Econometrics II January 2023

Core European Stock Portfolio

Alexandre Poirier

Data Manipulation: We first drop the first column as it contains the “market benchmark index”. In addition, I can see some abnormal values on the 31/12/2021 when plotting the returns. Therefore, I decided to delete the row at this period.

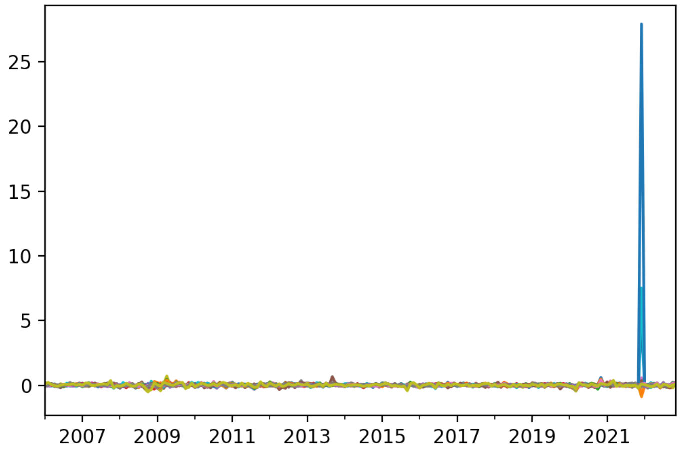


Chart 1: Returns before Data Processing

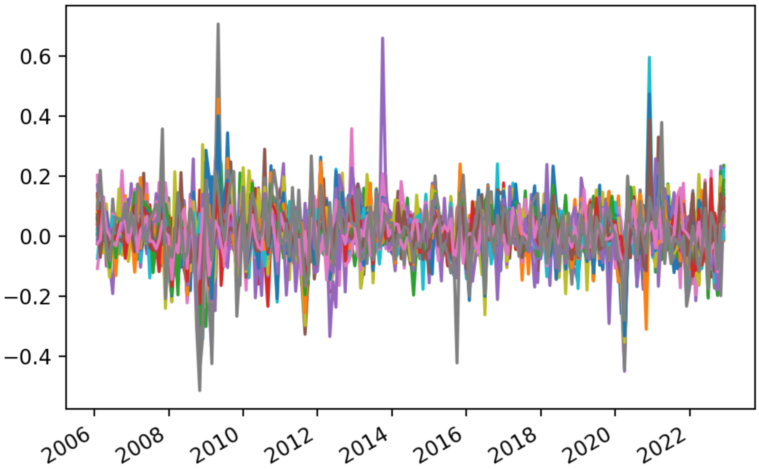


Chart 2: Returns after Data Processing and scaling

Part 1) Extraction of Principal Components and determination of the main factors

- Principal component analysis (PCA) is a statistical technique that is used to reduce the dimensionality of a dataset. It does this by transforming the original variables into a new set of variables, called principal components, which are uncorrelated and capture the maximum amount of variance in the data. The first principal component captures the maximum variance, the second principal component captures the second-maximum variance, and so on.

- PCA is performed on the covariance matrix of the normalized returns by using the “statsmodel” library in Python. After extracting each principal components, we can plot the graphs below:

- The first component captures more than 45% of the variance of the data set and has a correlation of 0.98 with all our stocks whilst the other PC have much less explanatory power. We could say that the first PC represent the market risk.

