EPIGENE LABS

Documentation for Data Preprocessing Notebook

By: Laura Fuentes

Overview

Objective: Preprocessing and exploratory analysis of a biomedical/genomic dataset.

- Focuses on feature extraction and transformation based on keywords search, and variability of data across columns.
- Includes various steps to process gender, age, cancer characteristics, and more.

Environment Setup

Main libraries used

- `os:` File management.
- `pandas, numpy:` Data manipulation.

Sections Overview

- Base Dataframe
- Gender Feature
- Age Feature
- Cancer Stage Feature
- Cancer Grade Feature
- Cancer Type
- Sample Type
- Cancer Subtype

1. Base Dataframe

Objective: Load datasets, and filter out non informative columns based on their variability.

Approach:

- Identify columns with high variability on first file (**categorical variables:** more than 1 category, **numerical variables:** std>0).
- Merge files, by keeping columns identified as variable ones on the first dataset, plus the ones containing `'characteristics'` information.
- Exclude time-related columns like `last_update_date`.

2. Gender Feature

Objective: Process and analyze gender information.

Approach:

- Get gender feature based on keyword search, where `'ovarian'` cancer type, is directly assigned as female patients.
- If no cancer type found, gender is assigned from columns with the word `'gender'` on it.
- For the remaining samples, the gender is unknown.
- Standardize gender values for consistency.

3. Age Feature

Objective: Extract, clean, and analyze age-related data.

Approach:

- Assign gender based on keyword search. Where columns with the word ‘age’ in its name, or content are taken.
- Format the values to integer.

4-5 Cancer Stage and Grade

Objective: Standardize cancer stage and grade data for consistency and analysis.

Approach:

- Identify relevant columns using keyword searches (`stage, grade`).
- Fill missing values using backfill (`bfill`).
- Normalize values into categories:
  - **Stage**: `I, II, III, IV` (e.g., `1, i, early` → `I`).
  - **Grade**: `I, II, III, IV` (e.g., `1` → `I`).
- Assign `NaN` to unrecognized or missing values.

6. Cancer Type

Objective: Extract and categorize cancer type for better analysis.

Approach:

- Identify columns related to cancer type using keyword searches (`ovar`, `breast`).
- Add the categorized data as a new column (`cancer_type`) in the main DataFrame.
- Outcome: cancer type is classified into clear categories (`ovarian`, `breast`, `unaffected`) for improved consistency and downstream processing.

7. Sample Type

Objective: Extract and categorize clean sample types.

Approach:

- Identify columns related to sample type using keywords (`type`, `tumor`, `tissue`, `primary`, `normal`).
- Combine and clean data, filling missing values.
- Assign values (**"primary"** or **"normal"** if matching keywords are found)
- Assign `NaN` for unmatched samples.

8. Cancer Subtype

Objective: Extract and categorize sample types and cancer subtypes for enhanced analysis.

Approach:

- Identify subtype-related columns using keywords (`type`, `tumor`, `tissue`, `serous`, `endometrioid`).
- Extract unique subtypes from raw data and clean values.
- Assign subtype values (e.g., **"serous"**, **"mucinous"**) based on matches.
- Assign `NaN` if no matching subtype is found.

More details?

Contact: lauracarolinafuentesquintero@gmail.com