PANDAS CHEATSHEET:

PYTHON DATA WRANGLING TUTORIAL

This Pandas cheatsheet will cover some of the most common and useful functionalities for data wrangling in Python. Broadly speaking, data wrangling is the process of reshaping, aggregating, separating, or otherwise transforming your data from one format to a more useful one.

Pandas is the best Python library for wrangling relational (i.e. table-format) datasets, and it will be doing most of the heavy lifting for us.

To see the most up-to-date full tutorial and download the sample dataset, visit the online tutorial at elitedatascience.com.

SETUP

First, make sure you have the following installed on your computer:

- Python 2.7+ or Python 3
- Pandas
- Jupyter Notebook (optional, but recommended)

*note: We strongly recommend installing the Anaconda Distribution, which comes with all of those packages. Simply follow the instructions on that download page.

Once you have Anaconda installed, simply start Jupyter (either through the command line or the Navigator app) and open a new notebook.

IMPORT LIBRARIES AND DATASET

```
import pandas as pd
pd.options.display.float_format = ‘{:,.2f}’.format
pd.options.display.max_rows = 200
pd.options.display.max_columns = 100

df = pd.read_csv(‘BNC2_sample.csv’,
                names=[‘Code’, ‘Date’, ‘Open’, ‘High’, ‘Low’
                       ‘Close’, ‘Volume’, ‘VWAP’, ‘TWAP’])
```

*The sample dataset can be downloaded here.

FILTER UNWANTED OBSERVATIONS

```
gwa_codes = [code for code in df.Code.unique() if ‘GWA_’ in code]
df = df[df.Code.isin(gwa_codes)]
```

PIVOT THE DATASET

```
pivoted_df = df.pivot(index=‘Date’, columns=‘Code’, values=‘VWAP’)
```

SHIFT THE PIVOTED DATASET

```
delta_dict = {}
for offset in [7, 14, 21, 28]:
delta_dict[‘delta_{}’.format(offset)] = pivoted_df /
                                      pivoted_df.shift(offset) - 1
```

MELT THE SHIFTED DATASET

```
melted_dfs = []
for key, delta_df in delta_dict.items():
    melted_dfs.append( delta_df.reset_index().melt(id_vars=[‘Date’],
                                 value_name=key) )

return_df = pivoted_df.shift(-7) / pivoted_df - 1.0
melted_dfs.append( return_df.reset_index().melt(id_vars=[‘Date’],
                              value_name=‘return_7’) )
```

REDUCE-MERGE THE MELTED DATA

```
from functools import reduce

base_df = df[[‘Date’, ‘Code’, ‘Volume’, ‘VWAP’]]
feature_dfs = [base_df] + melted_dfs

abt = reduce(lambda left,right: pd.merge(left,right,on=[‘Date’,
                                         ‘Code’]), feature_dfs)
```

AGGREGATE WITH GROUP-BY

```
abt[‘month’] = abt.Date.apply(lambda x: x[:7])
gb_df = abt.groupby([‘Code’, ‘month’]).first().reset_index()
```