**Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional**
Andrés Mendez Vásquez, Eduardo Arturo Rodríguez Tello
Departamento de Ingenieria Eléctrica y Computación
Responsable: Por asignar

# Encuesta Nacional de Vivienda 2020

*Avance de proyecto*

Lunes 1 de Mayo, 2023
M.C Jesús Iván Ruíz Martínez - ID: CIE-601028-1U2

# Contents



# List of Figures

# 1 Introducción

La Encuesta Nacional de Vivienda 2020 (ENV) del Instituto Nacional de Estadística y Geografía (INEGI) es una fuente confiable de información sobre las condiciones de vida de la población mexicana. Los datos recopilados en esta encuesta ofrecen una mirada detallada a los hogares y las comunidades del país, lo que permite a los investigadores analizar las tendencias demográficas, económicas y sociales que afectan a las familias mexicanas [7].

En este proyecto, se utilizarán los datos de la ENV2020 para explorar y analizar varios aspectos de la vivienda y las condiciones de vida de la población mexicana. En donde se examinaran detalles como el nivel de estudio, e ingresos por zona geográfica con lo que se buscará identificar desigualdades económicas que puedan existir en cuestión de vivienda presentes en México.

Para determinar ello se harán uso de técnicas de machine learning (ML) para llevar al cabo una clasificación de las clases presentes en la ENV.

## 1.1 ¿Que es el machine learning?

El Machine Learning es una rama de la inteligencia artificial que se enfoca en desarrollar algoritmos que pueden aprender de los datos y hacer predicciones o tomar decisiones sin ser programadas explícitamente [1].

Para este proyecto el Machine Learning es útil para identificar patrones o tendencias en los datos cuya detección resulta complicada de manera manual por la cantidad de datos que se incluyen [6]. También puede ser usado para crear modelos predictivos que permitan producir eventos futuros basados en datos históricos, como predecir la demanda de vivienda en las diferentes zonas geográficas de México en los próximos años.

# 2 Encuesta Nacional de Vivienda 2020

La razón por la cual fue llevada esta encuesta por el INEGI tiene como objetivo producir información estadística sobre las características de la vivienda en México que permita generar un panorama amplio sobre la situación de la vivienda en el país, necesidades y demanda de la población al respecto.

Esta fue realizada en 2021-08-23 y cuenta con los siguientes apartados:

- Demanda y necesidades de vivienda
- Características del hogar
- Características de los residentes del hogar
- Segunda vivienda 1
- Segunda vivienda 2

- Gastos en las viviendas secundarias
- Características de las viviendas

Las cuales brindan encuestas con diferentes preguntas enfocadas al área en específico, las cuales se pueden visualizar en la figura 1.


Figure 1: Imagen mostrando características principales de la encuesta de vivienda 2020.

Con el fin de este proyecto, se utilizará en mayor medida la encuesta denominada ” Características de los residentes del hogar” por que brinda una mayor información socio económica acerca de los ingresos familiares por entidad geográfica. Donde se buscará clasificar cuanto influye la entidad geográfica asi como otras tablas en una clasificación de clase social.

# 3 Objetivos

Este proyecto tiene como objetivo contribuir al conocimiento sobre la situación de la vivienda y las condiciones de vida en México, proporcionando información acerca de como afectan distintas variables a lo que es una clase social basado en los ingresos familiares. Esto usando herramientas como ML mediantes distintos clasificadores a través de un feature engineering.

## 3.1 Objetivos específicos

- Adquisición de base de datos abiertos ENV2020 del INEGI
- Montar base de datos en Mariadb
- Normalización de los datos
- Implementación de los distintos clasificadores
- Precisión y estimación de una media global
- Reporte de resultados

# 4 Avance de objetivos

En esta sección se describe lo que se ha realizado respecto a los objetivos mencionados anteriormente.

## 4.1 Base de datos

La base de datos fueron obtenidas mediante la página del INEGI en la sección de datos abiertos. Estos datos se encontraron en formato ”coma separated values” (csv) a través del enlace ENVI2020. Como se mencionó en la sección 2 de estos datos se extrajeron el conjunto de datos perteneciente a las características de residentes del hogar, la cual cuenta con n columnas y 202596 filas.

