modulo\_2/

│

├── data/

│   ├── raw/               # Datos originales (no se suben a GitHub)

│   └── processed/         # Datos limpios/listos para análisis

│

├── notebooks/

│   ├── EDA.ipynb          # Exploración y visualización

│   └── Modeling.ipynb     # Pruebas y evaluación de modelos

│

├── src/

│   └── preprocessing.py   # Funciones reutilizables (cargar, limpiar, transformar)

│

├── main.py                # Para ejecutar todo junto (opcional, estilo producción)

├── README.md

├── .gitignore

├── pyproject.toml

└── poetry.lock

**2. Contenido de los notebooks**

Ahora que ya tienes tus notebooks, es importante definir qué debe ir en cada uno:

* **01\_analisis\_inicial.ipynb**:
  + Este notebook debe tener el análisis preliminar de cada dataset antes de hacer cualquier transformación.
  + Incluye la carga de los datos, estadísticas descriptivas, la identificación de valores nulos, duplicados y tipos de datos.
  + Este es el lugar adecuado para hacer un primer vistazo a cada dataset y decidir si se necesitan hacer transformaciones.
* **02\_preprocesamiento.ipynb**:
  + Aquí puedes manejar la limpieza y transformación de los datos. Realiza operaciones como la eliminación de valores nulos, la conversión de tipos de datos, el tratamiento de outliers, y la creación de nuevas características.
  + Este notebook es clave para dejar los datos listos para el análisis y modelado.
* **03\_EDA.ipynb**:
  + Este archivo debe centrarse en el análisis exploratorio de datos (EDA) profundo una vez que los datos estén limpios.
  + Realiza visualizaciones, correlaciones, histogramas, gráficos de dispersión, etc., para entender las relaciones entre las variables y encontrar patrones interesantes en los datos.
* **04\_modelado.ipynb**:
  + Este notebook debe ser el lugar donde implementas los algoritmos de modelado y validación.
  + Puedes dividir este archivo en diferentes secciones, como **entrenamiento**, **evaluación** y **ajuste de hiperparámetros**.

¡Perfecto! Ahora que tienes la estructura del proyecto organizada, vamos a definir un **paso a paso claro y ordenado** para desarrollar tu proyecto de predicción de riesgo de mora.

**🧭 PASO A PASO PARA DESARROLLAR TU PROYECTO**

**✅ 1. Inicializa y organiza tu proyecto**

🔹 Ya lo hiciste, pero para registrar:

* Estructura de carpetas:

css

CopyEdit

data/

├── raw/

└── processed/

notebooks/

src/

└── preprocessing.py

main.py

pyproject.toml

README.md

* Usa **Poetry** para gestionar dependencias
* Ignora data/raw/ en .gitignore

**🧪 2. Exploración de datos (EDA)**

🔸 **Archivo:** notebooks/EDA.ipynb

1. **Importa las librerías necesarias**
2. **Carga los datasets** usando funciones del archivo preprocessing.py
3. **Analiza columnas**: tipos de datos, valores nulos, duplicados
4. **Visualiza**:
   * Histogramas
   * Boxplots
   * Correlaciones
   * Distribución de la variable objetivo (mora)
5. **Genera insights** sobre el comportamiento de los clientes

📌 Tip: Si tienes varios datasets, usa pestañas en el notebook o secciones claras por archivo.

**🧹 3. Preprocesamiento de los datos**

🔸 **Archivo:** src/preprocessing.py

1. Crea funciones como:
   * cargar\_datos(path)
   * limpiar\_datos(df)
   * codificar\_variables(df)
   * normalizar\_variables(df)
2. Guarda el resultado limpio en data/processed/

✅ Luego vuelve a EDA.ipynb y carga el dataset limpio para análisis.

**📊 4. Reducción de dimensionalidad (PCA)**

🔸 **Archivo sugerido:** notebooks/PCA.ipynb o misma EDA

1. Aplica **StandardScaler**
2. Aplica **PCA**
3. Visualiza componentes principales
4. Interpreta qué variables aportan más a cada componente

**🧠 5. Entrenamiento de modelos predictivos**

🔸 **Archivo:** notebooks/Modeling.ipynb

1. Divide el dataset: train/test
2. Entrena modelos:
   * Regresión logística
   * Random Forest
   * XGBoost
3. Compara métricas:
   * Accuracy, Recall, Precision
   * AUC-ROC
4. Ajusta hiperparámetros si es necesario

**📈 6. Evaluación final del mejor modelo**

🔸 Visualiza:

* Curva ROC
* Matriz de confusión
* Importancia de variables

**🧩 7. Genera conclusiones y recomendaciones**

🔸 Archivo sugerido: notebooks/Conclusiones.ipynb o final del EDA.ipynb

* ¿Qué patrones encontraste?
* ¿Qué variables explican la mora?
* ¿Qué modelos funcionan mejor?
* ¿Qué decisiones puede tomar Confirmeza S.A.S.?

**🛠️ 8. Automatiza (opcional)**

🔸 Usa main.py para que ejecute en orden:

python

CopyEdit

from src.preprocessing import cargar\_datos, limpiar\_datos

from src.modeling import entrenar\_modelos

# Ejecutar todo con un solo script

**🗂️ 9. Documenta el proyecto**

🔸 README.md debe incluir:

* Resumen del problema
* Estructura del proyecto
* Requisitos y cómo instalar
* Ejemplos de uso o resultados

**💡 ¿Te gustaría que te genere una plantilla base para cada notebook?**

Puedo dejarte listos:

* EDA.ipynb
* Modeling.ipynb
* PCA.ipynb

Con títulos, secciones y algunas celdas predefinidas. ¿Lo hacemos?