Data
/**P**rocessing
/**C**leaning
/**I**mputation
Pipeline

on Real-Time data

Sana Wajid

12/6/19

0P. Step

P = Processing
C = Cleaning
I = Imputation

Assumptions:

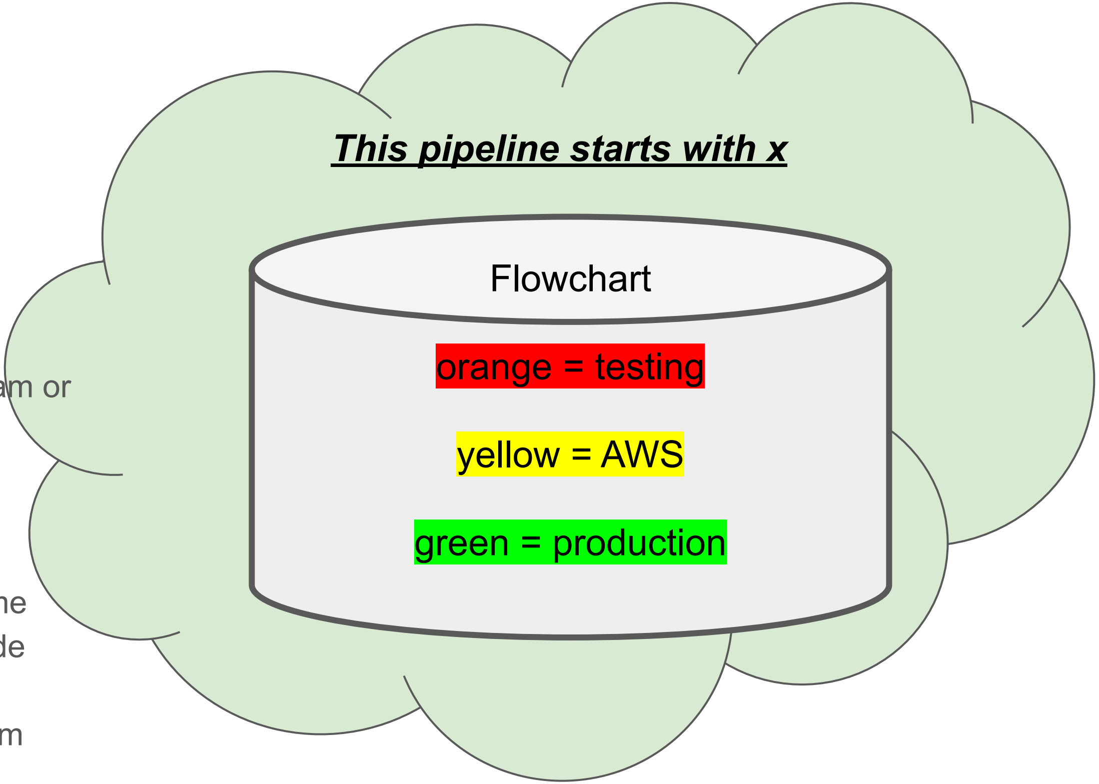
- For testing using csv files
- A pipeline can move it’s starting point upstream or downstream.

Checks:

1. Listed in order of least to most processing time
2. Numbers will refer to function numbers in code comment or headers in Jupyter notebook
    a. e.g. 1P-2: File contains minimum number of headers

Libraries:

Libraries used in {pandas, scikit-learn} ∈ Python, sed/awk, unix



This pipeline starts with x

Flowchart

orange = testing

yellow = AWS

green = production

2P. Raw data conversion to pandas

Assumptions:

