Jason Schlottman

HWRS 401

10/21/2021

Cheat Sheet 4 -Pandas\_Dataframes

* **Define:** Pandas:

A pandas\_dataframe (aka pandas) refers to a two-dimensional labeled data structure consisting of columns which may contain varying types (heterogeneous types) of tabular formatted data. They are extremely useful for accessing, editing, and modifying large amounts of data. Common uses include use for time series analysis, involvement in machine learning applications, etc.

\*Note= Since these dataframes are often very large, try using the head() and tail() functions to view certain sections since all the rows might not be practical or visible at once.

* Pandas dataframes are generally created using two common methods. They are either composed from scratch or accessed by reading through a 'comma-separated value' file (.csv).

To create a pandas from scratch:

In a Python script write a function composed of a variable for both the columns and rows after importing Pandas. The first variable may commonly be the function data, and the second variable may define labels, for example. The data can be passed as a list, tuple, NumPy Array, or dictionary. Other less common types exist but these are the primary important choices for our needs.

To access a pandas by reading a .csv file:

Recall there is the read\_csv() function and we can call using pandas.read\_csv() or pd.read\_csv() after we have ensured that pandas has been imported. Useful instructions related to access through reading a .csv file can be found in course materials for Week9 or at the following URL:

https://pythonbasics.org/read-csv-with-pandas/

**Returns:** vary depending on what data type was utilized, but expect to access dictionaries and form new variables many times as there may be several layers of dictionaries in the data.

**Loc, iloc, slicing:**

Very useful site for help with operations/property function: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.loc.html

Slice with labels for row, single label for the column

loc example: df.loc['a':'z', 'q']

Recall that when using the .loc property, if we call using a number this refers to the label of the index rather than an integer position of the index.