Machine Learning Pipelines

➢ *What is a Machine Learning Pipeline?*

A **pipeline** in machine learning is a step-by-step *assembly line* that automatically handles everything from **data preprocessing** to **model training**—all in a single streamlined structure.

➢ *What Does It Include?*

A typical ML pipeline includes:

1. **Data cleaning & preprocessing** (e.g., handling missing values)
2. **Feature transformation** (e.g., scaling, encoding)
3. **Model training**
4. **Evaluation**
5. **Model saving (Pickling)**
6. **Deployment** or **Prediction**

All of this is tied together so you don't have to manually repeat steps.

➢ *Why Use Pipelines?*

- **Clean code**: Everything is organized in one structure
- **Reproducibility**: You can rerun the same process reliably
- **Less error-prone**: Avoid mistakes from copy-pasting code
- **Works with cross-validation & grid search easily**
- **Production ready**: You can save and load pipelines directly

➢ *Example(Titanic Dataset):*

In that dataset, you went through:

- Splitting the Titanic dataset
- Imputing missing values (e.g., age)
- Creating a pipeline with preprocessing + model
- Comparing Pipeline vs make_pipeline (both create pipelines, but Pipeline lets you name the steps)
- Pickling (saving) the model
- Cross-validating using the full pipeline

➢ *Pipeline vs make_pipeline:*

- Pipeline: You define and name each step explicitly (good for customization).
- make_pipeline: Quick and automatic, but steps are unnamed.

➢ *What is Pickling?*

Pickling is just saving your trained pipeline (model + preprocessing steps) so that you can reuse it later—without retraining from scratch.

➢ *Summary:*

A Machine Learning Pipeline is your complete ML workflow automated, efficient, and ready to deploy.
It's like wrapping your entire process in one neat box and pressing *play*.
