**save\_sets Function Documentation**

**Purpose:**

Saves multiple datasets to CSV files in a specified directory.

**Parameters:**

* X\_train (pd.DataFrame): Training features dataset.
* y\_train (pd.Series): Training target dataset.
* X\_val (pd.DataFrame): Validation features dataset.
* y\_val (pd.Series): Validation target dataset.
* base\_path (str): Directory path where CSV files will be saved.
* X\_test (pd.DataFrame, optional): Testing features dataset.
* y\_test (pd.Series, optional): Testing target dataset.

**Functionality:**

* Saves X\_train to X\_train.csv
* Saves y\_train to y\_train.csv
* Saves X\_val to X\_val.csv
* Saves y\_val to y\_val.csv
* If X\_test is provided, saves it to X\_test.csv
* If y\_test is provided, saves it to y\_test.csv

**Example Usage:**

*from my\_krml\_24701184.data.sets import save\_sets*

*base\_path=’path directory'*

*save\_sets(base\_path, X\_train, y\_train, X\_val, y\_val, X\_test)*

*note: only input whichever you have*

**load\_sets Function Documentation**

**Purpose:**

Loads multiple datasets from CSV files in a specified directory into a dictionary of Pandas DataFrames and Series.

**Parameters:**

* base\_path (str): Directory path from where CSV files will be loaded.

**Returns:**

* **dict**: A dictionary where keys are dataset names ("X\_train", "y\_train", "X\_val", "y\_val", "X\_test", "y\_test") and values are corresponding DataFrames or Series. Missing files will not be included in the dictionary.

**Functionality:**

* Loads X\_train.csv into DataFrame
* Loads y\_train.csv into Series
* Loads X\_val.csv into DataFrame
* Loads y\_val.csv into Series
* If X\_test.csv exists, loads it into DataFrame
* If y\_test.csv exists, loads it into Series

**Example Usage:**

*from my\_krml\_24701184.data.sets import load\_sets*

*datasets = load\_sets("path\_to\_directory")*

*X\_train = datasets.get("X\_train")*

*y\_train = datasets.get("y\_train")*

*X\_val = datasets.get("X\_val")*

*y\_val = datasets.get("y\_val")*

*X\_test = datasets.get("X\_test")*

*y\_test = datasets.get("y\_test")*