Essays on Fitting Factor Models for Asset Returns

Sangeetha Srinivasan

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2018

Reading Committee:

Eric Zivot, Chair

Douglas Martin

Yu-chin Chen

Program Authorized to Offer Degree:

Department of Economics

University of Washington

Abstract

Essays on Fitting Factor Models for Asset Returns

Sangeetha Srinivasan

Chair of the Supervisory Committee:
Professor Eric Zivot
Department of Economics

Factor models are used to describe the fundamental drivers of financial asset returns. There are 3 types: time-series factor, statistical factor and fundamental factor models. While factor models have existed for almost 60 years, industry-wide adoption with factor-based investing has surged in the last decade. This dissertation is centered on *factorAnalytics,* an open source R package co-developed with other UW students and faculty members, that demystifies the industry black-box models, making model fitting tools readily available for any interested academic or practitioner.

Chapter 1 compares the characteristics of the three types of models in terms of model specification, estimation, interpretation and various in-sample and out-of-sample performance metrics using S&P 500 stock returns. Like Connor (1995), we find that the fundamental factor model outperforms the time-series and statistical factor models since it makes use of additional

information on asset-specific characteristics. Moreover, we find that adding statistical factor(s) extracted from the residuals of time-series or fundamental factor models, or, fitting fundamental factors to the residuals of a time-series factor model, to create hybrid models, further improves performance.

Investment management firms need to understand peer positioning for a variety of reasons, including risk management. Factor models provide a framework to estimate peer exposures, especially useful when holdings-based information is lacking. Chapter 2 presents a multi-asset time-series factor model constructed from long-short portfolios of asset class index returns, applied to peer-average returns from the Morningstar U.S. fund allocation categories. We show that factors are better than asset classes for assessing unknown exposures and decomposing risk in multi-asset portfolios. Furthermore, there is an opportunity to create more efficient, better risk-diversified portfolios using factors when making allocation decisions. We use the multi-factor model to construct equal-asset-risk and equal-factor-risk portfolios and compare them to the equal-weighted and minimum-variance portfolios. We also show that a zero-investment equal-factor-risk portfolio sleeve helps bridge the gap between pure risk parity and traditional portfolios, enhancing Sharpe ratio across all risk categories.

Chapters 3-5 contain vignettes for each type of factor model that describe and demonstrate model fitting, factor risk (volatility, value-at-risk and expected shortfall) decomposition, and related S3 generic methods.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I sincerely thank my advisor, Eric Zivot, for his guidance, inspiration and support. I'm grateful to my committee members Doug Martin, Yu-chin Chen and Michael Rosenthal for their inputs. I'd also like to express my gratitude to Sam Pittman for mentoring me on asset allocation and the retail business and enabling real-world applications of factor models.

A big thanks to the UW Computational Finance & Risk Management (CFRM) program, founded by Doug Martin and Eric Zivot, for inspiring and enabling my interest and specialization in this field. I thank Google Summer of Code (2014 and 2015) for supporting open source code development for the R package "factorAnalytics". I'm grateful to all the UW faculty (Doug Martin, Eric Zivot & Guy Yollin) and students (Yi-an Chen, Lingjie Yi, Avinash Acharya & Chindanai Uthaisaid) who have co-developed the package over the years. I'd also like to acknowledge the financial support from UW Economics and UW CFRM via teaching assistantships, and educational assistance from Russell Investments over the years.

A special thanks to my family and friends – you know who you are – for your prayers, encouragement, wise counsel and steadfast belief that I would succeed.

DEDICATION

To my parents, Bhuvana and Srinivasan.

Chapter 1. PERFORMANCE COMPARISON OF FACTOR MODELS BASED ON U.S. EQUITY RETURNS

1.1 INTRODUCTION

Just as food labels provide a breakdown of food into essential nutrients such as protein, carbohydrates, fat, water, vitamins, minerals etc., risk factors (or simply, factors) help investors understand the fundamental sources of financial asset returns, reducing the dimension of the problem from thousands of assets to less than 100 factors. Put differently, investors are identifying the common, systematic sources of risk that they are compensated for holding because they can't be diversified away by forming a portfolio of a large number of assets. The factors are usually combined linearly[1] to form factor models, where the exposure to a factor (also known as factor loadings) typically varies across assets and over time. Grinold & Kahn (2000) and Zivot & Wang (2007) describe the different types of models and the econometric methods used in their estimation. There are three main types of factor models: time-series factor models, statistical factor models and fundamental factor models.

Which is the most appropriate factor model to use? Gronold & Kahn (2002) refer to the art of building multi-factor risk models. Here, we attempt to shed some light on this topic by comparing the characteristics of the three types of models in terms of model specification, estimation and interpretation, as well as various in-sample and out-of-sample performance metrics when applied to S&P 500 historical stock returns.

[1] Non-linear factor model specifications are beyond the scope of this paper, For reference, well-known examples in the literature include Bansal et al. (1993), Bansal & Viswanathan (1993), Dittmar (2002), etc.

In *time series factor models*[2], factor realizations are observable time series, such as, broad market index returns, inflation surprises, output gap, interest rate changes, oil price shocks, etc. and the factor loadings, are estimated via a time series regression. A well-known example is the capital asset pricing model (CAPM) by Sharpe (1964) and Lintner (1965), where the excess return of the market over the risk-free asset is the common factor capturing a single economy-wide risk. Chen et al. (1986) use a multi-factor framework to find that expected and unexpected inflation, the spread between long and short-term interest rates (term spread) and between high- and low-grade bonds (credit spread) have significant explanatory power.

In *statistical factor models*, factor returns are unobserved and statistical methods such as principal components analysis (PCA), asymptotic principal components analysis (APCA) or factor analysis are used to identify the pervasive factors in asset returns, and the factor loadings are estimated using a time series regression. For example, PCA uses the eigen decomposition of the $N \times N$ covariance (or correlation) matrix of asset returns to find the first K principal components that explain the largest portion of the sample covariance (or, correlation) matrix of returns, where N is the number of assets and K is the desired number of factors[3].

A *fundamental factor model* uses observed cross-sectional data on fundamental characteristics of each company (such as industry or sector classification, earnings yield, market capitalization, price volatility etc.) to determine common risk factors. The factor exposures are observed characteristics, while the factor returns are the coefficients from a cross-sectional regression at each period, resulting in a time series of estimated factor returns. The heteroscedasticity of residual

[2] These are also known as macro-economic factor models when factors are based on "pure" macro-economic variables. We use the more generic name, time-series factor models, to include financial variables or returns of specific portfolios such as the market (Sharpe, 1964), size and value (Fama & French, 1992, 1993).

[3] For PCA, if the required number of factors is not known, any number of factors less than N can be used. For APCA, statistical methods such as Connor & Korajczyk (1993) and Bai & Ng (2002) are available to determine the optimal number of factors.

variances across assets makes the least squares estimate of factor returns inefficient and usually a two-step weighted least squares regression is performed using market cap or the estimated residual variances from the first step as weights. This method was first introduced in Rosenberg & Marathe (1976) and is referred to as the “BARRA” approach, after Barr Rosenberg who founded the investment analytics firm BARRA, Inc., now a part of MSCI, Inc.

