**Explanation of Each File and Directory**

**config.yml**

* **Purpose**: This configuration file contains all the necessary parameters for your project. It includes details for the MongoDB connection, data paths, data processing parameters, and model training parameters.
* **Content**:
  + MongoDB connection URI, database name, and collection name.
  + Column names for dataframes.
  + Data paths for datasets.
  + Parameters for sliding window algorithm, filtering, and PCA.
  + Model training parameters for classifiers and neural networks.

**data/**

* **Purpose**: This directory contains all the raw data files for your project. The data is organized into subdirectories by class (e.g., class\_A, class\_B).
* **Content**:
  + CSV files containing the data for each class.

**notebooks/**

* **Purpose**: This directory contains Jupyter Notebooks that perform various stages of your project, such as data creation, EDA, and model training.
* **Files**:
  + aiot\_project.ipynb: The main notebook for your project that integrates data processing, visualization, and model training.
  + 01\_dataset\_creation.ipynb: Notebook for creating the dataset and uploading it to MongoDB.
  + 02\_single\_instance\_eda\_data\_engineering.ipynb: Notebook for exploratory data analysis and data engineering on a single data instance.
  + 03\_dataset\_eda\_data\_engineering.ipynb: Notebook for exploratory data analysis and data engineering on the entire dataset.

**utils/**

* **Purpose**: This directory contains utility scripts that provide reusable functions for data processing and other tasks.
* **Files**:
  + \_\_init\_\_.py: Makes the utils directory a package. It imports functions from data\_processing.py so they can be easily accessed.
  + data\_processing.py: Contains functions for data processing, including sliding window application, filtering, flattening data, and renaming columns.

**README.md**

* **Purpose**: Provides an overview of your project, including setup instructions, usage, and any other relevant information.
* **Content**:
  + Introduction to the project.
  + Instructions for setting up the environment and running the code.
  + Description of the project structure and files.

**requirements.txt**

* **Purpose**: Lists all the Python packages required for your project. It allows easy installation of dependencies using pip.
* **Content**:
  + List of Python packages such as pandas, numpy, matplotlib, scipy, pymongo, pyyaml, and any other dependencies.

**Detailed Breakdown of the Notebooks**

**aiot\_project.ipynb**

* **Purpose**: Main notebook that ties together all aspects of your project, including data fetching, processing, visualization, and model training.
* **Content**:
  + Loads configuration and data from MongoDB.
  + Applies filtering and data transformations.
  + Splits the data into training and test sets.
  + Scales the data and performs PCA.
  + Trains statistical and neural network models.
  + Evaluates the models and visualizes the results.

**01\_dataset\_creation.ipynb**

* **Purpose**: Handles the initial data setup by uploading the dataset to MongoDB.
* **Content**:
  + Connects to MongoDB.
  + Loads and processes the raw data.
  + Uploads the processed data to the MongoDB collection.

**02\_single\_instance\_eda\_data\_engineering.ipynb**

* **Purpose**: Performs exploratory data analysis and data engineering on a single data instance.
* **Content**:
  + Loads a single data instance.
  + Performs EDA including plotting and statistical analysis.
  + Applies data transformations such as sliding window and filtering.

**03\_dataset\_eda\_data\_engineering.ipynb**

* **Purpose**: Performs exploratory data analysis and data engineering on the entire dataset.
* **Content**:
  + Loads the entire dataset.
  + Performs EDA including plotting and statistical analysis for the whole dataset.
  + Applies data transformations such as sliding window and filtering.

**Detailed Breakdown of utils/data\_processing.py**

* **Functions**:
  + sliding\_window\_pd: Applies the sliding window algorithm to a DataFrame.
  + apply\_filter: Applies a filter to a NumPy array.
  + filter\_instances: Applies a filter to a list of DataFrames.
  + flatten\_instances\_df: Flattens a list of DataFrames into a single DataFrame.
  + df\_rebase: Reorders and renames DataFrame columns.
  + rename\_df\_column\_values: Renames DataFrame column values based on unique labels.
  + are\_lists\_equal: Checks if two lists are equal.
  + encode\_labels: Encodes target labels using LabelEncoder.
  + list\_files\_in\_folder: Lists CSV files in a folder.

**utils/init.py**

* **Purpose**: Makes the utils directory a package and provides easy access to utility functions.
* **Content**:
  + Imports all functions from data\_processing.py.

This structure and explanation should give you a clear understanding of what each part of your project does and how everything fits together.