Se muestra en la siguiente tabla 2, las preguntas realizadas por los encuestadores que fueron tomadas en cuenta para los distintos modelos.

| Clave | Preguntas | Tipo de respuesta |
|---|---|---|
| ENT | Localidad perteneciente | 1,2,...,32 |
| SEXO | Sexo de la persona dueña o encargada de la vivienda? | 1, 2 |
| EDAD | Edad del dueño o encargado de la vivienda? | 1, 2 , .. , 99 |
| P2_5 | Habla alguna lengua indígena? | 1, 2, 9 |
| P2_8 | Actualmente vive en (estado civil) | 1, 2, .., 6 |
| P3_1 | La persona encargada, trabaja, estudia, es pensionado, etc... | 1, 2, ..., 8 |
| P3_3 | El trabajo de la semana pasada fue trabajador, empleado, empleador, jornalero... | 1, 2, ..., 5 |
| P3_4 | Cuanto gana por su actividad? | 00000, ..., 99999 |

Figure 2: Tabla de preguntas realizada para la encuesta ENVI2020 con su respectivo rango y clave.

## 4.2 Montura de base de datos

Para crear la base de datos, se optó por montar un servidor de Mariadb en un contenedor de Docker, y para la inserción de la base de datos se utilizó MySQL Workbrench conectado a nuestro servidor Mariadb para añadir los datos del archivo csv.

Como MySQL Workbrench presenta problemas cuando hay celdas vacías primero se rellenaron con un valor de 0 antes de importar a Mariadb. Un problema con esta base de datos, es que se basó principalmente en una de las preguntas con titulo de columna ”P3.4” la cual consiste en el ingreso que percibe la cabeza de familia de dicha vivienda, esta pregunta no fue contestada por casi dos terceras partes de los encuestados.

Se muestra tambien el tipo de encuesta realizada para cada columna en la siguiente figura. Con lo cual decidimos remover las filas que tuviesen esta celda vacia, para ello usamos un query en python que removiera estas filas, el query fue el siguiente:

```
update_query = "UPDATE conjunto_de_datos_tsdem_envi_2020 SET
P3_4 = NULL WHERE P3_4 = 0 OR P3_4 = 99888 OR P3_4 = 99999"
cur.execute(update_query)
```

Con ello se mantuvieron los datos de interés que abordaban el ingreso de la unidad familiar.

## 4.3 Variable de predicción

Cabe aclarar en esta sección que la forma en la que se va a hacer la predicción se basa en los ingresos que percibe el dueño del hogar, esto marcado en la columna "P3.4". Sin embargo, para poder llevar al cabo un sistema de clasificación se asignaron 3 clases correspondientes a la clase social baja, media y alta. Esto se muestra mejor en la figura 3 que se usó como base para los límites de dichas clases, siendo de $1 - 13000$ para la clase baja, $13000 - 77000$ clase media y lo restante clase alta.


Figure 3: Diagrama del sistema de clases según el INEGI

Para el sistema de clasificación se utilizó una funcion en específico que engloba los datos de la tabla P3.4 y los devuelve en 3 clases, la cual se lista a continuación.

```
def get_class_label(value):
    if value <= 13000:
        return 0  # Clase baja
    elif 13000 < value <= 77000:
        return 1  # Clase media
    else:
        return 2  # Clase alta
```

Con ello buscaremos ver cuales variables influyen más al momento de querer pertenecer a una clase más alta.

## 4.4 Feature engineering

Este paso consiste en dividir la base de datos en n partes, las cuales tendrán como motivo brindar las medias de predicción entre los distintos modelos de clasificación que se utilizarán, los cuales se listan a continuación:

1. Regresión lineal
2. Regresión logística
3. Modelo mixto - Gaussiana
4. Perceptron Multicapa (MLP)
5. Arból de decisión

Estos distintos modelos brindaran distintas medias $\mu_1, \mu_2, ..., \mu_5$ en las cuales se realizará una media promedio $\mu_k$ que será similar a la media de de todo el dataset $\mu_G \approx \mu_k$ donde $\mu_G$ es la media predicha de la población total.

# 5 Procedimiento

En este apartado se describe la forma que va a ir tomando el presente proyecto, como ya se mencionó anteriormente se realizara un ada-boosting en torno a diferentes clasificadores para observar la diferencia en el error entre los distintos modelos, se puede apreciar el esquema en la figura 4.

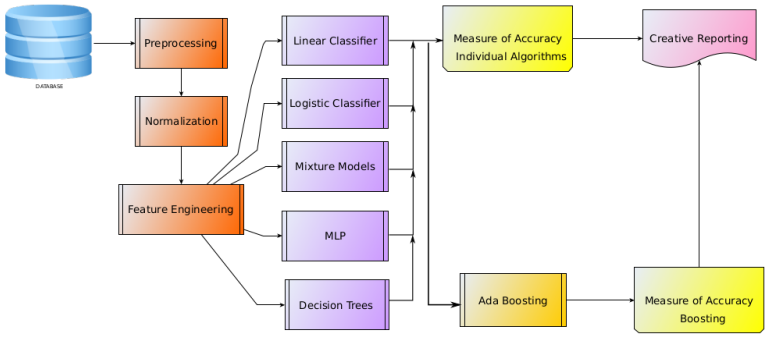



Figure 4: Pipeline del proyecto tomado del archivo de avance de proyecto 2 .

