Exponentially Weighted Regression

Name: Qiqi Chen

Student Number: 12249648

1. Obtain Daily Data
   (1) I used the ticker list of EOD dataset and randomly chose 350 tickers. For tickers that have every day adjusted price from Jan 4 2016 to Dec 31 2019, I selected them into my research space. Therefore, I totally got 247 tickers and SPY end of date prices and their daily returns.
2. Exponentially Weighted Regressions
   (1) The method of exponentially weighted regressions
   Here I basically explain how to implement the exponentially weighted regressions in python. Firstly, since we are doing univariate regression to individual stock and SPY returns, we can use covariance divided by the variance of SPY returns to get beta. For the exponentially weighting parameter, I used alpha which can be calculated with our lambda input, to be the weight of the most recent observation. Therefore, it mainly gives all historical observations a weight, when the age increases one day, the weight decreases by (1-alpha).
   (2) The method of window regressions
   The implementation of window regressions is relatively simple. For different lambda, we have 2/lambda as our period to calculate the covariance and variance, then we get the regression coefficient b for each ticker at the specific four times. When lambda is larger, the window is shorter.
   (3) The parameter lambda
   According to the notes below,
   where λ has units of $1/\text{time}$, and our update formula becomes, with observation x_j at time t_j
   $$A_{t_j} = e^{-\lambda\cdot\left(t_j - t_{j-1}\right)} A_{t_{j-1}} + \left(1 - e^{-\lambda\cdot\left(t_j - t_{j-1}\right)}\right) x_j$$
   We refer to the quantity $1/\lambda$ as the *characteristic time* of our averaging, and the weight of data up to age $1/\lambda$ is of course $1/e$.
   I assigned different lambda values and then calculated $1 - e^{-\lambda}$ as alpha. As we can infer, when lambda is larger, alpha is larger, which assigns a bigger weight to the most recent observation.
3. Analysis
   (1) MAE and RMSE out of sample
   Trying to compare exponentially weighted regressions and window regressions without out of sample data is not very meaningful. Therefore, I summarized the estimation error of the regression coefficient between exponentially weighted regressions and out of sample window regressions, and window regressions and out of sample window regressions.
   For each lambda, the MAE(Mean Absolute Error) and RMSE are shown in the tables below.

Lambda = 0.05

	mae_ewm	mae_window	rmse_ewm	rmse_window
6/29/2018	2.9898	0.6956	19.0899	0.9955
12/31/2018	2.3175	0.3622	14.1231	0.3311
6/28/2019	7.3803	0.7529	181.1243	1.7569
11/29/2019	2.0614	0.7279	19.9336	1.4473

Lambda = 0.1

	mae_ewm	mae_window	rmse_ewm	rmse_window
6/29/2018	2.3946	0.8804	12.3727	1.6970
12/31/2018	2.0968	0.3911	11.6102	0.3797
6/28/2019	5.2492	1.0352	114.9907	14.5183
11/29/2019	2.0423	1.2968	18.9394	4.3864

Lambda = 0.2

	mae_ewm	mae_window	rmse_ewm	rmse_window
6/29/2018	1.8382	1.0033	7.1456	2.2378
12/31/2018	1.5879	0.4618	6.7404	0.5327
6/28/2019	3.6773	1.6196	99.8141	78.7981
11/29/2019	2.0060	1.2788	17.2595	6.0470

Lambda = 0.4

	mae_ewm	mae_window	rmse_ewm	rmse_window
6/29/2018	1.4627	0.9927	4.6157	2.1227
12/31/2018	0.9848	0.5176	2.6703	0.6382
6/28/2019	2.8617	2.3796	115.4013	251.2932
11/29/2019	1.9428	1.7002	14.9679	7.2498

Lambda = 0.5

	mae_ewm	mae_window	rmse_ewm	rmse_window
6/29/2018	1.3944	1.4567	4.2192	4.7021
12/31/2018	0.8422	0.5976	1.9826	0.9213
6/28/2019	2.7202	2.3887	122.1196	260.2992
11/29/2019	1.9184	1.7122	14.1932	7.3555