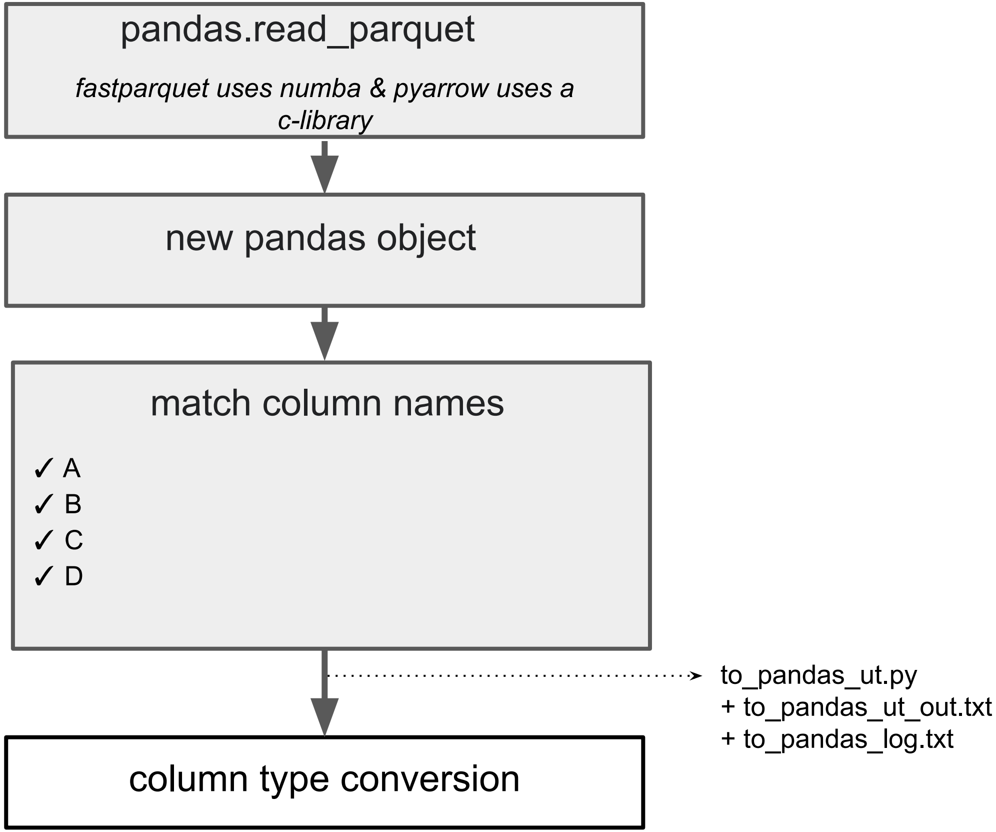
- File contains minimum number of headers
- File contains minimum number of rows
- File contains correct headers
- Column names don’t contain spaces

Functions:

1. `to_pandas`

Libraries:

`read_parquet` ∈ pandas ∈ Python



pandas.read_parquet

fastparquet uses numba & pyarrow uses a c-library

new pandas object

match column names

✓ A
✓ B
✓ C
✓ D

to_pandas_ut.py
+ to_pandas_ut_out.txt
+ to_pandas_log.txt

column type conversion

7I. Impute strategy, simple: scikit-learn

Assumptions:

- fill value is defined per building and per x
- every t_0 : t_{end} has a row

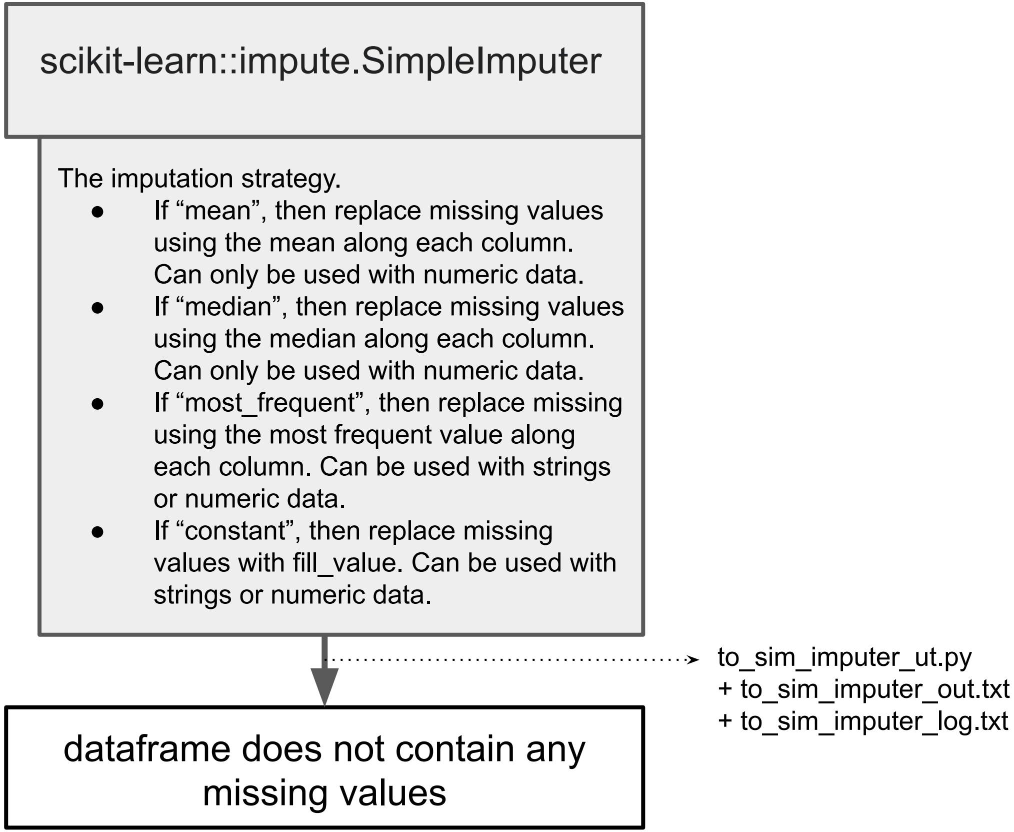
Functions:

accessor (get) functions return dataframe

1. `get_flanking_cluster` (dataframe obj of pandas type)

Libraries:

scikit-learn ∈ Python



scikit-learn::impute.SimpleImputer

The imputation strategy.

- If “mean”, then replace missing values using the mean along each column. Can only be used with numeric data.
- If “median”, then replace missing values using the median along each column. Can only be used with numeric data.
- If “most_frequent”, then replace missing using the most frequent value along each column. Can be used with strings or numeric data.
- If “constant”, then replace missing values with fill_value. Can be used with strings or numeric data.

to_sim_imputer_ut.py
+ to_sim_imputer_out.txt
+ to_sim_imputer_log.txt

dataframe does not contain any missing values

7I. Impute strategy, simple: scikit-learn

Assumptions:

- fill value is defined per building and per x
- every t_0 : t_{end} has a row

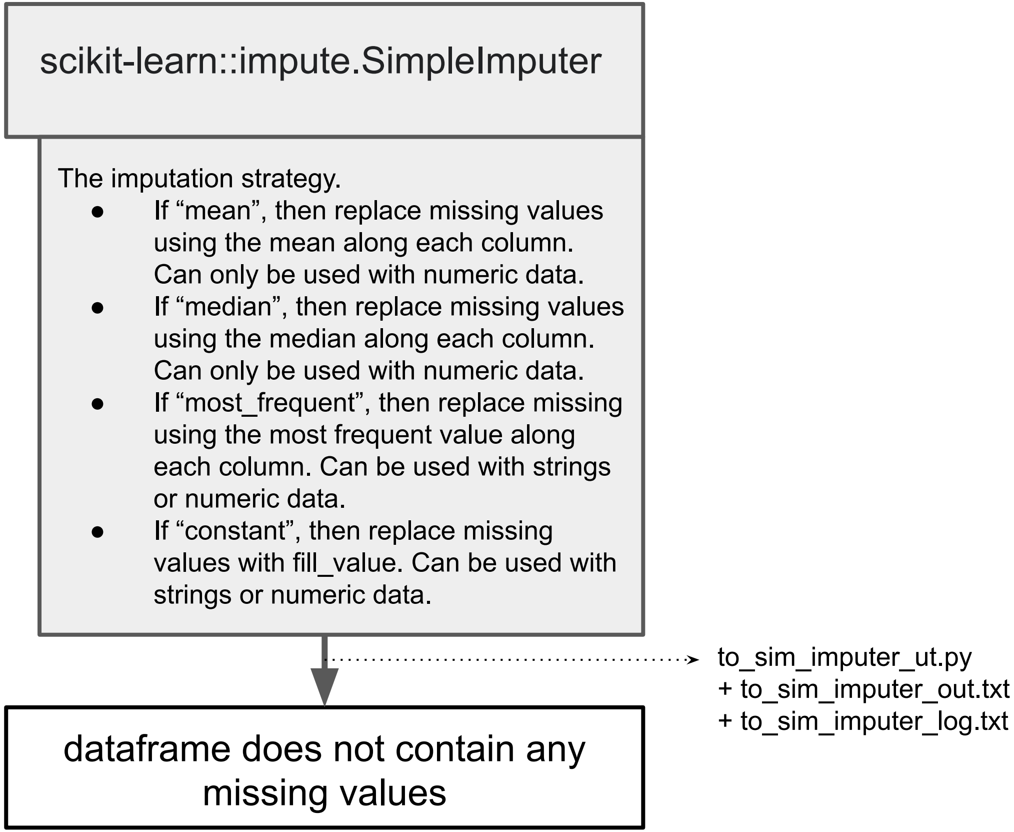
Functions:

mutators return True or False

2. `impute_cluster_by_mean`
3. `impute_cluster_by_median`
4. `impute_cluster_by_most_freq`
5. `impute_cluster_by_constant`