The distinction between the different types of models is ambiguous in some cases. Fama & French (1992) derive a time series of returns for the size and value factors using a two-step process. They split stocks into groups based on observable metrics for size (market capitalization) and value (book-to-market ratio). The difference between the average returns on the small- and big-stock portfolios (SMB) and high- and low-value stock portfolios (HML) is a proxy for size and value factors respectively. The factor exposures are then estimated via a time series regression. The Fama-French approach can be considered a fundamental factor model since it uses firm-specific characteristics in the first step, or as a time series factor model since the factor realizations are observed returns of specific portfolios and the exposures are estimated via time series regression. For the purposes of this research, we classify the Fama-French approach under time series factor models.

In comparing the three basic types of factor models, we observe some advantages and disadvantages. Time-series factor models have the benefit of an intuitive interpretation of factors, publicly available data and direct application to stress testing macro events. On the other hand, time series models are susceptible to overfitting and spurious exposures, while overall explanatory power tends to be low due to omitted firm-specific factors. Statistical factor models are least data intensive and can detect unknown factors from asset returns. However, to interpret the statistical factors, we need to map them to other financial or macro-economic variables via factor rotation or correlation analysis. In some cases, higher frequency return data is required to capture some of the more transient firm-specific risk factors such as momentum (Miller, 2006).

Connor (1995) compares the explanatory power of the three types of models for U.S. equity returns and finds that the fundamental and statistical factor models outperform macroeconomic factor models, while the fundamental factor model slightly outperforms the statistical factor model as well. Though by definition the statistical factor model aims to maximize explanatory power, the fundamental factor model has the advantage of using a much larger dataset with additional explanatory variables, compared to a statistical model only using asset returns.

Given the pros and cons of the three primary types of factor models, researchers and practitioners have made use of different kinds of hybrid factor models. Herskovic et al. (2016) extract a common factor called common idiosyncratic volatility (CIV) from the variance of the idiosyncratic or residual return from other models such as the Fama & French (1993) 3-factor model among others to find that it adds significant explanatory power. Common idiosyncratic risk in cash flows driven by firm-level productivity and demand shocks are a proxy for the consumption risk faced by households and investors that they cannot perfectly hedge. Menchero (2008) adds a statistical factor to a fundamental factor model and shows that, in the context of the hybrid model, the additional factor can be interpreted as an omitted factor that’s orthogonal to the exiting fundamental factors. Maio & Philip (2015) extract 4 principal components from a large panel of 107 macroeconomic variables creating a hybrid between statistical and macroeconomic factor models. Northfield’s “Everything Everywhere” (2013) global equity model adds 5 statistical factors to other fundamental and macroeconomic factors for a total of 90 factors. Another industry example is Axioma’s U.S. equity model by Brown (2014) that adds fundamental factors such as size, value and sectors to a macroeconomic factor model.

Perhaps the answer isn't just any one type of factor model. Axioma's research report by Brown & Canova (2011) makes a compelling argument for using multiple risk models across different time horizons. Other aspects for innovation include time-varying coefficients (to account for regime changes), non-linear specifications (to capture interacting and quadratic risk factors), eigen adjusted covariance matrices (to improves the bias statistics), etc.

In this paper, we provide an update on Connor's research using a recent dataset of 122 U.S. stocks and various in-sample and out-of-sample metrics for performance measurement. The fundamental factor model outperforms the others since it makes use of additional information on asset-specific characteristics and is not as susceptible to overfitting or systematically under-predicting volatility. Additionally, adding statistical factors extracted from the residuals of time-series or fundamental factor models to form a hybrid model improves their performance. Section 2 outlines the econometric model specification. Section 3 and 4 describe the data and methodology used for model fitting and performance comparison. Section 4 presents the results and Section 5 concludes.

1.2 Factor Model Specification

The general form of a multi-factor model is given by

$$r_{i,t} = \alpha_{i,t} + \beta_{1,i}^{t}\, f_{1,t} + \beta_{2,i}^{t}\, f_{2,t} + \cdots + \beta_{K,i}^{t}\, f_{K,t} + \varepsilon_{i,t} \tag{1.1}$$

where, $r_{i,t}$ is the return on asset i at time t, $f_{k,t}$ is the return on factor k at time t, $\beta_{1,i}^{t}$ is the factor exposure for asset i on the k^{th} factor at time t and $\varepsilon_{i,t}$ is the idiosyncratic or asset-specific residual return, for all $i = 1, 2, \dots, N$ assets, $k = 1, 2, \dots, K$ factors over $t = 1, 2, \dots, T$ periods. Typically, real returns or excess returns over a risk-free asset or other benchmark are used. It is assumed that the residuals have zero mean, are uncorrelated with the factor returns across all factors and time periods for all assets, as well as with each other (serially across time and contemporaneously across assets). It is also assumed that the factor returns are covariance stationary.

In time-series and statistical factor models, the factor exposures (and intercept) are estimated via a time series regression, and unless time-varying coefficients are assumed, the time t superscript can be dropped. We can also stack the equations for asset i across time and re-write the time-series regression model in matrix notation as,

$$\boldsymbol{r}_i = \mathbf{1}_T\,\hat{\alpha}_i + \boldsymbol{F}\,\widehat{\boldsymbol{\beta}}_i + \hat{\boldsymbol{\varepsilon}}_i\,, \qquad i = 1, 2, \dots, N \tag{1.2}$$

where, $\boldsymbol{r}_i$ is the $T \times 1$ vector of asset i's total return, $\hat{\alpha}_i$ is the estimated intercept, $\boldsymbol{F}$ is a $T \times K$ matrix of factor returns (either observed macroeconomic variables, or extracted principal components or returns of a specially constructed portfolios as in the Fama-French approach) , $\widehat{\boldsymbol{\beta}}_i$ is a $K \times 1$ vector of asset i's estimated factor exposures and $\hat{\boldsymbol{\varepsilon}}_i$is the $T \times 1$ vector of asset i's estimated asset-specific return. The factor model can also be expressed as the multi-variate regression,

$$\boldsymbol{R} = \mathbf{1}_T\,\widehat{\boldsymbol{\alpha}'} + \boldsymbol{F}\,\widehat{\boldsymbol{B}} + \widehat{\boldsymbol{E}} \tag{1.3}$$

where, $\boldsymbol{R}$ is the $T \times N$ matrix of asset returns, $\hat{\boldsymbol{\alpha}}$ is the $N \times 1$ vector of estimated intercepts, $\boldsymbol{F}$ is the $T \times K$ matrix of factor returns, $\widehat{\boldsymbol{B}}$ is the $K \times N$ matrix of estimated factor exposures and $\widehat{\boldsymbol{E}}$ is the $T \times N$ matrix of estimated residuals. And, the covariance matrix of asset returns based on the multi-factor model above is given by,

$$\widehat{cov(\boldsymbol{R})} = \widehat{\boldsymbol{\Omega}} = \widehat{\boldsymbol{B}}'\,cov(\boldsymbol{F})\,\widehat{\boldsymbol{B}} + \widehat{\boldsymbol{D}} \tag{1.4}$$

where, $\widehat{\boldsymbol{D}}$ is the $N \times N$ diagonal matrix with the estimated residual variances $\hat{\sigma}_i^2 = var(\hat{\boldsymbol{\varepsilon}}_i)$ along the main diagonal.

