

Pandas Tutorial

By: Yotam Perlitz - yotam.pe@samsung.com

This is a small summary of pandas commands, it represents a small set of most needed DataFrame manipulation, in case a deeper dive in to the subject is wanted, check out this [much better tutorial](#).

Import pandas

```
1 | import pandas as pd
```

Write/Read DF as .h5:

```
1 | my_df.to_hdf(path_to_folder+'/my_df_saving_name.h5', 'my_df_saving_name')
2 | my_df = pd.read_hdf(path_to_folder+'/my_df_saving_name.h5')
```

Write/Read DF as .json:

```
1 | my_df.to_json(path_or_buf=path_to_folder+'/my_df_saving_name.json')
2 | my_df = pd.read_json(path_or_buf=path_to_folder+'/my_df_saving_name.json')
```

Initialize DataFrame with a few columns:

```
1 | my_df = pd.DataFrame(columns = ['dir_name', 'frame_name', 'false_positive'])
```

Add a row to DF (note the use of `ignore_index=True` {python} which tells the DataFrame to set the index as row enumeration, as in a simple list).

```
1 | my_df = my_df.append({'dir_name': dir_name, 'frame_name': frame_name,
2 |                      'false_positive': num_fp}, ignore_index=True)
```

Add a new column to DF:

```
1 | my_df['false_negative'] = pd.Series(false_negative_list)
2 |
3 | # note that len(false_negative_list) has to be equal to my_df.shape[0]
```

Access a certain column of the DF:

```
1 | my_series = my_df['dir_name']
```

A [series can be turned in to a list](#) using:

```
1 | series.to_numpy()
```

Create [iterator](#) of rows of DF:

```
1 | my_df.iterrows()
```

[Collapse rows](#) and apply manipulation over duplicates:

```
1 | my_df =  
  | my_df.groupby('dir_name').agg({'false_negative':'mean','false_positive':'mean'})  
2 |  
3 | # note that all column values must be numeric, in case not true (for example for  
  | false_positive), can use: my_df['false_positive'] =  
  | pd.to_numeric(my_df['false_positive'])
```

[Drop duplicates](#) (taking only the first value):

```
1 | my_df_row_per_dir = my_df.drop_duplicates(subset=['dir_name'],keep='first')
```

[Drop columns](#)

```
1 | my_df.drop(columns=['false_negative'])
```

Keep certain values of DF:

```
1 | my_df_zero_fp = my_df[my_df['false_positive']==0]
```

[Count occurrences](#) in a certain column:

```
1 | my_df['false_positive'].value_counts()  
2 |  
3 | # this will produce a table in which each row holds a value (of 'false_positive') and  
  | the number of occurrences, for example:  
4 |  
5 | false positive value - number of occurrences  
6 |      5                1218837  
7 |      1                288189  
8 |     10                167364  
9 |      6                118085  
10 |     17                68663  
11 |     14                47808  
12 |     18                45225
```

Sort DF by column:

```
1 | my_df.sort_values(by='false_positive',ascending=False)
```

View top / bottom:

```
1 my_df.head(3)
2 my_df.bottom(3)
```

Saving metadata along with DF to .h5

```
1 def h5store(filename, df, **kwargs):
2     store = pd.HDFStore(filename)
3     store.put('mydata', df)
4     store.get_storer('mydata').attrs.metadata = kwargs
5     store.close()
6
7 def h5load(store):
8     data = store['mydata']
9     metadata = store.get_storer('mydata').attrs.metadata
10    return data, metadata
11
12 metadata = {'City': 'Tel-Aviv'}
13 h5store(filename_to_save, my_df, **metadata)
14
15 with pd.HDFStore(filename) as store:
16     data, metadata = h5load(store)
```