Posteriormente se describen los valores predichos para cada uno de estos clasificadores y en el apartado del apéndice se describe los detalles generales de cada método. Además se anexa a este mismo documentos los códigos en python usados para obtener estos resultados.

## 5.1 Regresión lineal

La regresión lineal es un modelo simple en el cual se ajusta los valores a una recta, en este modelo se usó un número de interacciones (batches) variable de 30000 y un learning rate de 0.001. Siendo este el valor más alto alcanzado de precisión, esto se visualiza de mejor forma en la figura 5

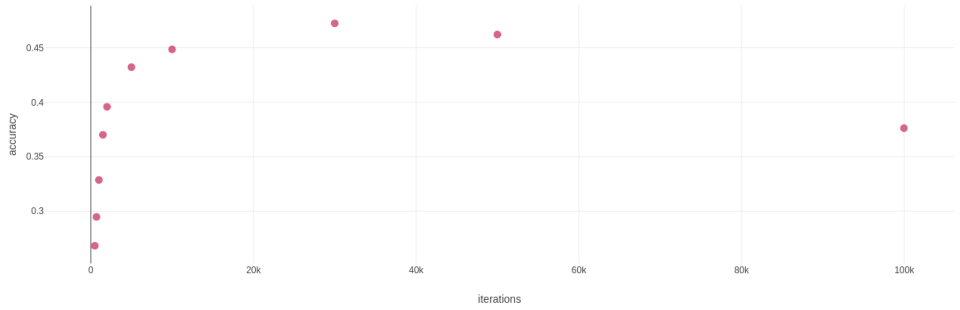



Figure 5: Gráfica del valor cuadrado medio respecto al número de iteracciones.

En el cual apreciamos un rápido acercamiento a la unidad después de cien mil interacciones. También en los resultados de clasificación que se muestra en la figura 6 elucidan la influencia en la Edad y a que labor se encuentra la persona actualmente, al sexo no le da mucha relevancia, el modelo a punta a que la columna referente a si se habla una lengua indígena esta influye negativamente.

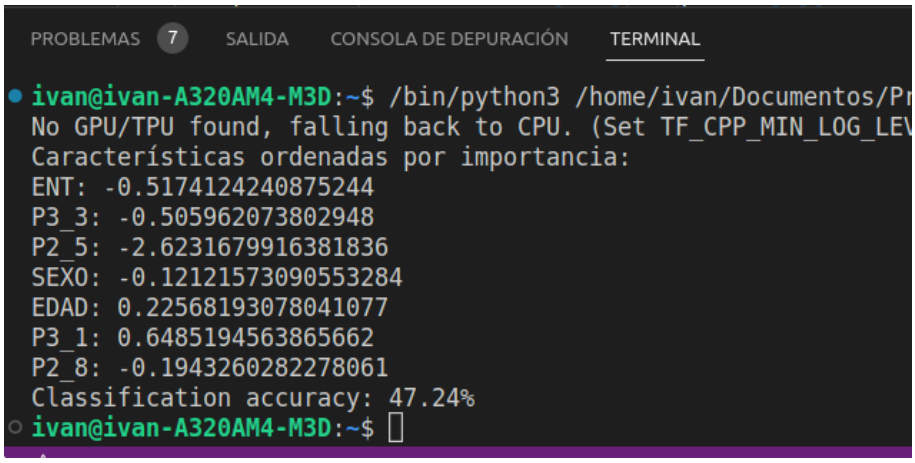



Figure 6: Resultado de la clasificación del modelo de regresión lineal.

## 5.2 Multilayer Perceptron

En este modelo se utilizaron diferentes capas, la estructura del modelo MLP es la siguiente:

- Capa de entrada: 7 unidades (correspondientes a las 7 features)
- Primera capa oculta: 64 unidades (w1: 32x64, b1: 64)
- Segunda capa oculta: 32 unidades (w2: 64x32, b2: 32)
- Capa de salida: 1 unidad (w3: 32x1, b3: 1)

Este modelo tiene una tendencia de elevarse a infinito ante un learning rate muy alto, por lo que se ajusto a un valor de $lr = 0.001$ con cien interacciones. Se describe en la figura 7 los valores de precisión obtenidos durante el entrenamiento a varias interacciones.

