Data Analysis
August Review 2022

Mark Hendricks
UChicago FINM

Homework #5
Regularized Regression

Data

- This homework uses the file, "`sp500_returns.xlsx`".
- Find the data in the Github repo associated with the module, (link on Canvas.)

The data file contains...

- Return rates, r_t^{GLD}, for the GLD, (an ETF,) which tracks the returns on gold.
- Return rates, r_t^i, for 451single-name equities.[1]

1 Penalized Regression

Consider a regression of GLD, denoted r^{GLD}, on all 451returns of the S&P 500 stocks.

$$r_t^{GLD} = \alpha + \sum_{j=1}^{k} \beta^j r_t^j + \epsilon_t \tag{1}$$

1. Estimate (1) with OLS.[2]
   (a) Report the estimated intercept and betas.
   (b) Report the R-squared.
   (c) Which factors have the largest betas in explaining r^{GLD}?
   (d) Calculate $\beta^j \sigma^j$ for each regressor.[3] Which of these is largest in magnitude, and thus most influential in explaining r^{GLD}?
   (e) Report the matrix condition number[4] of $R'R$, where R denotes the matrix of single-name equity return data. Why should this condition number give us pause about trusting the OLS estimates out-of-sample?

[1] These are all securities in the S&P 500 as of June 2022, filtered for names with sufficient return histories.

[2] For this OLS estimation, along with the estimations below, try using scikit-learn in Python

```
from sklearn import LinearRegression Lasso Ridge
```

For OLS, try

```
model_ols = LinearRegression().fit(X,y)
```

[3] The beta being large may simply be because the regressor volatility is small. By scaling by the volatility, we get a better idea of which regressor is driving the most variation.

[4] In Python, use numpy.linalg.cond()

2. Estimate (1) with Ridge Regression. Use a penalty of 0.5 in the estimation.[5]

   (a) Report the R-squared.

   (b) Based on $\beta^j \sigma^j$, which factor is most influential for r^{GLD}?

3. Estimate (1) with LASSO Regression. Use a penalty of 2e-5 in the estimation.[6]

   (a) Report the estimated intercept and betas.

   (b) Report the R-squared.

   (c) Based on $\beta^j \sigma^j$, which factor is most influential for r^{GLD}?

   (d) How many regressors have a non-zero beta estimates?

4. How do the estimations compare across the three methods?

   (a) Create a histogram of estimated betas across the three methods, (OLS, Ridge, LASSO.) Are they all nonzero? Are there positive and negative values? Do they range widely in magnitude?

   (b) Which has the largest R-squared? Is this a surprise?

5. Try using cross-validation (with K-folds) to estimate the penalty parameter for Ridge and LASSO.

   Estimate this CV using two functions from `sklearn.linear_model`

   - RidgeCV
   - LassoCV

   Feel free to use the default parameters, including the default number of folds.

   Report the CV penalty parameter for Lasso and Ridge.

6. Use your estimations based on data through 2020 to fit the model for 2021-2022.

   Use the CV penalty parameters (from the previous problem) for Ridge and Lasso.

   What is the r-squared in these out-of-sample fits?[7]

   Which method does better out-of-sample?

[5] Try using

```
model_ridge = Ridge(alpha=0.5).fit(X,y)
```

[6] Try using

```
model_lasso = Lasso(alpha=3e-4).fit(X,y)
```

[7] Doing this is really easy in Python. For instance, for the LASSO estimation, you could try

```
model_lasso = Lasso(alpha=3e-4).fit(X,y)
score_insamp = model_lasso.score(X_insamp,y_insamp)
score_oos = model_lasso.score(X_oos,y_oos)
```