Unsupervised Multi-factor Risk Models for US Equities

CS229 Final Project
Kyle Kelley
December 10, 2010

Abstract

This project compares equity risk models developed using the conventional supervised approach as well as a PCA-based unsupervised approach. A fair-out-of-sample comparison shows that PCA based models, which are readily developed without economic expertise, can meet and sometimes exceed the performance of industry-standard 1 and 3-factor supervised models.

Motivation

Many successful financial traders will argue that risk management is the most important long-run aspect of trading, as it's primary goal is to avoid catastrophic losses. Note that there is no universally agreed-upon measure of a portfolio's risk, but some commonly-used examples are variance, Value-at-Risk (VaR), and worst-case loss. This project only addresses variance-based models. To be useful (and successful), these risk models must somehow accurately *predict* future variances, which (as we will see) is very difficult.

If a portfolio contains *n* stocks, then the covariance matrix among the stock returns mathematically describes the variance of portfolio returns through the formula

$\sigma_p^2 = x^T \Sigma x$, where

$\Sigma \in \mathbb{R}^{(n \times n)}$ is the covariance matrix among the stocks

$x \in \mathbb{R}^n$ is the vector of stock holding weights in the portfolio

Hence, we need to have reliable predictions of the covariance matrix to accurately assess portfolio risk. A reasonable approach is to assume the historical covariance equals the future covariance, implying that we would need to accurately estimate $\frac{(n+1) \cdot n}{2}$ different parameters, which is extremely difficult (both from standpoints of computation and accuracy) to do in practice.

Arbitrage Pricing Theory (APT) [1] postulates that the returns of each stock are driven by a linear combination of a small set of *k* underlying factors:

$r_s = \beta r_f + \alpha$, where

$\beta \in \mathbb{R}^{n \times k}$ is a matrix of factor exposures for each stock

$r_f \in \mathbb{R}^k$ is a vector of factor returns

$\alpha \in \mathbb{R}^n$ is a vector of idiosyncratic returns (uncorrelated to the other factors)

Assuming APT holds, then we can compute the covariance as

$\Sigma = \beta \Sigma_f \beta^T + \Delta$, where

$\Sigma_f \in \mathbb{R}^{k \times k}$ is a covariance matrix among the factor returns

$\Delta \in \mathbb{R}^{n \times n}$ is the covariance matrix among the idiosyncratic returns

If we assume that the idiosyncratic returns are uncorrelated, the idiosyncratic covariance becomes diagonal, which means we now only need to estimate $n \cdot k + \frac{(k+1) \cdot k}{2} + n$ parameters, which is

1 For more info see *Active Portfolio Management* by Grinold & Kahn

much more feasible when $k \ll n$.

The problem with the APT approach is that it doesn't specify what the factors are, and they must somehow be determined either empirically or through theory. In practice, teams of highly-trained economic analysts work on determining the factors they think drive equity returns, and then determine exposure forecasts that each stock has to these factors. Some examples of such factor models are the 1-factor market model, the 3-factor Fama/French model, and the 68 factor model sold by MSCI/Barra. All of these industry-standard supervised approaches are very time-consuming to build (and/or expensive to buy), which leads us to wonder how effective an unsupervised approach can be in a side-by-side comparison.

Furthermore, the general difficultly of estimating covariance with a factor-based APT approach, coupled with the difficulty in dealing with an *n*-x-*n* matrix in portfolio optimization, often leads portfolio managers to ignore covariance altogether and instead focus on variances (e.g., using a diagonal covariance matrix). This approach is briefly addressed in the results.

Figure 1 shows the magnitudes of the principal components for the daily returns of the S&P500 for 2006, providing some motivation for a PCA-based approach. Note that the first principal component explains nearly 3 times the variance as the second principal component, suggesting why historically 1-factor (market) risk models have been used effectively.