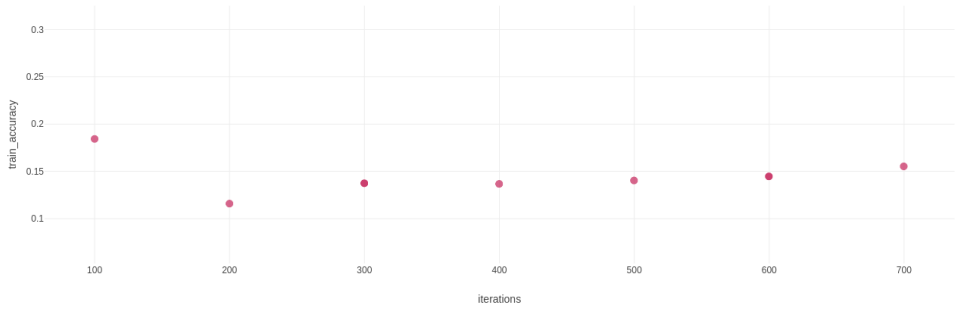



Figure 7: Gráfica la precisión respecto al número de iteracciones.

Se aprecia un error muy alto en la primer interacción que decae casi inmediatamente después de las primeras interacciones. Los resultados obtenidos para este modelo se observan en la figura 8

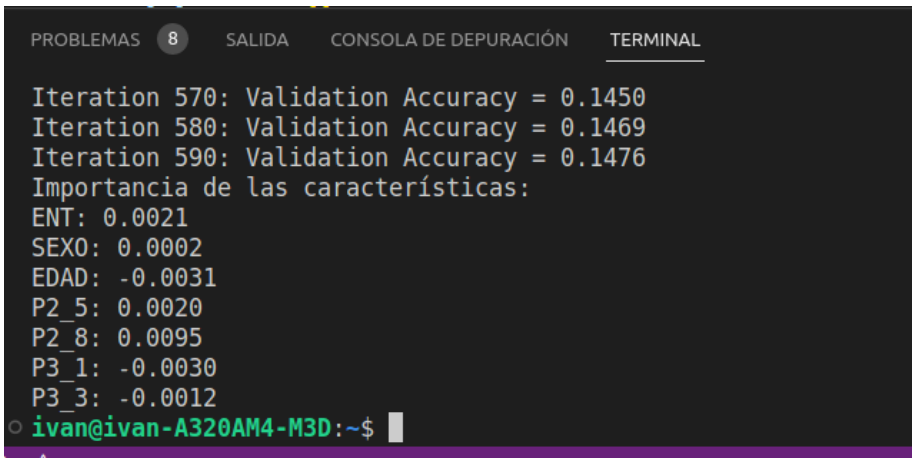



Figure 8: Imagen de los resultados obtenidos como clasificación del MLP.

## 5.3 Regresión logística

En este modelo se hace uso de la regresión logística para la clasificaciíon de variables en lo que son las 3 clases. Este modelo mostró un 90.79% de precisión con 1000 interacciones y un learning rate de 0.01. Se grafico la relación entre el MSE de entrenamiento y el MSE de validación el cual se muestra en la figura 9.

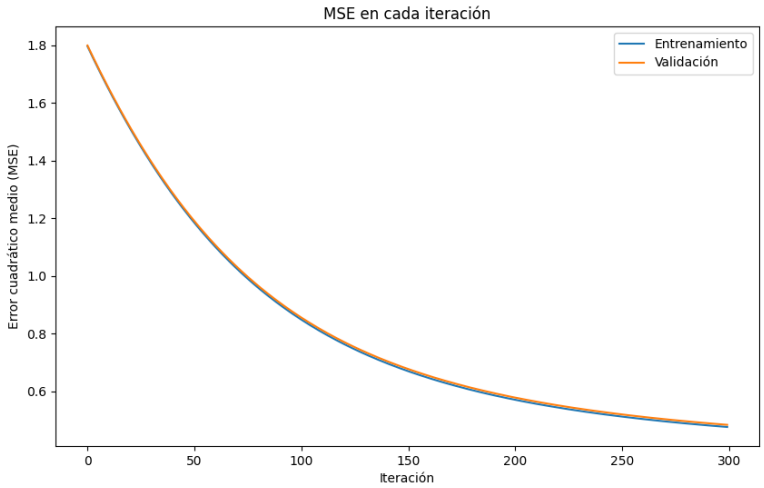



Figure 9: Gráfica del valor cuadrado medio respecto al número de interacciones.

También de los resultados de predicción se muestra lo arrojado por la terminal en la figura 10, en este se muestra como influye las variables de entrada siendo la P2.5 una variable que aleja de ser de una clase más alta y como la Edad, el sexo y p3.1 juegan un papel importante para aumentar la posibilidad de ser de una clase más alta.

