Coding Exercise #3

1. Load etfdata\_2018.sas7bdat file. (ETF data of many assets at many points in time)

2. Drop 'COMNAM', 'SHRCD' columns.

3. For every date, compute the mean return.

4. To see what the groupby is doing, use a multi-index to organize the data by ‘PERMNO’ (permanent number is the unique identifier of a given ETF) and ‘DATE’ (hint: use .set\_index()). In particular, we want the first index to be date, and the second to be the permanent number. Make sure the dates are in chronological order (hint: use .sort\_index()).

5. Create a column for the lagged return (i.e. one day return) for every ETF, and drop any null values.

6. For every date, use pandas cut or qcut function to bin the lagged returns into quintiles (5 categories, hint: going to need a groupby and an apply method). That is, we’re breaking up stocks every day into quintiles.

7. Now, for every quintile, and on every date, compute the mean returns of every stock (grouped by quintile and date), and drop any null values.

8. Note we can think of what we’re doing as creating a portfolio based on quintiles and adjusting every day. Compute some summary statistics for these portfolios (i.e. for the ETFs selected by quintile) using the .describe() method. More specifically, for every quintile, compute the summary statistics of the portfolios using .describe().

9. Compute the portfolio that every day buys some weight (you choose) of the highest quintile, and sells (shorts) some weight of the lowest quintile. When you do this, you should have a column or series of daily returns this “momentum” style portfolio. Compute the mean and standard deviation of this portfolio.