Homework: Robust Regression And Inter-Asset Beta

May 8, 2021

1 Data

The class website contains 5 year CDS rates for debt from several companies over a multi-year range in `Liq5YCDS.delim`. Read this data, and load the corresponding adjusted close prices for the corresponding equity[1].

CDS spreads are not directly investable in the way equities are, but we can still learn a lot from treating them similar to how we treat asset prices. In particular, just like equity prices they are bounded below by zero and have no functional upper bound. Thus, we will compute weekly Wednesday to Wednesday returns r^{Equity} on the adjusted equity close prices, and similar "returns" r^{CDS} on the CDS spreads.

2 Models

Begin by forming a CDS "index return" r^{Index} as the arithmetic average of the r^{CDS}.

For each ticker $E = E_1, \ldots, E_N$, you will be working with a contemporaneous model[2] for its "spread returns" of the form

$$r_E^{\text{CDS}} \sim r_E^{\text{Equity}} + r^{\text{Index}} + \epsilon. \tag{1}$$

[1] Not all debt issuers *have* publicly traded equity. I have selected CDS rates from issuers that do.

[2] Our model is contemporaneous as opposed to predictive because we are not asking the equity returns to *predict* CDS spread "returns", but rather proxy for them.

Starting from the 17th week of available returns, define in-sample calibration data as the returns from the previous 16 weeks, and out-of-sample data as the returns in the immediately following 4 weeks. You might want to classify the out-of-sample data by their distance (1-4 weeks) from the calibration sets.

Do this both (A) with intercept forced to zero and (B) with intercept allowed to be nonzero.

3 Analysis

Estimate your linear models[3] using

- OLS
- The Huber penalty function
- The Tukey Bisquare penalty function

Make sure your fitting algorithm is setting the scale parameter to something reasonable[4].

Obtain the residuals of each regression for the out-of-sample returns. Contrast the performance of the estimators.

[3] In **R** you can use the `MASS` package. In Python try `statsmodels`.

[4] For weekly returns, it should be obvious that 1.0 is *not* reasonable.