**DATASET:** [**https://www.kaggle.com/datasets/arnabchaki/data-science-salaries-2023**](https://www.kaggle.com/datasets/arnabchaki/data-science-salaries-2023)

**Planteamiento del problema de negocio**

En este estudio, nuestro objetivo es **predecir el salario de profesionales en ciencia de datos** en función de diversas variables relacionadas con el trabajo, la experiencia y la ubicación.

Hemos elegido el conjunto de datos **"Salarios de ciencia de datos 2023"** porque:

1. Contiene información relevante como **título del trabajo, nivel de experiencia, tipo de empleo, ubicación de la empresa y salario** .
2. Incluye variables **numéricas y categóricas** , lo que permite aplicar técnicas de preprocesamiento como **PCA (no supervisado) y modelos de regresión (supervisado)** .
3. El salario es una variable **continua** , lo que nos lleva a un problema de **regresión** en lugar de clasificación.
4. Es útil en la vida real: entender qué factores influyen en el salario puede ayudar tanto a compañeros como a profesionales de datos.

**Estructura del análisis**

Vamos a dividir el análisis en varias fases:

**1️ Exploración y análisis inicial de los datos.**

Antes de aplicar cualquier modelo, debemos entender cómo están estructurados los datos:

* **Distribución de la variable objetivo (salario)** → Ver si es normal, sesgada, etc.
* **Correlación entre variables numéricas** → Identificar si hay relaciones fuertes con el salario.
* **Análisis de la cardinalidad de las variables categóricas** → Ver cuántos valores únicos hay en cada una (ejemplo: ¿cuántos títulos de trabajo diferentes existen?).
* **Detección de valores atípicos (outliers)** → Salarios extremadamente altos o bajos podrían distorsionar el modelo.
* **Manejo de valores nulos** → Determinar si hay datos faltantes y cómo tratarlos.

**2️ Aplicación de un modelo no supervisado: PCA**

Antes de entrenar un modelo de regresión, queremos **reducir la dimensionalidad** con PCA para:

* **Identificar patrones ocultos** en los datos numéricos.
* **Reducir la redundancia** si hay variables muy correlacionadas.
* **Visualice los datos en un espacio de menor dimensión** y ver si se pueden agrupar según ciertas características.

El PCA solo se aplicará a **variables numéricas**, así que las categóricas aún no se transforman en esta etapa.

**3️ Preparación de los datos para el modelo supervisado**

Aquí ya nos preparamos para la regresión:

**a) Creación del Pipeline de preprocesamiento**

Dentro del Pipeline aplicaremos:

* **OneHotEncoding a variables categóricas**: Convertimos texto en variables numéricas.
* **Estandarización o normalización de variables numéricas**: PCA y modelos de regresión funcionan mejor con datos escalados.
* **Manejo de valores desconocidos en prueba** : Configuramos OneHotEncoder(handle\_unknown='ignore')para evitar errores si aparecen categorías nuevas en prueba.

**b) División Tren/Prueba**

Dividimos los datos en entrenamiento y prueba para evaluar el rendimiento del modelo.

**4️ Entrenamiento de modelos supervisados**

* Probamos diferentes modelos de regresión:
  + **Regresión Lineal** (modelo base para interpretar resultados).
  + **Random Forest Regresor** (captura relaciones no lineales).
  + **XGBoost o LightGBM** (modelos más avanzados y eficientes).

**5 Evaluación del modelo**

* **Clave métrica**: Usaremos **MSE (Error Cuadrático Medio) y R²** para medir la precisión del modelo.
* **Validación cruzada**: Para evaluar el rendimiento real y evitar el sobreajuste.
* **GridSearchCV** : Para optimizar hiperparámetros y encontrar la mejor configuración del modelo.

**Orden en el que haremos cada paso**

1. **Exploración de los datos** (distribuciones, correlaciones, cardinalidad, valores atípicos, etc.). MIERCOLES SI QUEREIS MARIO
2. **Aplicación de PCA** (solo a variables numéricas, para análisis exploratorio). JUEVES
3. **Creación del Pipeline de preprocesamiento** (OneHotEncoding, escalado, etc.).

**VIERNES CLASE MAS DUDAS**

1. **División Tren/Prueba** . FINDE
2. **Entrenamiento de modelos supervisados**. FINDE
3. **Evaluación del modelo con validación cruzada y GridSearchCV** . FINDE