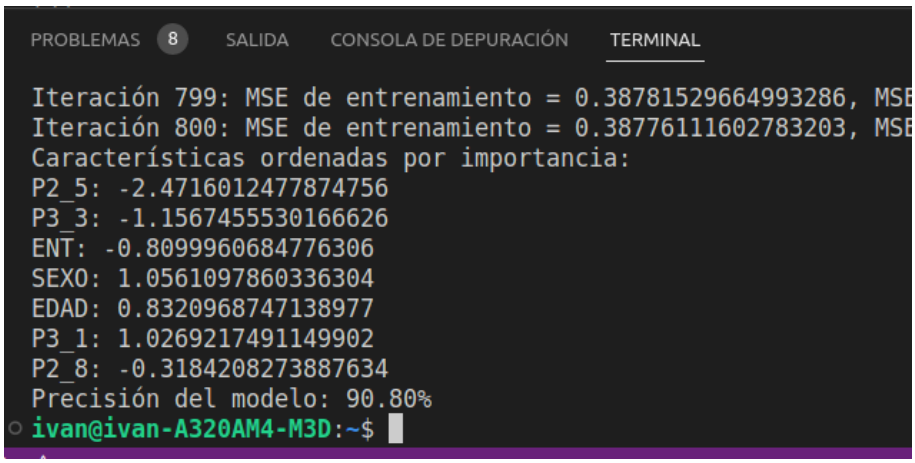



Figure 10: Imagen del resultado obtenido por terminal del modelo de regresión logística.

## 5.4 Mezcla Gaussiana

En este modelo se buscaron diferentes número de componentes que ayudaran a mejorar la precisión del clasificador, se obutvo una precisión de 62.2% con un número de componentes igual a dos, lo que denota que el problema tiende a no ser complejo, llegando a un 100% con un solo componente. Para dos componentes se visualiza en la siguiente figura 11 la nube con los dos componentes principales.

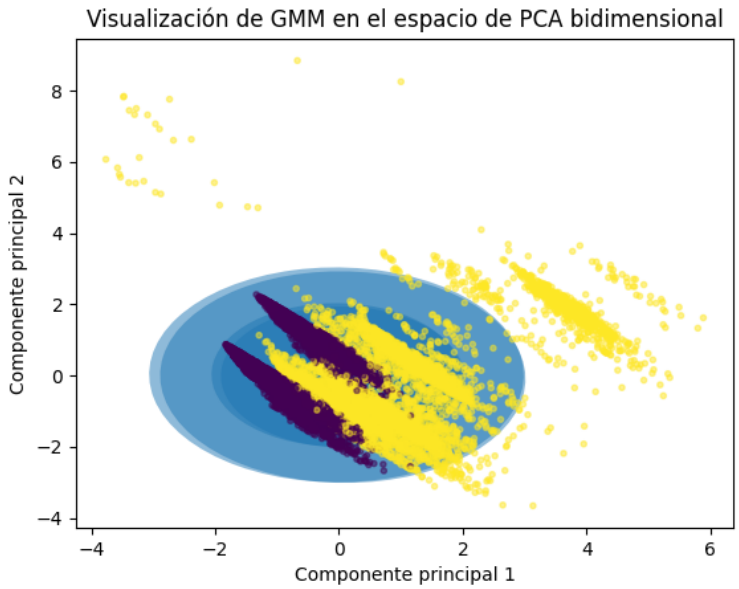



Figure 11: Gráfica del GMM obtenido para los dos componentes principales.

Los resultados obtenidos por el GMM se visualizan en la figura 12. Se observa que indica que las características principales que influyen para pertenceser a una clase social más alta son la localidad, el sexo y la edad, mientras los que menos influyen son referentes a que tipo de trabajo realizan o si son jubilados, trabajadores o pensionados.

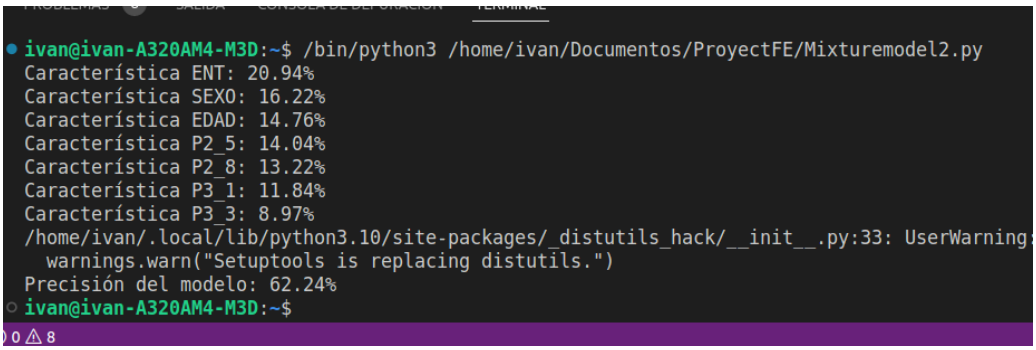



Figure 12: Imagen del resultado obtenido por terminal el GMM.