Libraries:

scikit-learn ∈ Python



scikit-learn::impute.SimpleImputer

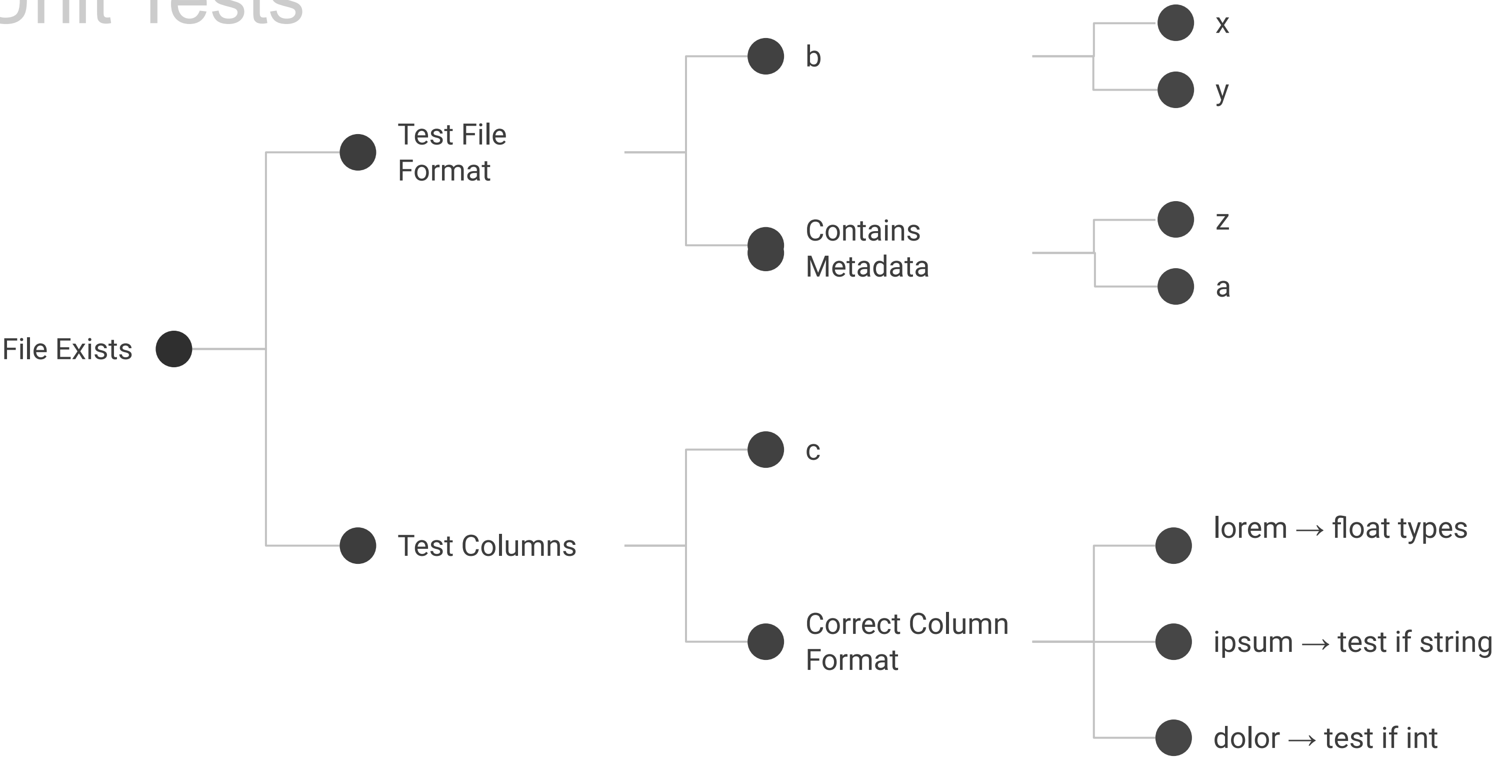
The imputation strategy.

- If “mean”, then replace missing values using the mean along each column. Can only be used with numeric data.
- If “median”, then replace missing values using the median along each column. Can only be used with numeric data.
- If “most_frequent”, then replace missing using the most frequent value along each column. Can be used with strings or numeric data.
- If “constant”, then replace missing values with fill_value. Can be used with strings or numeric data.

to_sim_imputer_ut.py
+ to_sim_imputer_out.txt
+ to_sim_imputer_log.txt

dataframe does not contain any missing values

Unit Tests


File Exists
Test File Format
Test Columns
b
Contains Metadata
c
Correct Column Format
x
y
z
a
lorem → float types
ipsum → test if string
dolor → test if int

References

etc, more will be added