegress..IngresoFamiliarEst.)

>View(GastosBasicos\_IngresosFamiliarEstim)

#### # Se crea un dataframe con los datos predichos anteriormente, por comunidad autónoma\*

>ComunidadAutonoma <- SalarioNeto\_Comunidad$ComunidadAutonoma

>dataConsolEstimFamiliarPorComunidad <- data.frame(ComunidadAutonoma, IngresoFamiliarEst, GastosBasicosFamiliar =GastosBasicos\_IngresosFamiliarEstim[,“GastosBasicosFamiliarEst”] , AhorroFamiliar = Ahorro\_IngresosFamiliarEstim[,“AhorroFamiliarEst”])

>dataConsolEstimFamiliarPorComunidad <- dataConsolEstimFamiliarPorComunidad %>% rename(IngresoFamiliar = Ingresos)

#### # Se incorporan nuevas columnas de porcentaje de gastos y de ahorro a la tabla.

>dataConsolEstimFamiliarPorComunidad <- dataConsolEstimFamiliarPorComunidad %>% mutate( Tasa\_GastoFam = GastosBasicosFamiliar / IngresoFamiliar \* 100, Tasa\_AhorroFam = AhorroFamiliar / IngresoFamiliar \* 100 ) %>% select(ComunidadAutonoma, IngresoFamiliar, GastosBasicosFamiliar, Tasa\_GastoFam, AhorroFamiliar, Tasa\_AhorroFam)

>dataConsolEstimFamiliarPorComunidad <- dataConsolEstimFamiliarPorComunidad %>% mutate(across(2:6, ~abs(round(.x, 0))))

>View(dataConsolEstimFamiliarPorComunidad)

>summary(dataConsolEstimFamiliarPorComunidad[,c(2:6)])

#### # 2.6.5. Visualizar los resultados por comunidad autónoma.

>data\_familiar <- dataConsolEstimFamiliarPorComunidad %>% pivot\_longer(cols = c(IngresoFamiliar, GastosBasicosFamiliar, AhorroFamiliar), names\_to = “variable”, values\_to = “valor”)

>ggplot(data = data\_familiar, aes(x = valor, y = ComunidadAutonoma, fill = variable)) + geom\_bar(stat = “identity”, position = “dodge”) + labs(x = “en euros”, y = “Comunidades Autónomas”, fill = “Concepto”) + scale\_fill\_manual(values = c(IngresoFamiliar = “blue”, GastosBasicosFamiliar = “red”, AhorroFamiliar = “green”)) + ggtitle(“Comparativa ingresos, gastos y ahorro familiar por Comunidad”)

#### # 2.7. Modelo de regresión no lineal polinomial

#### #Se ajusta el modelo con los ingresos y ahorro por grupo familiar

>modelo <- lm(Ahorro ~ Ingresos + I(Ingresos^2), data = dataConsol2020\_2024AdjPorMes)

#### # Se predice nuevos datos a partir de los datos de entrenamiento de salario neto familiar por comunidad

>nuevos\_datos <- data.frame(Ingresos = SalarioNeto\_Comunidad$SalarioNetoFamiliar) >predicciones <- predict(modelo, newdata = nuevos\_datos, interval = “confidence”)

#### # Se crea un nuevo dataframe con los resultados predichos

>datosConsolEstimAhorroIngresosPolinom <- data.frame( Ingresos = nuevos\_datos$Ingresos, Ahorro\_Predichos = predicciones, Inferior = predicciones[, 2], Superior = predicciones[, 3] )

>View(datosConsolEstimAhorroIngresosPolinom)

#### # Se ajusta el modelo con los ingresos y gastos básicos por grupo familiar

>modelo <- lm(GastosBasicos ~ Ingresos + I(Ingresos^2), data = dataConsol2020\_2024AdjPorMes)

#### # Se predice nuevos datos a partir de los datos de entrenamiento de salario neto familiar por comunidad

>nuevos\_datos2 <- data.frame(Ingresos = SalarioNeto\_Comunidad$SalarioNetoFamiliar) predicciones <- predict(modelo, newdata = nuevos\_datos2, interval = “confidence”)

#### # se crea un nuevo dataframe con los resultados predichos

>datosConsolEstimGastosBasicosIngresosPolinom <- data.frame( Ingresos = nuevos\_datos$Ingresos, GastosBasicos\_Predichos = predicciones, Inferior= predicciones[, 2], Superior= predicciones[, 3] )

>View(datosConsolEstimGastosBasicosIngresosPolinom)

#### # Se comparan los dos modelos en la tabla resultante

>ComunidadAutonoma <- SalarioNeto\_Comunidad$ComunidadAutonoma

>dataConsolEstimFamiliarPorComunidad <- data.frame(ComunidadAutonoma, Ingresos = round(datosConsolEstimAhorroIngresosPolinom[,“Ingresos”],0), GtosInferior =round(datosConsolEstimGastosBasicosIngresosPolinom[,“Inferior”],0), GtosRegresion = round(dataConsolEstimFamiliarPorComunidad[,“GastosBasicosFamiliar”],0), GtosSuperior = round(datosConsolEstimGastosBasicosIngresosPolinom[,“Superior”],0), AhorroInferior = round(datosConsolEstimAhorroIngresosPolinom[,“Inferior”],0), AhorroRegresion = round(dataConsolEstimFamiliarPorComunidad[,“AhorroFamiliar”],0), AhorroSuperior = round(datosConsolEstimAhorroIngresosPolinom[,“Superior”],0) )

>View(dataConsolEstimFamiliarPorComunidad)

#### # Se exporta el dataframe a excel

>write.xlsx(dataConsolEstimFamiliarPorComunidad, file = “C:\Users\Administrador\Documents\Big Data y Business Analytics UNED 2023-2024\TFM\dataConsolEstimFamiliarPorComunidad.xlsx”)