

### Read and Write

```
import pandas as pd

pd.read_csv('filepath_or_buffer', sep=',',
names=None, index_col=None)

df.to_csv('path_or_buf', sep, columns=None,
header=True, index=True, index_label=None,
mode='w')
```

### Numerical Features and Nans

```
df.sort_values(by=feat, ascending=False)
df[feat].dropna()
df[feat].isna()
```

sort table according to the values of a columns  
remove lines with a NaN value  
check if the value of *feat* is NaN in the table

### Categorical Features

```
pd.get_dummies(df[feature])
```

Transform a categorical feature into dummy variables

### Features Visualization

```
df[feats].plot(kind=['density', 'bar'],
subplots=True, layout=(1, 2),
sharex=True, figsize=)
```

distribution of numeric features

### General Infos and Basic Statistics

```
df.info()
df.describe()
df.describe(include=['object', 'bool'])
```

general infos  
basic statistics on numerical features  
include non-numerical features

### Apply Functions

```
df.apply(my_function)
# ex: df.apply(lambda x: )
df['feat'] = df['feat'].map(d)
# or
df = df.replace({'feat': d})
```

apply a function  
replace values in a column according to dict d

### Group by

```
df.groupby(['feat']).agg([list_of_functions])
[columns_to_keep].func()
df.groupby([feat]).agg([list_of_functions])
[columns_to_keep].agg([list_of_functions])
```

group by a feature  
group by a feature and apply several functions

### Cross Tables

```
pd.crosstab(df['feat1'], df['feat2'],
normalize=)
df.pivot_table(['features_to_analyze'],
['grouping_feat'], aggfunc='mean')
```

confusion matrix  
pivot table