## 5.5 Arbol de decisión

En el arbol de decisión se estudiaron diferentes profunidades límites del modelo, se encontró que la profundidad máxima que generaba una mejor presición del modelo era de siete, la cual generaba una presición de 12.92% esto se puede apreciar mejor en la figura 13, los resultados de las características dadas se muestran en la figura 14.

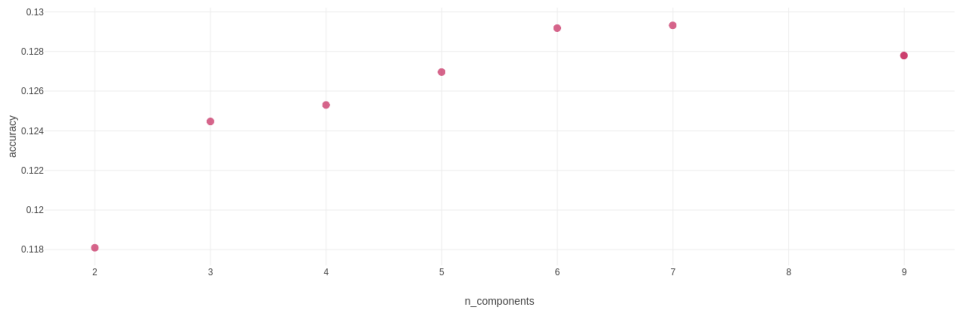



Figure 13: Gráfica de precisión versus profundidad máxima.

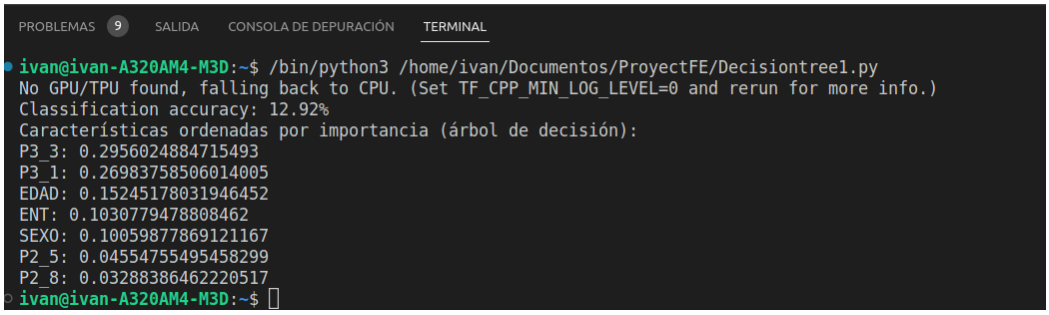



Figure 14: Imagen del resultado obtenido por terminal del modelo de árbol de decisión.

Se puede apreciar que los valores que más influyen son las columnas referente al tipo de puesto que desempeñan y a si se encuentran trabajando, siguendole la edad y la localidad, por otro lado el valor más bajo fue el relacionado a si la persona habla una lengua indígena.

# 6 Conclusiones

En base a los anteriores modelos se puede detallar varias cuestiones, entre ellas que un método sirve mejor para resolver unos problemas que otro, de estos métodos los que consiguieron mejor estimación y rápído fueron la regresión logística y el GMM, tuvieron buenos resultados de precisión y no llevo mucho optimizarlos, y su predicción pareció estar en concordancia, la regresión logística no dio malos resultados, sin embargo su precisión mejoraba lentamente y después de muchas interacciones su resultado mejoro, pero tardo un estimado de 20 min para el calculo. El mlp arrojo resultados de cierta forma confusos por lo que fue el que tuvo peor desempeño en esta dinámica en cuestión.

Posiblemnete sea posible mejorar la precisión en algunos de estos modelos, como agregar capas ocultas al MLP, usar algoritmos de random forest para mejorar el decisión tree, o bien usar el ada boosting para mejorar el performance entre algorítmos.

Finalmente se puede concluir que tres parámetros en específico destacan a primera vista en orden de conseguir pertenecer a una clase social más alta, estos fueron:

- Que la persona se encuentre trabajando (P3.1)
- La edad de la persona influye en gran medida
- La localidad o estado en el que te encuentres

Un dato que sale a relucir en casi todos los modelos, es la pregunta ”p2.5” donde se pregunta si se habla una lengua indígena. Todos los modelos apuntaron a que esto daba un valor negativo o bien una influencia marcable; es decir, influye de manera negativa el hablar una lengua indigena.

# A Apéndice

En esta sección se incluiran todos los datos técnicos, asi como parte de las ecuaciones y sketch de código en cada sub-elemento que componen el feature selection.