In fundamental factor models, the observed factor exposures can vary with time, while the factor returns are estimated by a cross-sectional regression for each period. We can stack the equations for time t across assets and re-write the model in matrix notation as,

$$\boldsymbol{r}_t = \boldsymbol{\alpha} + \boldsymbol{B_t}\,\hat{\boldsymbol{f}}_t + \hat{\boldsymbol{\varepsilon}}_t\,, \qquad t = 1, 2, \ldots, T \tag{1.5}$$

where, $\boldsymbol{r}_t$ is the $N \times 1$ vector of the N asset returns at time t, $\boldsymbol{B_t}$ is a $N \times K$ matrix of observed factor exposures for time t and $\hat{\boldsymbol{f}}_{\boldsymbol{t}}$ is a $K \times 1$ vector of estimated factor returns for asset i. Since the estimated residual variances $\hat{\sigma}_i^2 = var(\hat{\boldsymbol{\varepsilon}}_i)$ are heteroskedastic across assets, the least squares estimate of $\hat{\boldsymbol{f}}_t$ will be unbiased but inefficient. An additional step is added to the estimation process, where the estimated residual variances can be used as weights in a weighted least squares regression to re-estimate $\hat{\boldsymbol{f}}_{t,WLS}$ and $\hat{\sigma}_{i,WLS}^2$. Then, the covariance matrix of asset returns based on the estimated fundamental multi-factor model is given by,

$$\widehat{cov(\boldsymbol{R})} = \widehat{\boldsymbol{\Omega}} = \boldsymbol{B_t}\,\widehat{\boldsymbol{\Omega}}_{F,WLS}\,\boldsymbol{B}_t' + \widehat{\boldsymbol{D}}_{WLS} \tag{1.6}$$

where, $\widehat{\boldsymbol{D}}_{WLS}$ is the $N \times N$ diagonal matrix with $\hat{\sigma}_{i,WLS}^2$ along the main diagonal. Note that the factor model covariance depends on the factor exposures $\boldsymbol{B_t}$ and the model builder needs to choose the appropriate period t for this purpose. In this paper, we use the factor exposures from the last period in the data. Other options might be historical average factor exposures or new anticipated exposures.

1.3 Data

To represent time-series factor models in the performance comparison, we use the Fama-French-Carhart 4-factor model (excess return on market, size, value and momentum) based on Fama & French (1992, 1993) and Carhart (1997). The size, value and momentum factors were constructed from long-short hedge portfolios of stocks sorted by the respective fundamental characteristic as described earlier. Monthly factor returns for these 4 factors (including the risk-free rate of return) were obtained from the CRSP database[4]. The monthly stock returns used in all 3 models are in terms of excess returns vs. the risk-free asset. Note that the statistical factor model only requires the time series of stock returns.

A panel data of monthly returns and fundamental characteristics for 200 stocks in the S&P 500 index was provided by S&P Global Market Intelligence[5]. We extract a balanced panel of returns and 5 fundamental characteristics for 122 stocks with 209 contiguous months from Aug-1995 to Dec-2012. The resulting sample of stocks has representation across large, mid and small cap classifications and well as 9 of the 10 GICS[6] sectors. Table 1.1 provides a summary description of the panel data.

Table 1.1: Description of return and fundamentals panel data

Variable	**Description**	**Details**
Date	Monthly periods	209 months from Aug-1995 to Dec-2012
TICKER	Stock names	122 stocks from the S&P 500 index
RET	Monthly returns	Excess returns over the risk-free rate
BP	Value	Book-to-Price ratio
AnnVol1M	Volatility	Annualized 1-month volatility
PM12M1M	Price Momentum	12M - 1M Price Momentum
LogMktCap	Size	Log market capitalization
GSECTOR	10 GICS sectors	Energy, Materials, Industrials, Consumer Discretionary, Consumer Staples, Health Care, Financials, Information Technology, Telecommunication Services, Utilities

[4] Calculated (or Derived) based on data from database name ©2015 Center for Research in Security Prices (CRSP), The University of Chicago Booth School of Business.

[5] Source: S&P Global Market Intelligence; Compustat, S&P Capital IQ Estimates

[6] The 11th sector for Real Estate was added only in 2016.

Error! Not a valid bookmark self-reference. shows the breakdown by GICS sectors and market cap classification at the terminal date in the dataset. And, Figure 1.1 shows that the correlations between the monthly stock return and 4 fundamental numeric characteristics stacked across assets and time, are not prohibitively large for any pair.

Table 1.2: Breakdown of stocks by sector and market cap

GICS Sector	**No. of stocks**
Energy	4
Materials	11
Industrials	37
Consumer Discretionary	21
Consumer Staples	10
Health Care	9
Financials	6
Information Technology	16
Telecommunication Services	0
Utilities	8

Market cap classification	**No. of stocks**
Large-cap	38
Mid-cap	35
Small-cap	48
Micro-cap	1

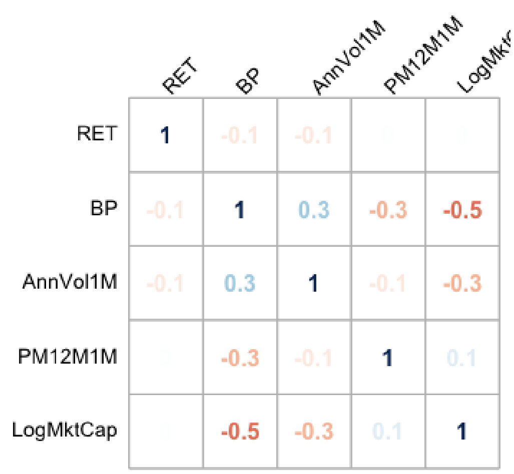

Figure 1.1: Correlation between returns and fundamental characteristics

1.4 METHODOLOGY

1.4.1 *Factor Model Fitting*

The open-source R package *factorAnalytics* (co-developed with other UW students and faculty members) contains the necessary tools for fitting all 3 types of factor models. Chapters 3, 4 and 5 of this dissertation describes and demonstrates the model fitting functions and related methods.

In addition to comparing the 3 main types of models, within each class of models we make comparisons in terms of other features outlined below. We also add the sample covariance and expected return as a naïve forecasting model.

Within the class of time series factor models based on the 4 Fama-French-Carhart factors, we compare regression methods: ordinary least squares, robust regression (which is resistant to outliers) and lasso subset selection method. Given the small number of factors under consideration here, we realize there may not be much differentiation afforded by the different choices for subset selection (stepwise, best subset based on AIC, BIC or Mallows'Cp and shrinkage methods like lasso). The time series factor model equation for asset i, where $i = 1, 2, \dots, N$, is given by,

$$\boldsymbol{r}_i - \boldsymbol{r}_f = \mathbf{1}_T\,\hat{\alpha}_i + \left(\boldsymbol{Mkt} - \boldsymbol{r}_f\right)\hat{\beta}_{i,mkt} + \boldsymbol{SMB}\,\hat{\beta}_{i,size} + \boldsymbol{HML}\,\hat{\beta}_{i,val} + \boldsymbol{UMD}\,\hat{\beta}_{i,mom} + \hat{\boldsymbol{\varepsilon}}_i \quad (1.7)$$

where, $(\boldsymbol{r}_i - \boldsymbol{r}_f)$ is the $T \times 1$ vector of excess returns for asset i over the risk-free rate, $(\boldsymbol{Mkt} - \boldsymbol{r}_f)$, $\boldsymbol{SMB}$ and $\boldsymbol{HML}$ are the 3 Fama-French factors (excess return on the market, size and value respectively), while $\boldsymbol{UMD}$ is the Carhart momentum factor, and $\hat{\alpha}_i$, $\hat{\beta}_{i,mkt}$, $\hat{\beta}_{i,size}$, $\hat{\beta}_{i,val}$ and $\hat{\beta}_{i,mom}$ are the estimated intercept and corresponding factor exposures for asset i from the time series regression.