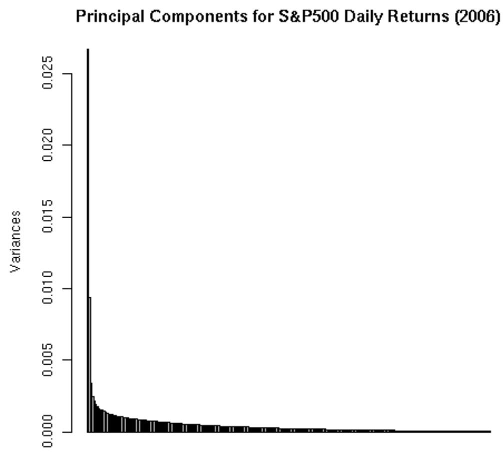

Figure 1: Magnitudes of Principal Components for Daily S&P500 Returns (2006)

One obvious problem with PCA is that the factors produced don't necessarily have any physical meaning, and hence we have less confidence they will continue to have any relevance in the future. This experiment indirectly measures this.

Methodology

General

The goal of this project is to use the historical returns of *n* stocks measured at time-intervals of length *d* to predict the covariances of these stocks in future time periods. We define the return of a stock at time *t* to be $r_t = \log(\frac{p_t^{close}}{p_{t-d}^{close}})$, where p_t^{close} is the closing price at time *t*.

We indicate the time a model is constructed as t_{model} . We then form $r_{train} \in \mathbb{R}^{n \times m_{train}}$ with the returns of *n* stocks in m_{train} prior consecutive time periods (i.e., at times $t_{model} - m_{train} d, t_{model} - (m_{train} - 1) d, \ldots, t_{model} - d$) to predict the covariance matrix of returns over m_{test} consecutive future time periods $\Sigma_{test} \in \mathbb{R}^{nxn}$ (i.e., using times $t_{model}, t_{model} + d, \ldots, t_{model} + m_{train} d$).

Weights

As an additional tuning knob, we use a decaying exponential to weight our historical observations, giving more weight to more recent observations. The weight for a return at time *t* is $w_t = 2^{-\frac{t_{model} - t}{d m_{hl}}}$ where m_{hl} is the tunable half-life parameter.

Idiosyncratic Variance

As mentioned we will assume that idiosyncratic variances are uncorrelated. Hence, we can account for it simply by ensuring that the diagonals of our final covariance matrix equal the predicted variance of each stock. $\sigma_s^2 = \sigma^2 - diag(\Sigma_{test})$, where $\sigma_s^2 \in \mathbb{R}^n$ is the vector of idiosyncratic stock variances, $\sigma^2 \in \mathbb{R}^n$ is the vector of overall stock variances.

There are a number of sophisticated autoregressive techniques to forecast variance, but these are beyond the scope of this project. Instead, we use historical exponentially weighted variance as a predictor of future variance. Note that any improvements to this algorithm will improve the results of all methods equally.

Error Metric

To evaluate models, we compare the observed covariance matrix over m_{test} time periods, Σ_{actual} , to the trained covariance matrix for the same time periods, Σ_{test} . Our error metric is the Frobenius norm of the difference: $e_{test} = \|\Sigma_{actual} - \Sigma_{test}\|_F$.

Tuning

The set of tunable parameters are optimized by iterating the following cross-validation procedure on each parameter until all the parameters have reasonably converged. In practice, this never took more than 2 iterations.

Cross-validation

We use a rolling cross validation in favor of a typical k-fold cross-validation to avoid any *future bias* in our models (which is crucial to avoid with financial data). Given a full dataset of returns over m_{total} time periods, we can create $m_{models} = m_{total} - m_{test} - m_{train}$ different models with this data, and estimate the generalization error as the mean of the test errors:

$$e_{gen} \approx \overline{e_{test}} = \frac{1}{m_{models}} \sum_{i=1}^{m_{models}} e_{test}^i \quad .$$

By performing this procedure over a sweep of parameter values, we select the optimal parameter as the one that results in the lowest measured $\overline{e_{test}}$.

