Data Pre-Processing

Data Cleaning and Preparation

Data Pre-Processing

Data Cleaning and Preparation

Major Data Pre-Processing Tasks

- Missing Values
- Data Transformation
- String Manipulation

More than 80 % of the work in Data Analysis or in DataScience Involves Data Pre-Processing

Data Pre-Processing

Data Cleaning and Preparation ⟷ Handling Missing Values

Data Pre-Processing

Data Cleaning and Preparation ⟷ Handling Missing Values

To Handle Missing Data Pandas Uses;

Sentinel value: Used Represent The Missing Value NaN (Not A Number)

To Process NA values: The Pandas Provides

`isnull` `notnull` `fillna` `dropna`

Data Pre-Processing

Data Cleaning and Preparation ⟷ Handling Missing Values

Filtering Missing Values

`how='all'`

Will only drop rows that are all NA

`how='any'`

Will only drop any row or column having NA

Pass

Pass

`dropna`

By default drops any row containing a missing value

Pass

Pass

`how='all', axis=1`

Will only drop columns that are all NA

`thresh='n'; N=integer value`

Will drop only the n rows with NA values

Data Pre-Processing

Data Cleaning and Preparation ⟷ Handling Missing Values

Filling the Missing Values

Argument or Syntax	Description
`fillna()`	Scalar value object use to fill missing values
`fillna(dictionary)`	Dict-like object use to fill missing values
`fillna(dictionary, inplace=True)`	Modify the calling object without producing a copy
`fillna(mehod=ffill); or fillna(mehod=bfill)`	Interpolation; by default 'ffill' forward fills the missing vallues, bfill will backward fills the missing values
`fillna(method='ffill', limit=3)`	For forward and backward filling, maximum number of consecutive periods to fill
`fillna(0, axis=1)`	Axis to fill on; default axis=0, we can pass axis=1 to fill columns; helpful when we use method option to fill the missing values

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Removing Duplicate Rows

`duplicated`

- Returns a Boolean series indicating whether each row is a duplicate or not
- By default consider all of the columns

`drop_duplicates`

- Returns a DataFrame where the duplicated array is False
- By default consider all of the columns

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

DT Using Function Or Mapping

‘map’ method

The map is a method applied on a Series, it accepts a function or dict-like object containing a map-ping elements.

Ex: In the DataFrame ‘states’ shown here has 3 columns: A, B, C; of which A has a Series of languages. Now we can use map method to map the corresponding states and foods element by element wise one after the other.

states

	A	B	C
0	Kannada		
1	Telugu		
2	Hindi		
3	Tamil		
4	Malayalam		
5	Hindi		

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Replacing Values

'replace' method

It can be used to replace single values or Multiple values

To replace the multiple values at once we can use list of values to be replaced and then the sentinel value

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Renaming Indexes of Rows and Columns

‘index.map()’ method

- It modifies the original DataFrame or Series Rows and Columns
- We can assign index to a data to modify the Rows and Columns in-place

‘rename()’ method

- It creates a Transformed version of the DataFrame without modifying the original data.
- We can pass ‘inplace=True’ to modify the DataFrame inplace

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Data Binning or Discretization

'cut' method

- The Data values are groped based on discrete values or bin size
- The bin size are not regular size instead, the specified size is used

'qcut' method

- The Data values are discretized based using sample quantiles; so roughly equal size bins are obtained
- We can also pass our own quantiles to descritize the data values