Lambda = 1.0

	mae_ewm	mae_window	rmse_ewm	rmse_window
6/29/2018	1.2555	3.9771	3.5374	33.9705
12/31/2018	0.6159	2.4061	1.0367	15.1842
6/28/2019	2.4900	12.8134	139.3826	507.4186
11/29/2019	1.8452	2.0890	12.0671	21.0087

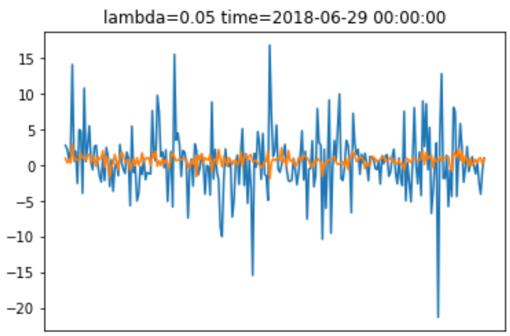
From the tables above, we can firstly see that when lambda is larger, the estimation error out of sample for exponentially weighted regressions are decreasing. I think this is because the large lambda gives large weight to the most recent observations, therefore the beta is more updated. So, the difference between it and out of sample beta is smaller.

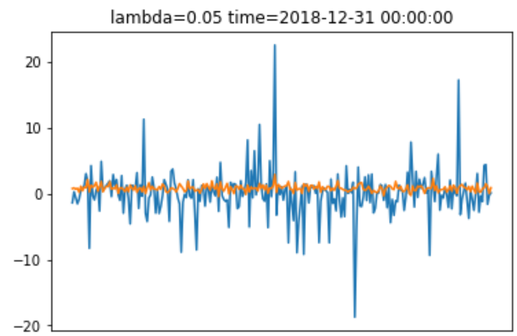
For window regressions, when lambda is larger, the window we use to run our regression is shorter and therefore the stability and reliability of beta are not quite trustable, which results in larger estimation error out of sample.

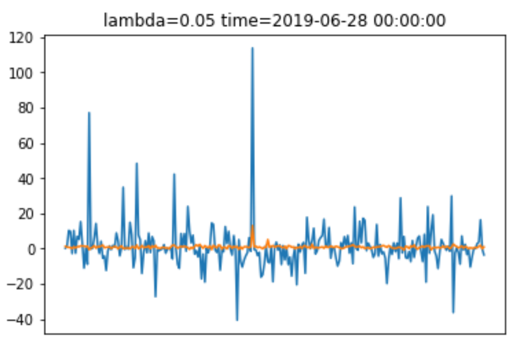
Comparing both method with same lambda, we can find the exponential method has better prediction capacity when lamda is large, and the window method has better prediction capacity when lambda is small.

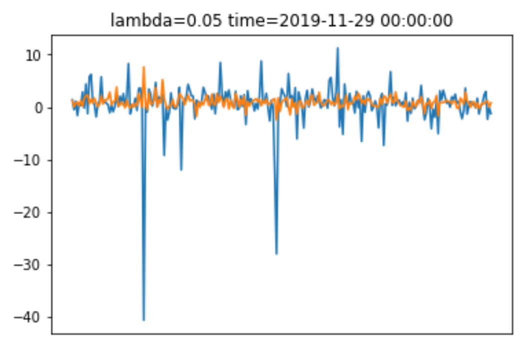
(2) Plots of betas from different methods

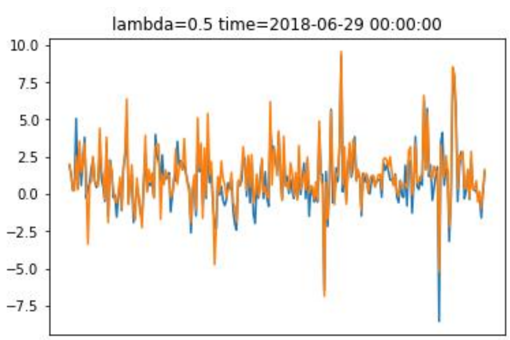
The plots are shown below, the x axis is the tickers and the y axis is the beta value.
For simplicity I only showed plots from lambda = 0.05,0.5 and 1.