### A.0.1 Regresión lineal

La regresión lineal es un modelo estadístico que modela la relación entre una variable dependiente entre una o más variables independientes, en este trabajo como variable dependiente u objetivo es la relación de los salarios con forme la localidad. El objetivo de la regresión lineal es encontrar la mejor linea recta que describa la relación entre estas variables, de forma que minimice la diferencia entre las predicciones del modelo y los valores reales observados [8]. La función lineal tiene la siguiente forma

$$y = mx + b \tag{1}$$

Donde:

- y es la variable dependiente
- x es la variable independiente
- m es la pendiente de la linea
- b es el punto de intersección con el eje y

Con varias variables independientes la ecuación se transforma a la siguiente:

$$y = b + m_1 x_1 + m_2 x_2 + ... + m_n x_n \tag{2}$$

Donde $m_1, m_2, ...$ son los coeficientes de las variables independientes $x_1, x_2, ...$ [4].

### A.0.2 Regresión logística

El modelo de regresión logística utiliza una función logística para transformar la salida del modelo en una probabilidad, que siempre está entre 0 y 1. La función logística se utiliza para ajustar los parámetros del modelo de tal manera que la probabilidad estimada se ajuste lo mejor posible a los datos de entrenamiento.

Este regresión logística se basa principalmente en el uso de la función sigmoide para modelar la relación entre las variables de entrada y la probabilidad de salida. Esta se define como:

$$f(z) = \frac{1}{1 + \exp(-z)} \tag{3}$$

Donde z es una combinación lineal de los valores de entrada ponderados por los coeficientes del modelo, como:

$$z = b_0 + b_1 x_1 + b_2 x_z + ... + b_n x_n \tag{4}$$

Donde $b_0, b_1, ...$ son los coeficientes del modelo y $x_1, x_2, ...$ son los valores de entrada. La función sigmoide convierte los valores de entrada ponderados en una probabilidad que siempre está entre 0 y 1. Si la probabilidad estimada es mayor que 0.5, el modelo predice que el evento pertenece a la categoría positiva (1), y si es menor que 0.5, el modelo predice que el evento pertenece a la categoría negativa (0). Esto en el caso de clasificación binario.

### A.0.3 Modelo mixto gaussiano

El modelo de mezcla Gaussiana (GMM) es una técnica de aprendizaje que se usa para encotrar clusteres en un conjunto de datos. Este modelo se basa en la idea de que los datos pueden ser generados por la combinación de varias distribuciones Gaussianas.

El modelo de mezcla Gaussiana se ajusta a los datos utilizando el algoritmo de maximización de la esperanza (EM), que es un algoritmo iterativo que alterna entre la estimación de los parámetros del modelo basados en los datos y la asignación de puntos de datos a clústeres es basada en los parámetros del modelo.

Una distribuición gaussiana se representa como:

y su likelihood se representa como:

$$p\left(x \mid \mu, \sigma^2\right)=\mathcal{N}\left(\mu, \sigma^2\right)=\frac{1}{\sqrt{2 \pi \sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2 \sigma^2}\right).$$

$$p(\mathcal{X}|\mu)=\prod_{n=1}^{N} \frac{1}{\sqrt{2 \pi \sigma^2}} \exp -\frac{\left(x_n-\mu\right)^2}{2 \sigma^2},$$

$$\mathcal{L}=\log p(\mathcal{X}|\mu)=\sum_{n=1}^{N}\left[\log\left(\frac{1}{\sqrt{2 \pi \sigma^2}}\right)-\frac{\left(x_n-\mu\right)^2}{2 \sigma^2}\right],$$

$$\frac{d \mathcal{L}}{d \mu}=\sum_{n=1}^{N} \frac{x_n-\mu}{\sigma^2}.$$

### A.0.4 MLP

Las redes neuronales multicapa(MLP) son un tipo de modelo de aprendizaje profundo que consta de multiples capas de neuronas interconectadas en un patrón jerarquico. Estas constan de al menos 3 capas, en las cuales una es de entrada, una o más capas ocultas y una última capa de salida.

En esta cada neurona en una capa está conectada a todas las neuronas de la capa anterior y la capa siguiente mediante conexiones ponderadas. Durante el proceso de entrenamiento, estas ponderaciones se ajustan para minimizar el error entre las predicciones del modelo y los valores observados [2]. El proceso de cálculo se lleva principalmente en dos fases:

- Propagación hacia adelante(forward): En esta los valores de entrada se van modificando desde la capa de entrada hasta la capa de salida. En cada neurona se realiza una suma ponderada y se aplica una función de activación, que puede ser ReLU o sigmoidal. El resultado de la capa de salida es la predicción del modelo.