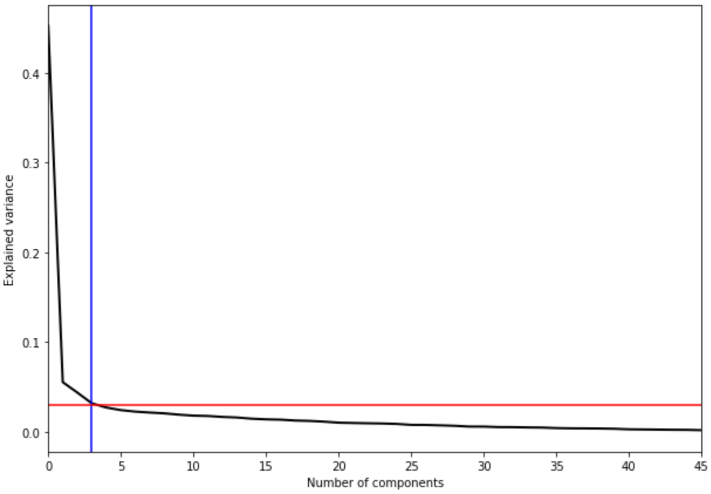


Chart 3: Explained Variance plot

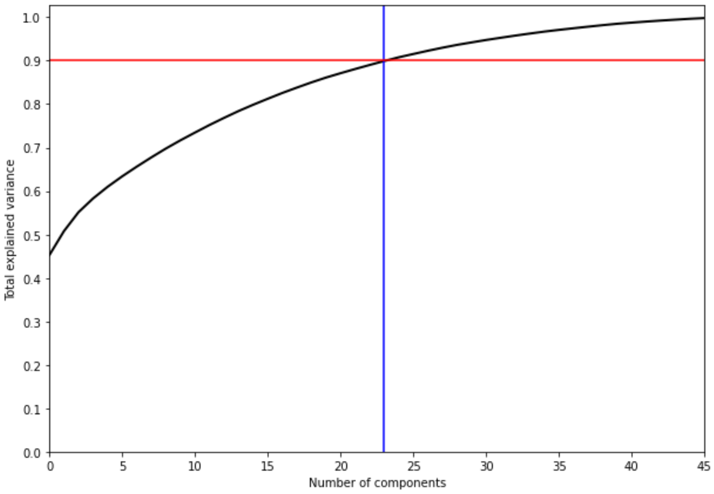
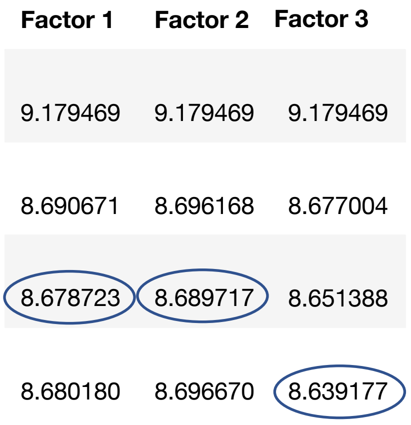


Chart 4: Cumulative Explained Variance plot



Factor 1	Factor 2	Factor 3
9.179469	9.179469	9.179469
8.690671	8.696168	8.677004
8.678723	8.689717	8.651388
8.680180	8.696670	8.639177

Chart 5: Bai and NG IC Criteria Selection

Determination of the main factors

- In order to determine the optimal number of PC we can use different criteria: according to the "scree plot" criteria we could choose 3 PC whilst according to the "90% variance threshold" criteria we would have to include 23 PC in our model.

- A more rigorous approach would be to rely on the Bai and Ng IC criteria. They allow to assess the "goodness of fit" of the PC decomposition whilst still penalizing the number of PC in the model.

- 2 out of 3 criteria are minised when we select 3 principal components, we therefore select the first 3 PC as the "optimal" number of PC for the European Equity Factors.

Part 2) Identification of the k significant factors and estimation of their associated exposure from a linear model.

- In order to identify, for each stock, the most significant factors among the 3 PC we will be using Lasso Regression (L1).