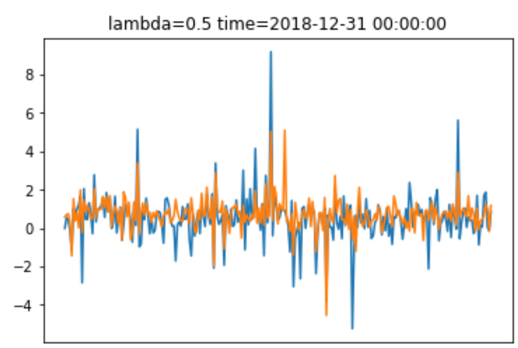


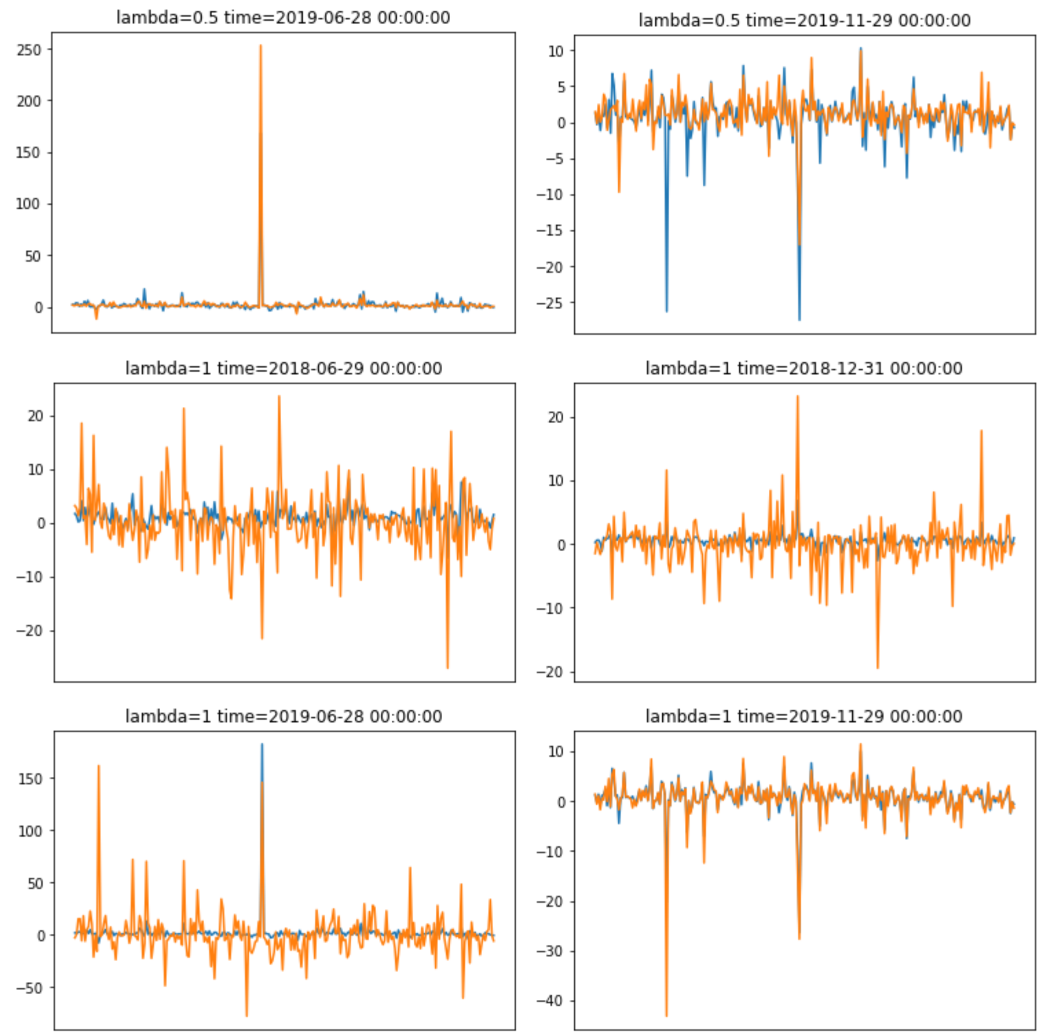












Though we didn't compare the beta with out of sample beta. We can still see that when lambda is pretty small, the estimation of beta from window regression is quite flat, which means for each ticker, the beta is smoothed by longer window. As lambda grows larger, the estimation of beta from exponentially weighted regressions are quite flat, which means for each ticker, the beta is smoothed by larger weights of recent observations.