For statistical factor models, we compare models with different number of factors. In our example, since the chosen time series factor model has 4 factors, we test a 4-factor statistical model. We also fit a 5-factor model to observe the impact of an additional factor on performance. Since the number of assets (122) is less than the number of time periods (209), principal components analysis is used. The equation for the statistical factor model with 5 factors for asset i, where $i = 1, 2, \dots, N$, is given by,

$$(\boldsymbol{r}_i - \boldsymbol{r}_f) = \mathbf{1}_T\,\hat{\alpha}_i + \boldsymbol{f}_{\mathbf{1}}\hat{\beta}_{i,1} + \boldsymbol{f}_{\mathbf{2}}\,\hat{\beta}_{i,2} + \boldsymbol{f}_{\mathbf{3}}\,\hat{\beta}_{i,3} + \boldsymbol{f}_{\mathbf{4}}\,\hat{\beta}_{i,4} + \boldsymbol{f}_{\mathbf{5}}\,\hat{\beta}_{i,5} + \hat{\boldsymbol{\varepsilon}}_i \qquad (1.8)$$

where, $(\boldsymbol{r}_i - \boldsymbol{r}_f)$ is the $T \times 1$ vector of excess returns for asset i over the risk-free rate, $\boldsymbol{f}_{\mathbf{1}}, \boldsymbol{f}_{\mathbf{2}}, \boldsymbol{f}_{\mathbf{3}}$, $\boldsymbol{f}_{\mathbf{4}}$ **and** $\boldsymbol{f}_{\mathbf{5}}$ are the first 5 principal components from the eigen decomposition of the return covariance matrix, and $\hat{\alpha}_i$, $\hat{\beta}_{i,1}$, $\hat{\beta}_{i,2}$, $\hat{\beta}_{i,3}$, $\hat{\beta}_{i,4}$ and $\hat{\beta}_{i,5}$ are the estimated intercept and respective factor exposures for asset i from the time series regression.

For the fundamental factor model, we use a 4-factor model (size, value, momentum and volatility) to enable like-for-like comparison in terms of number of factors used. We also test the effects of adding the 10 sector factors to the above model. Note that for estimating the factor model asset return covariance and for forecasting returns, we use the last period factor exposures (refer to equations 1.4 and 1.6). The equation for the fundamental factor model with 4 style factors and 10 sector dummies at time t, where $t = 1, 2, \dots, T$, is given by,

$$\left(\boldsymbol{r}_t - \mathbf{1}_N r_{f,t}\right) = \boldsymbol{\beta}_{size,t}\hat{f}_{size,t} + \boldsymbol{\beta}_{value,t}\hat{f}_{value,t} + \boldsymbol{\beta}_{mom,t}\hat{f}_{mom,t} + \boldsymbol{\beta}_{vol,t}\hat{f}_{vol,t} + \boldsymbol{B}_{sector,t}\hat{F}_{sector,t} + \hat{\boldsymbol{\varepsilon}}_t \qquad (1.9)$$

where, $(\boldsymbol{r}_t - \mathbf{1}_N r_{f,t})$ is the $N \times 1$ vector of excess returns over the risk-free rate for the N assets at time t, while $\boldsymbol{\beta}_{size,t}$, $\boldsymbol{\beta}_{value,t}$, $\boldsymbol{\beta}_{mom,t}$, $\boldsymbol{\beta}_{vol,t}$ are $N \times 1$ vectors of the observed numeric style exposures for size, value, momentum and volatility factors described earlier, and $\boldsymbol{B}_{sector,t}$ is a $N \times 10$ matrix of *1*'s and *0*'s indicating whether each of the N stocks belongs to one of the 10 sectors (with each stock belonging to one and only one sector at any given period) to, and $\hat{f}_{size}$,

$\hat{f}_{value}$, $\hat{f}_{mom}$, $\hat{f}_{vol}$, and $\hat{F}_{sector,t}$ are the estimated factor returns respectively at time t from the cross-sectional regression.

Lastly, we add 3 hybrid factor models to the horse race. The 3 hybrid models are formed by adding statistical factors extracted from the residuals to the time series and fundamental factor models as well as adding a fundamental factor model to the residuals of a time series factor model. For example, the equation for the hybrid factor model fitting 5 statistical factors to the residual of the fundamental factor model in equation 1.9 above, for asset i, where $i = 1, 2, \dots, N$, is given by,

$$\widehat{\boldsymbol{res}}_i = \mathbf{1}_T\, \hat{\delta}_i + \boldsymbol{g_1}\hat{\gamma}_{i,1} + \boldsymbol{g_2}\, \hat{\gamma}_{i,2} + \boldsymbol{g_3}\, \hat{\gamma}_{i,3} + \boldsymbol{g_4}\, \hat{\gamma}_{i,4} + \boldsymbol{g_5}\, \hat{\gamma}_{i,5} + \widehat{\boldsymbol{\theta}}_i \qquad (1.10)$$

where, $\widehat{\boldsymbol{res}}_i$ is the $T \times 1$ vector[7] of estimated residuals for asset i from the fundamental factor model in equation 1.9 with 4 style factors and 10 sector dummies, $\boldsymbol{g_1}$**,** $\boldsymbol{g_2}$**,** $\boldsymbol{g_3}$**,** $\boldsymbol{g}$ **and** $\boldsymbol{g_5}$ are the first 5 principal components from the eigen decomposition of the residual covariance matrix, $\hat{\delta}_i$, $\hat{\gamma}_{i,1}$, $\hat{\gamma}_{i,2}$, $\hat{\gamma}_{i,3}$, $\hat{\gamma}_{i,4}$ and $\hat{\gamma}_{i,5}$ are the estimated intercept and respective factor exposures for asset $i's$ residual from a time series regression, and $\widehat{\boldsymbol{\theta}}_i$ is the $T \times 1$ vector of newly estimated residuals for the hybrid model.

For comparison, Table 1.3 lists the factors used by Connor (1995) for each type of factor model. Note that the number of factors varied across the 3 models and there was only one specification representing each type of model (with 5 macroeconomic factors, 5 statistical factors and 12 fundamental risk indexes[8] with 55 industry dummies). Furthermore, his dataset consisted of 779 large cap U.S. equities over the 108-month period from January 1985 – December 1993.

---

[7] Note that the $N \times 1$ vector of residuals from each period T, from equation 1.9, can simply be re-stated as $T \times 1$ vector of residual returns for each of the N assets for the purposes of fitting a statistical factor model in the hybrid specification.