- Propagación hacia atrás(backward) : En este paso se calcula el error entre la predicción y el valor real observado. Posteriormente este error se propaga hacia atrás desde la capa de salida hasta la capa de entrada, ajustando los pesos o las ponderaciones en cada conexión con el fin de minimizar el error generado. Esto se repite varias veces hasta que el modelo converga [5].

El algoritmo de optimización, como el descenso de gradiente estocástico (SGD) o Adam, se utiliza para actualizar las ponderaciones de la red durante la retropropagación. También se pueden aplicar técnicas de regularización, como la parada temprana (early stopping) o la regularización L1/L2, para evitar el sobreajuste y mejorar la capacidad de generalización del modelo.

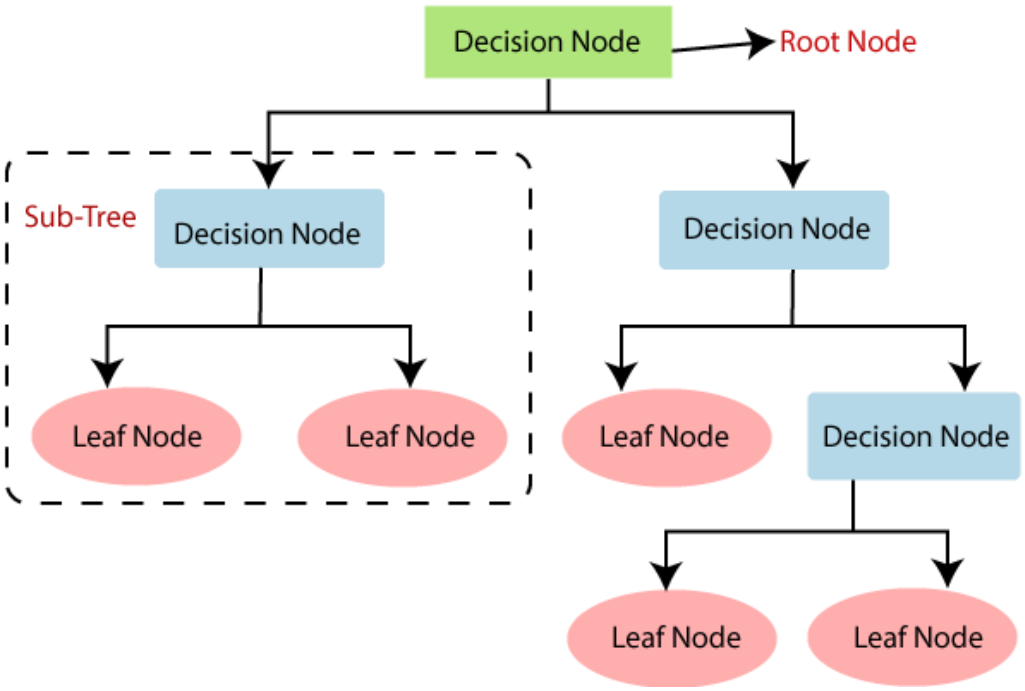
El proceso de construcción de un árbol de decisión comienza con el nodo raíz que representa a todos los datos disponibles. A continuación, se selecciona la característica que mejor divide

los datos en dos grupos más homogéneos en función de algún criterio de impureza. Se divide los datos en dos ramas, una para cada posible valor de la característica, y se repite el proceso recursivamente para cada subgrupo de datos resultante hasta que se alcanza algún criterio de detención (por ejemplo, una profundidad máxima del árbol o un número mínimo de datos en cada nodo).

### A.0.5 Arból de decisión

Un árbol de decisión es un modelo de aprendizaje automático que utiliza una estructura de árbol para tomar decisiones. Cada nodo del árbol representa una característica o variable que se utiliza para tomar una decisión, y cada rama del árbol representa el resultado de la decisión. El objetivo del árbol es dividir los datos en grupos más homogéneos posible en función de las características relevantes [3].

Un ejemplo de estos arboles de decisión se visualizan en la siguiente figura:




Aqui vemos como el arbol de decisión se divide en hojas por cada nodo, y las que no alcanzan a resolverse o a clasificarse genera otro nodo con una profundidad mayor.

# References


[1] E. Alpaydin. *Introduction to Machine Learning*. MIT Press, 2nd edition, 2010.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] L. Breiman. *Classification and Regression Trees*. Routledge, 1984.

[4] J. H. Friedman, T. Hastie, and R. Tibshirani. *The Elements of Statistical Learning*. Springer, 2001.

[5] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[6] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2nd edition, 2009.

[7] Instituto Nacional de Estadística y Geografía. Encuesta Nacional de Vivienda 2020 (ENV 2020). `https://www.inegi.org.mx/programas/env/2020/`.

[8] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning*. Springer, 2013.