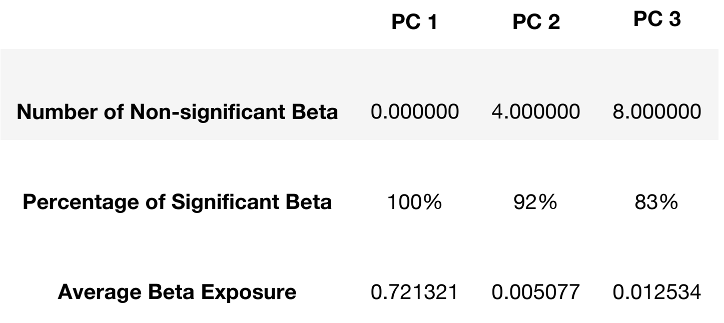
- Lasso regression is a type of linear regression that uses regularization to reduce the complexity of the model by shrinking the coefficients of the features towards zero. Lasso regression has the following advantages for factor selection in models.

- Lasso regression can automatically select features by setting the coefficients of the unimportant features to zero. We are using this to our advantage here as it will provide a more interpretable and optimal model compared to other regression methods. Null Beta = no exposure to the factor.

- We use cross-validation approach to determine the optimal regularization parameter. Cross-validation is a resampling procedure used to evaluate the performance of machine learning models. Cross-validation is a useful tool for selecting the regularization parameter in lasso regression because it helps to prevent overfitting and allows the model to generalize better to new data. It can also help to reduce the variance of the model, which can improve the stability and reliability of the model.

- Please note that the eigenvector of PC1 is negative. We consider -F1 instead of F1 to obtain a positive correlation between the first factor and the return of the individual stock.

- We can see below that all the coefficients of the first PC are kept while it is decreasing with the second and third pc. Which is what is expected when using Principal components as our betas.



	PC 1	PC 2	PC 3
Number of Non-significant Beta	0.000000	4.000000	8.000000
Percentage of Significant Beta	100%	92%	83%
Average Beta Exposure	0.721321	0.005077	0.012534

Chart 6: Summary of Statistics of the Lasso Betas for each PC

Part 3) weights of the K equity portfolios designed to replicate the K core equity factors

- We verify that $\sum^{N} w_{i=1\ k,i} = 1$ and $\sum_{i=1} w_{k,i} b_{k,i} = 1$ and $1 \geq w_{k,i} \geq -1$ in the conditions of our optimization function, it is the case here, a quick check is also done in the code.
- We obtain the following long/short equity replication portfolios:

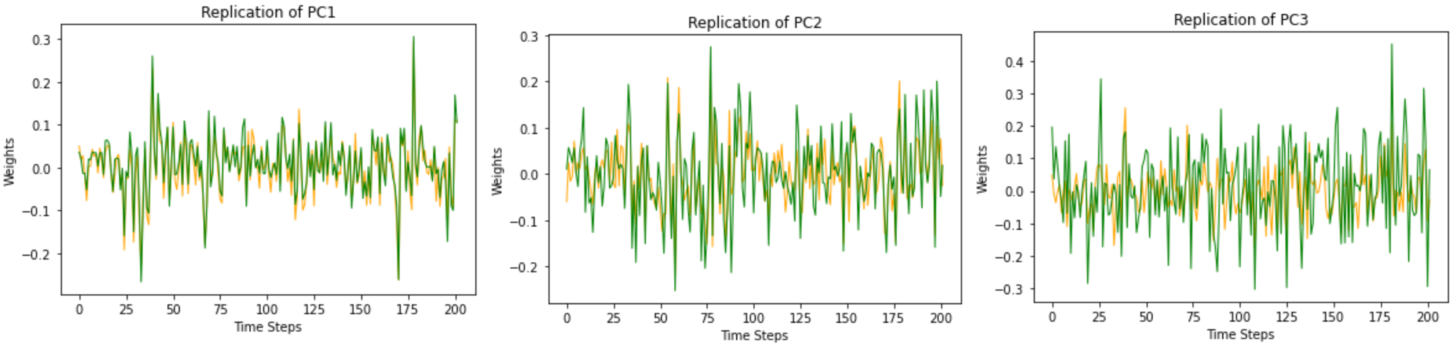


Chart 7: Replication portfolios vs Selected PC

- We note that the replication quality is decreasing with the PC level and PC exibits higher variance with increasing PC.