[8] For detailed descriptions of the industry categories and risk indexes used by Connor (1995), see *The United States Equity Model Handbook* (Berkeley, CA: BARRA Inc., 1994).

Table 1.3: List of factors used in Connor (1995)

Time-series (or) Macroeconomic factors	**Statistical factors**	**Fundamental factors**
Inflation Term Structure Industrial Production Default Premium Unemployment	First 5 principal components from the eigen decomposition of the asset return covariance matrix	55 Industry dummies Variability in markets Success Size Trade activity Growth Earnings-to-price Book-to-price Earnings variability Financial leverage Foreign investment Labor intensity Dividend yield

1.4.2 *In-sample Performance Comparison*

Connor (1995) compares the three types of factor models based on explanatory power given by $1 - \sigma_\varepsilon/\sigma$ **,** where σ_ε and σ are the average asset-specific and average total return variance over all securities respectively. This measure can also be used to define the explanatory power of an additional factor as the difference in explanatory power of the model when the factor is added to it. He finds that the fundamental factor model outperforms the statistical factor model by a small margin and they both significantly outperform the time series factor model. Further, adding a statistical factor improved the explanatory power of the fundamental factor model, while adding a time series factor model didn't help. However, these comparisons were made for a specification of each type of factor model only. We use this measure of explanatory power to compare multiple types and specifications of factor models as described earlier.

Next, we look at the distribution of R^2 across assets (or periods) for the different types of models to understand the role played by the number and type of factors used. Another tool used in the process of determining the right model specification is analyzing the structure of residual correlations to identify clustering. This may help to identify omitted factors, for example, a common industry risk factor. We perform simple hierarchical clustering on the residuals of the time series and statistical factor models to test for patterns that might indicate omitted sector effects.

There is mixed evidence on whether modeling time-varying betas is helpful. Some authors, such as Faff et al. (2000) and Swinkels & Sluis (2006), have shown that modelling time-varying beta even in single factor contexts performs better than more complicated GARCH type models. Whereas some others (Ghysels, 1998) show that in case there's a mis-specification, time-varying betas can actually hurt. Thus, it makes sense to inspect the stability of regression coefficients over time using rolling regressions. If the rolling estimate of a factor loading appears to be a random walk, we can expect that estimated coefficient to not be significant. We also compare the coefficients across different sub-samples, determined based on macro growth[9]. For this purpose, we split our dataset into 3 subsets, 1995-2000, 2000-2007 and 2007-2012, approximately 5-years each, that correspond to periods of differentiated economic growth or slowdowns - Table 3 from Jones (2016).

Mean-variance optimization (Markowitz, 1952), the foundation of modern portfolio theory, depends heavily on the asset return covariance matrix. To test the factor model estimated asset return covariance matrix (Equations 1.4 and 1.6), we compare the performance of the global

[9] Raffinot (2014) shows that a macro-based portfolio rebalancing at different positions of the growth cycle can improve long-term investment policy.

minimum variance portfolio across models. We also compare the strategy's allocations to different industries and asset classes.

1.4.3 *Out-of-sample Performance Comparison*

Rolling estimates over a 60-month observation window[10] are used to estimate the one-month-ahead forecasts of asset returns. For time series factor models, one-month ahead factor returns are known (hence these can be called pseudo-out-of-sample estimates). For fundamental factor models, the factor returns for the entire sample can be estimated by cross-section regression ahead of time, and so future factor returns are pseudo-known in this case as well. Alternately, an exponentially weighted moving average forecast can be used. This alternate technique is necessary for statistical factor models.

When testing the efficacy of the risk and return forecasts based on the factor models, we reduce the dimension for comparison from 122 stock returns to the return on one equally weighted portfolio (EWP). To test whether the risk forecasts were accurate on average over time, we use the bias statistic, following Menchero et al. (2013). The bias statistic, defined as the standard deviation of standardized returns as shown below, represents the ratio of realized risk to forecast risk.

$$Bias\ Statistic = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}\left(b_t - \bar{b}\right)^2} \tag{1.11}$$

where, $b_t = \hat{r}_t/\hat{\sigma}_t$ is the standardized return forecast and $\bar{b} = \sum_{t=1}^{T} b_t$ is the mean standardized return over T observation windows. For accurate forecasts, the bias statistic is expected to be close to 1 and the 95% confidence interval, assuming normal returns, is approximately $1 \pm 2/T$. A value higher (lower) than 1 implies that the model is under-predicting (over-predicting) risk. Though

[10] We acknowledge that the ad hoc window size can lead to sub-optimal use of the data. With smaller windows that make the model more responsive, the robustness of the estimates can further deteriorate (Shepard, 2011).

simple and intuitive, since the bias statistic averages errors across time periods, it can't identify if the model's forecast accuracy is better or worse in certain market environments or regimes.

The second metric for out-of-sample performance is the information coefficient, defined as the correlation between actual EWP portfolio returns and the return explained by the factor model using the rolling window of observations. The results from these tests are shown in the next section.

1.5 RESULTS

In the interest of truly replicable research, the R code used in model fitting and performance comparison are shown in Appendix A. The estimated factor exposures and/or factor returns for representative factor models are in Appendix B, C & D for the interested reader.

Table 1.4 shows the in-sample explanatory power of the different models using our dataset as well in comparison to Connor's (1995) findings. In general, our results agree: the fundamental factor model has similar explanatory power as the statistical factor model and they both have better explanatory power compared to the time series factor model. And as expected, the hybrid factor models had similar or better explanatory power than their pure-type counterparts.

When the number of factor is the same for all 3 types of models, the fundamental and statistical factor models have very similar explanatory power, but they both outperform the time series factor models using LS regression (with or without subset selection). Recall that Connor (1995) used different number of factors for the different types of models to find that the fundamental factor model (with far greater number of factors – 12 risk characteristics and 55 industries) outperformed the statistical factor model (with 5 factors). We get similar results from adding the sector factors to the fundamental factor model, and hence increasing the amount of information used in fitting the model. Interestingly, when a robust estimation method is used for the time-series factor model, it appears to outperform all other models in terms of explanatory power. However, when we compare the average R^2 across models, robust method underperforms LS regression. This is because robust estimation omits the outliers with large unexplained returns.

Table 1.4: In-sample performance: Explanatory power

	Connor (1995)		**Current paper**	
	Model	**Explanatory power**	**Model**	**Explanatory power**
Time-series factor models	5 Macroeconomic factors	10.9%	Fama-French-Carhart (FFC4)	29.7%
			FFC4 (Robust)	52.6%
			FFC4 (Lasso)	30.1%
Statistical factor models	5 Statistical factors	39.0%	4 Statistical factors	38.1%
			5 Statistical factors	40.6%
Fundamental factor models	13 Fundamental factors	42.6%	4 Fundamental factors	38.4%
			4 Fundamental factors + Sector	47.1%
Hybrid factor models	Macro + Statistical	31.0%	FFC4 + 5 Statistical	44.4%
	Fundamental + Statistical	44.8%	Fundamental + 5 Statistical	53.4%
	Macro + Fundamental	43.0%	FFC4 + Fundamental	39.1%

To evaluate the choice of using only 5 statistical factors, Figure 1.2 shows a screeplot that indicates the fraction of total variance in the asset returns explained or represented by each principal component (displayed in decreasing order of their contribution). The 1st principal component explains 28% of the total return variance and the first 5 factors together explain 42% of the return variance. Adding a 6th factor wouldn't significantly add to explanatory power of the model. Though not shown here, 53 factors are required to explain 90% of the return variance, at which point the benefits of dimension reduction from 122 stocks has been diluted unless suitable rotations of the factors can be found that correspond to intuitive drivers of risk.