Unsupervised (PCA)

Using PCA, we estimate the covariance matrix as $\Sigma_{test} = \beta_u diag(\lambda) \beta_u^T + diag(\sigma_s^2)$, where $\beta_u \in \mathbb{R}^{(nxk)}$ are the *k* first principal components of the centered/scaled returns data r_{train} , and

$\lambda \in \mathbb{R}^k$ are the k corresponding eigenvalues. Note that this method introduces k as another tunable parameter: if the principal components estimated from historical data remain accurate in the future, then this method will yield good results. The exponential weighting is used only for estimating stock-specific variances (as already mentioned).

Supervised

The supervised approach uses the same returns matrix r_{train} as well as a returns matrix for k factors over the same time intervals, $r_f \in \mathbb{R}^{k \times m_{train}}$. We then use a weighted linear regression (same exponential weights as before) to estimate $\beta \in \mathbb{R}^{(nxk)}$ so that $r_{train} \approx \beta f$. Additionally, we compute the weighted covariance among the factors to estimate Σ_f . The stock covariance matrix can then be estimated as $\Sigma_{test} = \beta \Sigma_f \beta^T + diag(\sigma_s^2)$.

Results

The described methodology was used on data for stocks in the S&P500. All parameters were tuned using data from 2006-2009, and then the tuned models were further tested on data from January 1, 2010 to November 30, 2010. Table 1 describes the different models being compared:

Table 1: Description of Models Tested

Model	Description
IVAR	Assumes ideal variance forecasts and 0 covariance between stocks.
SPY	1-Factor supervised model (market)
FF	3-Factor Fama/French supervised model (market, size, value)
PCA	Unsupervised PCA model

This methodology was used to build models on 2 different time horizons: one using daily data to predict covariance over 22 intervals (1 month), and one using 30-minute bar data to predict covariance over 26 intervals (2 days). The final tuning results and mean errors are reported in Tables 2 and 3. The time-series of test errors between PCA and FF for the 22 day test are shown in Figure 2.

Table 2: Results for d=1 day, m_{test}=22 (1 month), m_{train} = 252 (1 year)

Model	m_{hl}	k	$\overline{e_{test}}$ 2006-2009	$\overline{e_{test}}$ 2010
IVAR	NA	NA	0.3776	0.1106
SPY	12	1	0.2764	0.0742
FF	15	3	0.2766	0.0735
PCA	8	1	0.2702	0.0719

The most important tuning parameter to determine in all cases was m_{hl}. An example sweep is shown in Figure 3; note the minimum occurs at m_{hl}=8.

The results from tuning k (in PCA models) and tuning m_{test} (in all models) were less exciting but still interesting because they provide good guidelines when working with risk models. In all cases PCA models with k=1 were optimal. Also, once m_{hl} had been tuned, models with larger m_{train} had lower test error, suggesting the best method in practice is to use the largest possible window of historical data when building risk models.

Table 3: Results for d=30 minutes, m_{test}=26 (2 days), m_{train} = 1638 (6 months)

Model	m_{hl}	k	$\overline{e}_{test}$ 2006-2009	$\overline{e}_{test}$ 2010
IVAR	NA	NA	0.0290	0.0109
SPY	60	1	0.0223	0.0087
FF[2]	NA	3	NA	NA
PCA	30	1	0.0226	0.0087