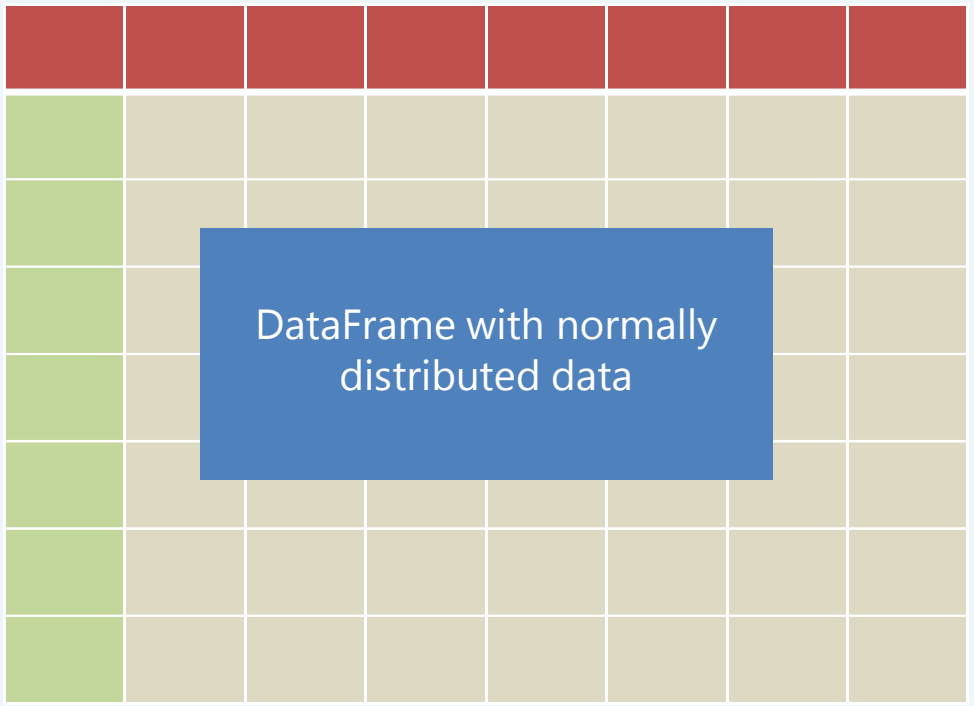
Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Detect & Filter Outliers

Boolean DataFrame on a Series of DataFrame to filter specific columns (column based)

Boolean DataFrame with 'any(1)' to filter all rows of DataFrame



Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Random Sampling

Randomly reordering a Series or the rows in a DataFrame is known as Random Sampling or Permuting

```
permute = numpy.random.permutation
```

```
Df.take(permute)
```

To select a random subset without replacement, you can use the sample method on Series and DataFrame

`Df.sample(permute)` : we can use `'inplace=True'` To select a random subset with replacement

Data Pre-Processing

Data Cleaning and Preparation ⟷ Data Transformation

Computing Dummy Variables

Another very important transformation for statistical modeling or machine learning applications is converting a categorical variable into a “dummy” or “indicator” matrix

```
get_dummies
```

```
get_dummies(pd.cut())
```

Data Pre-Processing

Data Cleaning and Preparation ⟷ String Manipulation

String Object Methods

String Data
Python_sentence = ‘python,Is, a programming, Language’

Some Simple String Manipulation Methods

Method	Its work
Python_sentence.split(',')	[‘python’, ‘Is’, ‘ a programming’, ‘ Language’]
ps = [x.strip() for x in Python_sentence.split(',')]	[‘python’, ‘Is’, ‘a programming’, ‘Language’]
':'.join(ps)	‘python:Is:a programming:Language’

Data Pre-Processing

Data Cleaning and Preparation ⟷ String Manipulation

String Object Methods

Other String Manipulation Methods

Argument	Argument	Argument	Argument
count	join	replace	lower
endswith	index	strip	upper
startswith	find	split	casefold

Data Pre-Processing

Data Cleaning and Preparation ⟷ String Manipulation

Regular Expressions

Regular expressions are very much helpful & provide a flexible way to search or match (often more complex) string patterns in text

Steps

`import re`

You can apply directly

You can compile it use repeatedly: ‘`re.compile()`’

Data Pre-Processing

Data Cleaning and Preparation ⟷ String Manipulation

Regular Expressions

Regular Expression Methods

Argument	Argument
`split`	`findall`
`finditer`	`match`
`search`	`sub, subn`

Data Pre-Processing

Data Cleaning and Preparation ⟷ String Manipulation

Applying String Methods On Pandas Series

Vectorized String Functions

Some of Vectorized String Functions

Argument	Argument
split	findall
count	lower, upper
search	contains