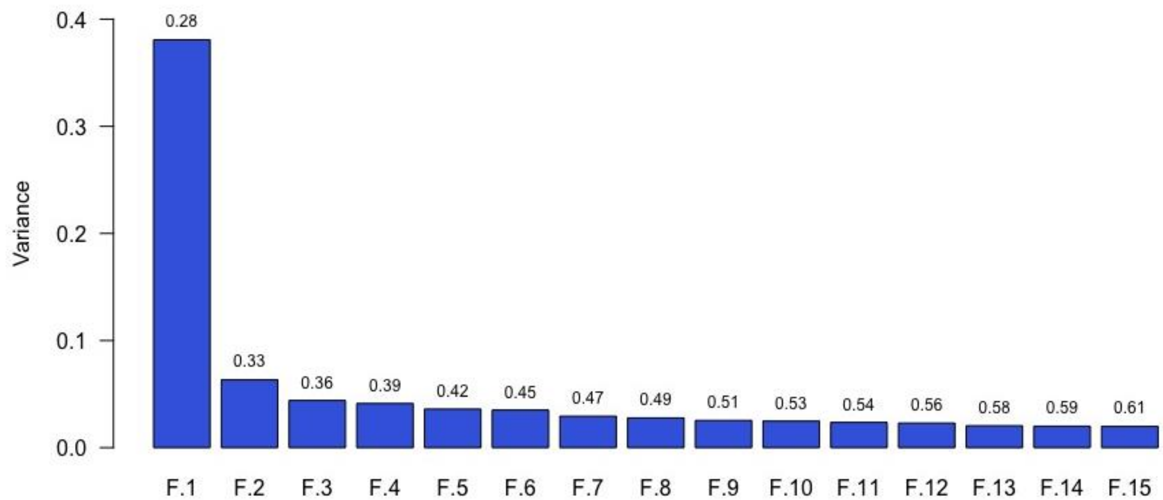

Figure 1.2: Screeplot of eigen values

Next, we show the distribution of R^2 (across assets or time, as appropriate) for the 10 types of models under comparison. Amongst the time-series models in Figure 1.3, lasso and least squares fits have very similar distributions, while the robust estimation method has lower R^2 on average. Given the small number of factors, variable selection didn't make a difference for most assets. The statistical factor models which also use least squares regression, can be compared in tandem, and are found to have a higher average R^2 as expected from the very definition of PCA analysis.

Figure 1.4 compares the distribution of cross-sectional R^2 across time periods for the two specifications of the fundamental factor model. Adding the GICS sector factors significantly increased the R^2. Figure 1.5 shows the R^2 from the additional or secondary models fitted on the residuals of the time series and fundamental factor models. The takeaway is that there is significant value added in fitting hybrid models that combine insights from different types of factors.

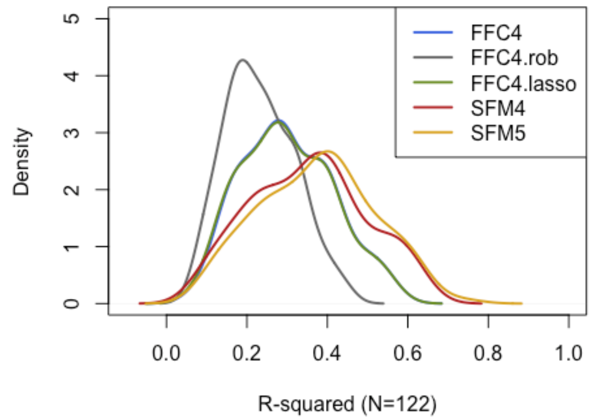

Figure 1.3: Distribution of R^2 across assets

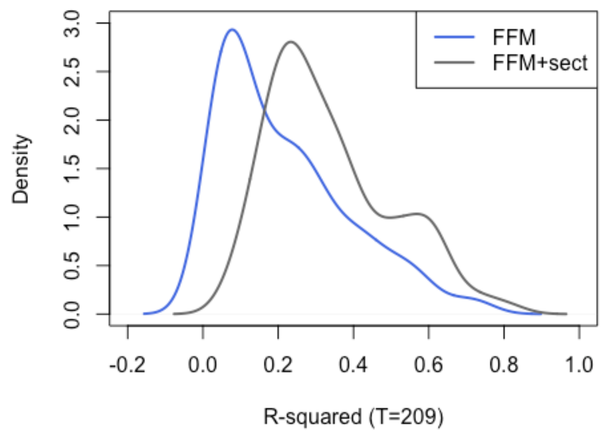

Figure 1.4: Distribution of R^2 across time

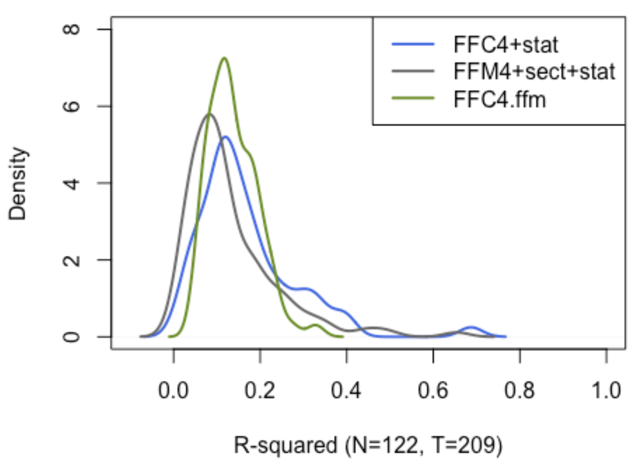

Figure 1.5: Distribution of R^2 for secondary models

To better understand the stability of the regression coefficients over time, we perform a rolling regression with 24-month observation windows for the Fama-French-Carhart 4-factor model. Figure 1.6 shows the estimated rolling regressions coefficients for asset “OXY”, chosen for demonstration. We visually note that the intercept, size (SMB) and momentum (UMD) exposures appear to be stationary around mean zero (except for the spike around 2008), while the market and value (HML) factors appear to have predominantly positive exposures. In this case, the p-values from the full-sample regression, shown in Table 1.5, agree with our assessment. However, if a smaller sub-sample in the latter part of the data was chosen, spurious exposures would have resulted as shown next.