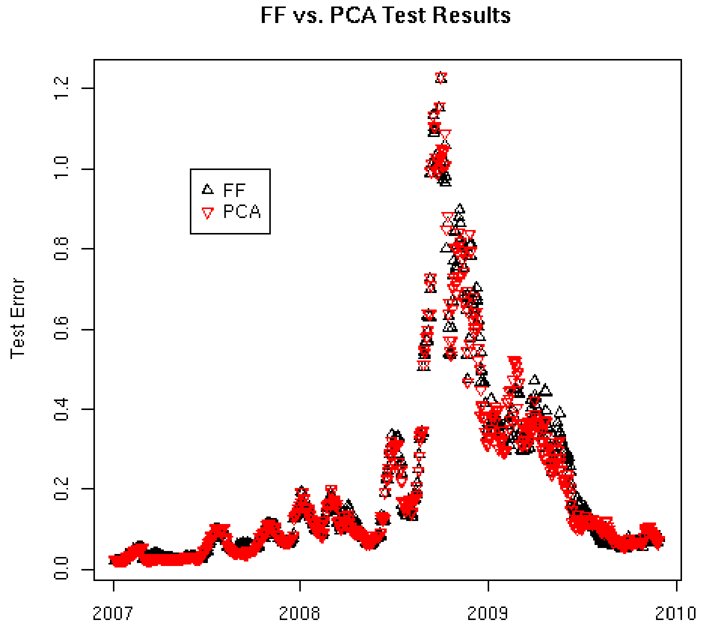

Figure 2: Time-series comparison of PCA and FF test errors

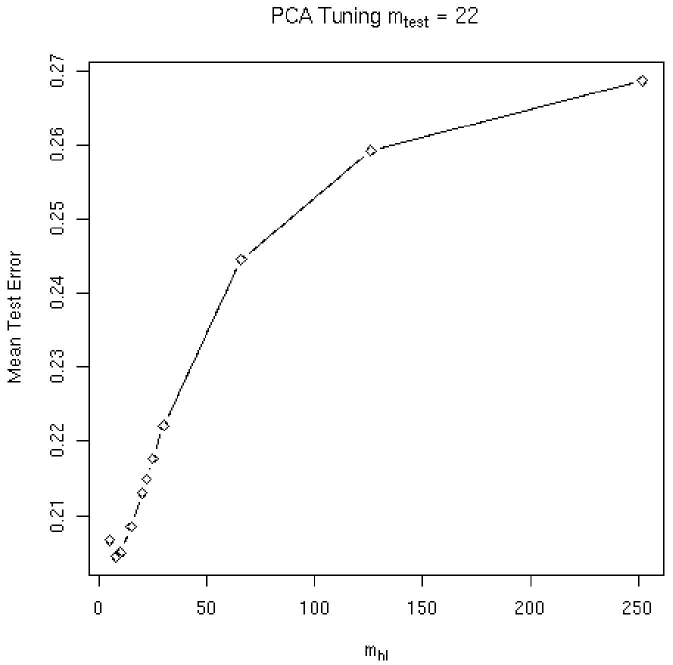

Figure 3: Tuning m_hl for PCA

2 This test wasn't performed due to a lack of intraday data for the FF factors

Conclusions

There are a number of interesting takeaways from these experiments. Primarily, it was demonstrated that, when tuned properly, unsupervised PCA-based risk modeling can meet and sometimes outperform a 1-factor SPY model and a 3-factor Fama-French model (tested over both a 1-month horizon and a 2-day horizon). It would be interesting to compare these results to the 68 factor BARRA model in same framework.

Another interesting result is that PCA always had the smallest error with only 1 factor. In literature (and in some commercial models[3]) there are often 10-20 factors retained from PCA, but these out-of-sample tests demonstrated this is suboptimal on average (at least for the horizons tested here).

Moreover, comparing the tests against the performance of IVAR demonstrates that covariance modeling is not a futile effort: in practice it is possible to improve substantially upon a perfect variance-only model.

On a final note, the prudent reader should recognize that historical-based risk models clearly have their limitations, as they don't generally predict situations that have never occurred. This is most apparent in the plot of test errors above during the period of Fall 2008 (see Figure 2), when all models performed far worse than their historical average. This project merely showed that PCA rivaled the alternatives (SPY/FF), not that it always did well on an absolute basis.

3 One such example is Northfield Short-Term Equity Risk Model, which keeps 20 PCA factors http://www.northinfo.com/documents/5.pdf