	Replication PC 1	Replication PC 2	Replication PC 3
Max Value	-0.266986	-0.251757	-0.302790
Min Value	0.305933	0.274599	0.451807
Variance	0.005036	0.007846	0.016531

Chart 8: Summary of Statistics of Replication Portfolios

Part 4) Estimation of the time-varying alpha $\hat{\alpha t}$ using Kalman Filter State Space Model

- The Kalman filter can be used to estimate the time-varying alpha ($\hat{\alpha t}$) of a process in the presence of noise. It is an algorithm that uses a series of measurements observed over time to estimate the state of a system. This algorithm consists of two steps: the prediction step and the update step. In the prediction step, the Kalman filter uses the system model to predict the state of the system at the next time step. In the update step, the Kalman filter compares the predicted state with the measured state, and updates the prediction based on the error between the two.

- Signal Equation: Y_t = X_t * B_t + e_t

  * e_t → N(0,1)

- State Equation: B_t = A * B_t-1 + C * U_t + u_t

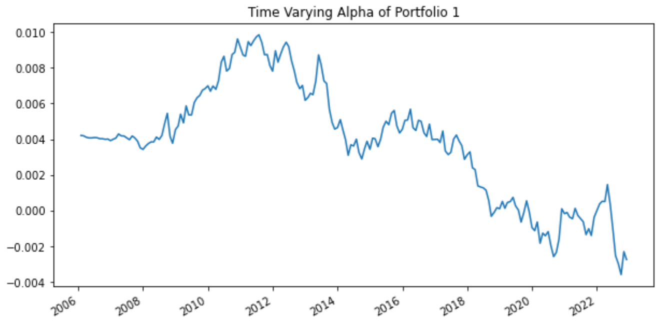
  * u_t →N(0 , Q)

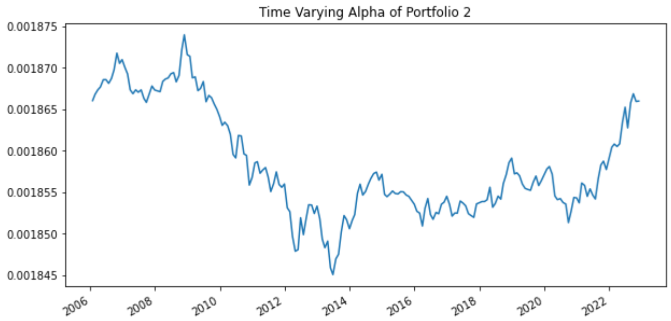
- We identify the following components:

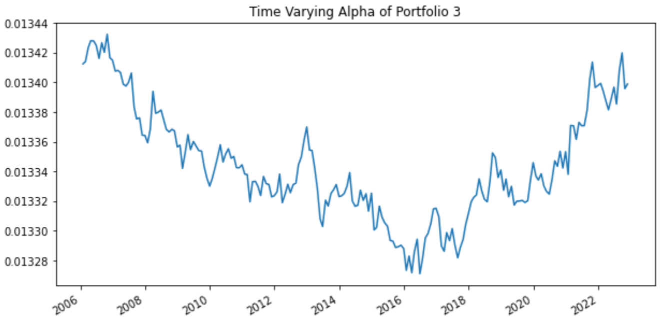
  B_t = (α_t β_i)’ X_t = (1 r_m,t) A = ((1,0)(0,1))

  C=(0,0)’ U_t = 0 Q = ((σ_αt 0)(0 0))

- Running the Kalman Optimization program we obtain the following time varying alphas:





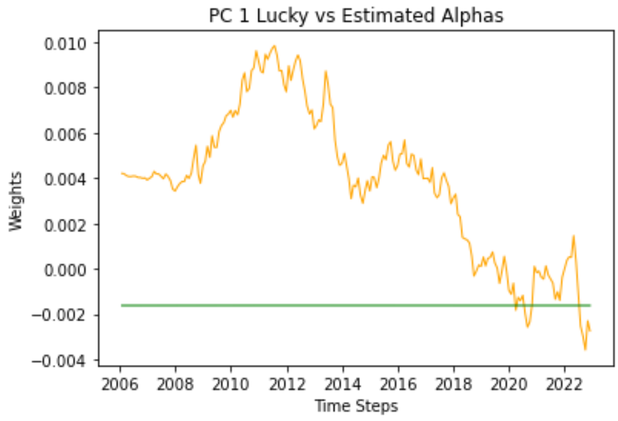


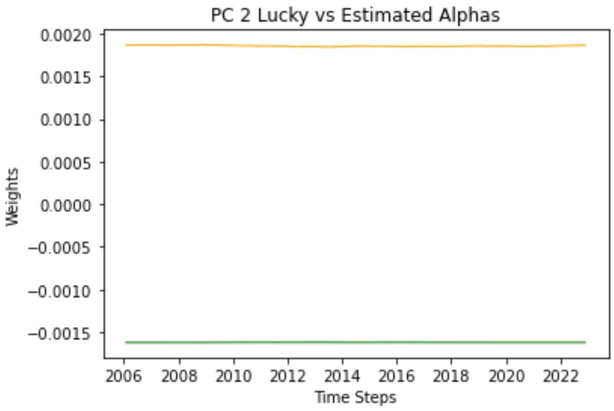
Part 5) Comparing the distribution of the estimated time-varying alphas ($\hat{\alpha}t$), against the distribution of “lucky time-varying alphas”

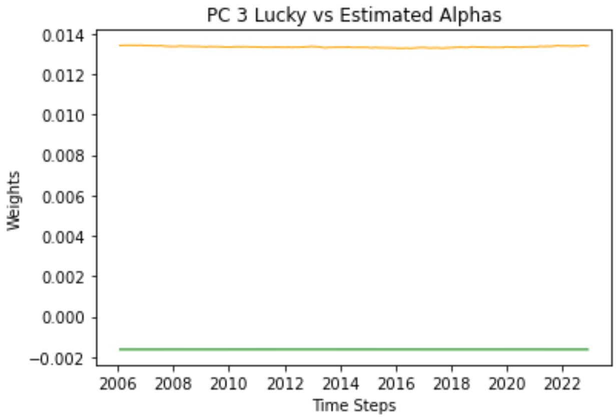
- Bootstrap can be used to compute "lucky" alpha, which is a measure of the significance of a stock's alpha (return above market return). We first run an OLS regression on our replication portfolios vs the market benchmark and extract the residuals of our model.

- We then perform a bootstrap simulation on the residuals before recreating bootstrapped returns values. This statistical method is used to estimate the precision of statistical estimates by resampling the data multiple times with replacement. It allows for the estimation of sampling error and the generation of confidence intervals without the assumption of a known population distribution.

- We now use the Kalman Filter on our bootstrapped return so that we retrieve the “lucky alphas”, allowing us to compute the rolling window alpha. Here are the plots of the “Lucky alphas vs Estimated Alphas”:







- As we see above the "Lucky alphas" are not significant. We can confirm it by performing a Kolmogorov Smirnov Test on the two distributions. The Kolmogorov-Smirnov (KS) test is a non-parametric statistical test used to determine whether two samples come from the same population. It compares the cumulative distribution function (CDF) of the two samples and calculates a test statistic, called the KS statistic, which measures the maximum difference between the two CDFs.

	Statistic	P-value
KS Test - PC1	0.96	0.00
KS Test – PC2	1.00	0.00
KS Test – PC3	1.00	0.00

- The KS statistic is used to calculate a p-value, which represents the probability that the observed difference in the CDFs between the two samples would occur by chance if the two samples were from the same population. If the p-value is less than a specified threshold, such as 0.05, it is concluded that the two samples are likely not from the same population. Here we reject the null hypothesis that αt = 0 and conclude our estimated alphas are not "lucky", i.e. they are significant.