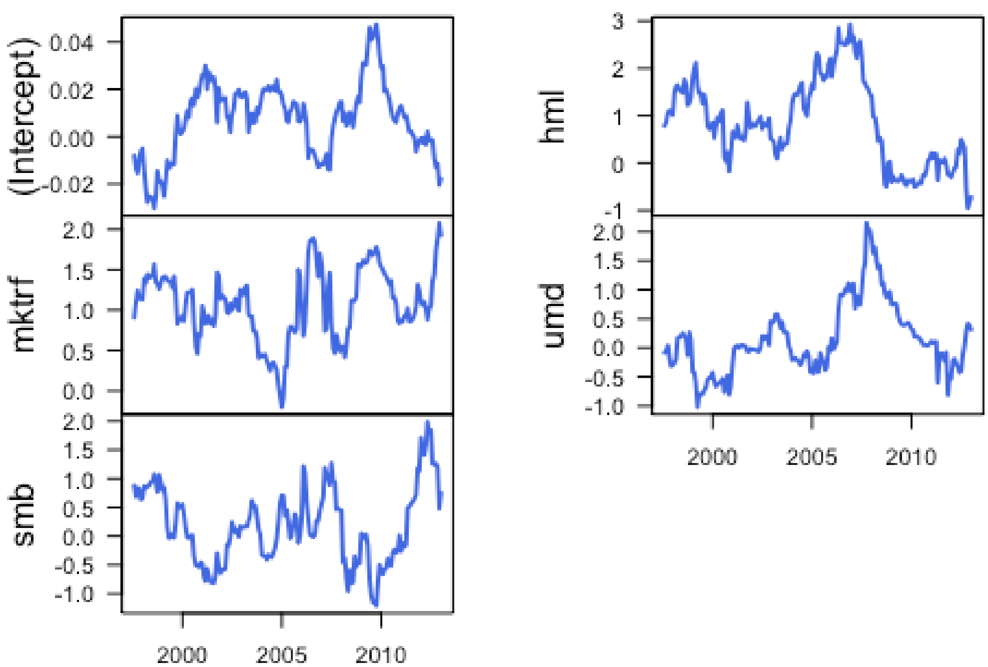

Figure 1.6: Rolling (24-month) regression estimates for "OXY"

Table 1.5: Estimated exposures for asset "OXY" (full-sample)

	Estimate	**Standard Error**	**t-value**	**p-value**
(Intercept)	0.006	0.005	1.159	0.248
Market – Risk-free	**0.960**	0.115	8.366	0.000
Size (SMB)	-0.130	0.146	-0.890	0.374
Value (HML)	**0.784**	0.156	5.043	0.000
Momentum (UMD)	0.072	0.093	0.778	0.437

Next, we take a closer look at the estimated coefficients for "OXY" from the time-series model fits across 3 independent sub-samples (1995-2000, 2000-2007, 2007-2012) with different economic growth regimes as explained earlier. Table 1.6 shows the estimated coefficients from the 3 sub-samples as well as the full sample. We observe that the estimated coefficients based on the (2008-2012) sample which was during and after the 2008 Global Financial Crisis (GFC) when economic or earnings growth was low, are significantly different, in fact has the opposite sign, for the value factor. Recall that the value factor had a significant positive exposure in the full-sample regression. Further analysis may be required to determine of this was caused by a change in the risk premium for the value factor or a fundamental change in "OXY" during and after the GFC.

Table 1.6: Estimated exposures for asset "OXY" (independent sub-samples)

	Exposure (full-sample)	**Exposure (1995-2000)**	**Exposure (2001-2007)**	**Exposure (2008-2012)**
(Intercept)	0.01	0.00	0.02	0.00
Market – Risk-free	0.96	1.17	1.03	1.35
Size (SMB)	-0.13	0.12	0.07	0.24
Value (HML)	0.78	1.06	0.63	-0.52
Momentum (UMD)	0.07	-0.48	0.34	0.20

Continuing with the in-sample comparison of the different types of factor models, Table 1.7 shows the properties of the global minimum variance (GMV) portfolio computed based on the factor model estimated expected returns and return covariance matrix. The statistical and time-series factor models result in very similar portfolio holdings and portfolio statistics (approximately 6% annualized volatility and 10% annualized expected return). The expected return and volatility of the GMV portfolio based on the fundamental factor models was significantly higher. This is possibly also due to the use of the last period factor exposures when estimating the covariance matrix. Observing the sample covariance based GMV portfolio, though the expected return and optimal volatility are similar to those of the time series and statistical factor models, the portfolio holdings have major differences highlighting the differential attribution of risks.

Table 1.7: Factor-model implied global minimum variance portfolios

	Sample covariance	Fama-French-Carhart (FFC4)	FFC4 (robust regression)	FFC4 (lasso regression)	SFM (4 factors)	SFM (5 factors)	FFM (4 factors)	FFM (4 factors with sectors
Ann. Volatility	6.1%	6.3%	5.2%	5.9%	5.8%	5.7%	12.2%	11.9%
Ann. Exp. Return	10.1%	9.7%	10.2%	9.7%	9.7%	9.9%	11.0%	11.6%
Cons. Discretionary	3	19	20	18	11	12	19	16
Consumer Staples	16	9	8	8	5	6	9	4
Energy	-4	10	10	10	8	8	-2	0
Financials	18	2	1	1	5	4	-7	3
Health Care	6	9	11	9	11	10	11	16
Industrials	25	21	20	24	31	30	29	40
Info. Technology	-3	13	10	15	9	9	20	9
Materials	28	14	11	12	15	16	13	4
Utilities	12	4	8	3	5	5	8	7

To motivate the need for adding fundamental factors to a time-series factor model, we perform clustering analysis[11] on the residuals from the Fama-French-Carhart 4-factor model as shown in Figure 1.7. By inspection (Figure 1.8 provides a closer look), some of the clusters were concentrated in certain sectors. For example, the top left cluster is formed by 6 stocks all of which belong to the "Utilities" sector, represented by 8 stocks in our dataset. Similar relationships can be found in the other clusters as well (sometimes as combinations of 2 or 3 sectors).

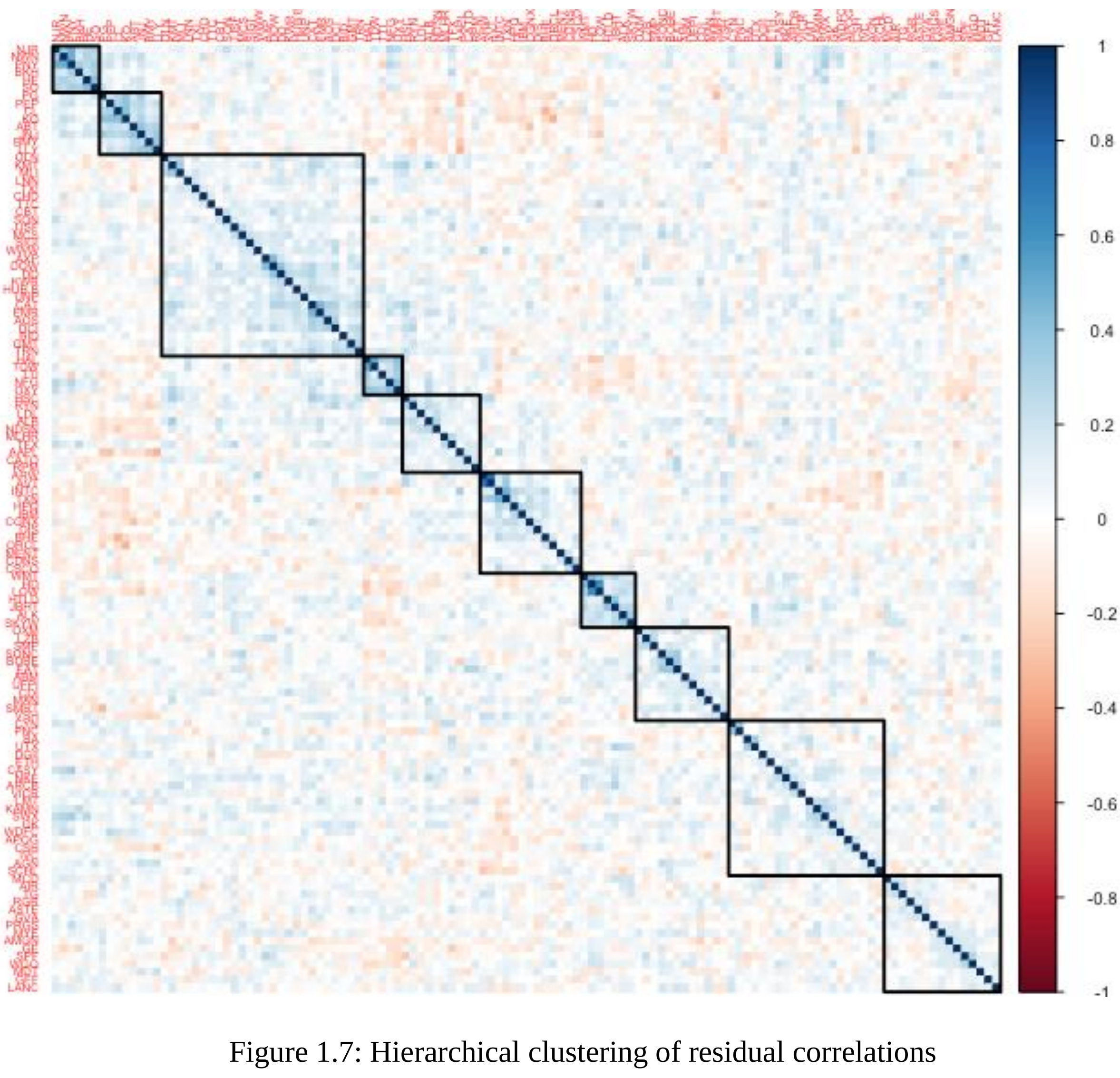

Figure 1.7: Hierarchical clustering of residual correlations

[11] Since we can choose the required number of clusters for the analysis, a cluster size of 10 was chosen, in part motivated by the 10 GICS sectors in our sample.

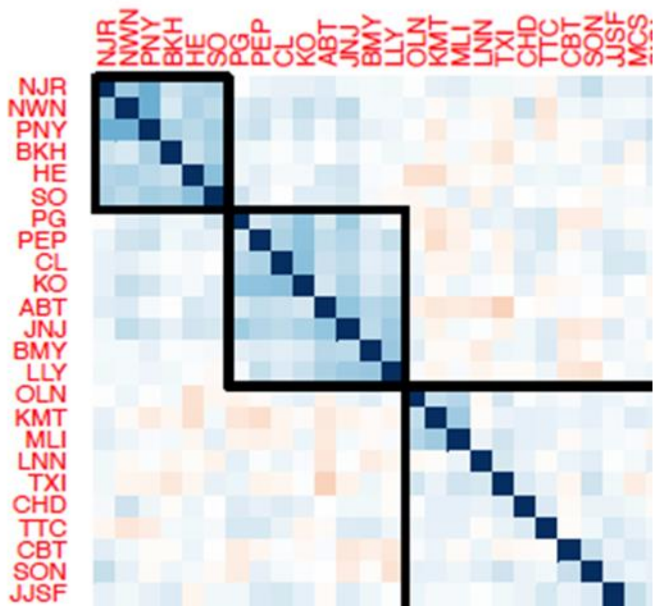

Figure 1.8: Top left clusters in the residual correlation matrix

Summary results from the out-of-sample performance comparison are shown in Table 1.8. These statistics are based on one-month ahead forecasts of the expected return and volatility of the equal-weighted portfolio, using rolling 60-month observation windows. The forecasted asset return covariance matrix is assumed to be the estimated covariance from the most recent 60-month fitted factor model. However, for the return forecast, we use the estimated (or, observed) factor exposures from the most recent 60-month fit in combination with pseudo-forecasted factor returns. For the purposes of this research paper, the pseudo-forecasts are either the ex post known factor returns (time series factor model) or estimated factor returns from the full-sample model (statistical and fundamental factor models). Other specifications for the 1-month ahead factor return forecasts are left to future research.

The time-series factor models outperformed the statistical and fundamental factor models in terms of the information coefficient, given the benefit of perfect foresight assumed for its forecasted factor returns. However, the bias statistics are greater than 1, indicating that the model under predicts risk. The fundamental factor model had the truest volatility forecasts, given the bias statistic very close to 1. The statistical factor models underperform both in terms of information coefficient as well as the bias statistic. The latter is extremely large indicating that the statistical factor model massively under predicts risk. This may be alleviated with the use of a larger number of factors.

Table 1.8: Out-of-sample performance comparison

Factor Model	**Bias Statistic**	**Information Coefficient**
Fama-French-Carhart (FFC4)	1.15	0.960
FFC4 (Robust)	1.17	0.964
4 Statistical factors	5.09	0.670
5 Statistical factors	5.07	0.784
4 Fundamental factors	0.98	0.945
4 Fundamental factors + Sector	0.99	0.944

1.6 Conclusion

The goal of this paper was to provide an update (using more recent data) on Connor's (1995) research into the performance comparison of the time-series, statistical and fundamental factor models when applied to a sample of U.S. equity returns, as well as add other dimensions to the analysis.

The comparison between different classes of models was made in terms of model specification, estimation and interpretation, as well as various in-sample and out-of-sample performance metrics. In-sample comparison was based on overall explanatory power of the model (similar to Connor, 1995), R^2 for individual fits, stability of regression coefficients over time and across sub-samples, examination of the residual correlation structure and performance of the estimated global minimum variance portfolio. Out-of-sample comparison was made using the bias statistics and information coefficient from rolling 5-year factor model forecasts of 1-month ahead return of the equally-weighted portfolio.

The in-sample results were generally in agreement with Connor's findings from more than 20 years ago. We find that the fundamental factor model outperforms the time-series model and has similar explanatory power as the statistical factor models for the same number of factors. The fundamental factor model benefits from the use of additional information on asset-specific characteristics. Additionally, we find that adding statistical factor(s) extracted from the residuals of time-series or fundamental factor models, or, adding fundamental factors to the time-series factor model to form hybrid models can further improve their performance.

Out-of-sample results (based on pseudo-forecasts) were also favorable to the fundamental factor model as it wasn't susceptible to under predicting risk. Further analysis using different forecasting methods is warranted.

Chapter 2. MULTI-ASSET FACTOR MODEL: ESTIMATING PEER FUND EXPOSURES, RISK BUDGETING AND PORTFOLIO CONSTRUCTION

2.1 INTRODUCTION

Historically, multi-asset portfolio managers have thought about their exposures and relative risks in terms of asset class exposures. A problem with this approach, discussed in Page and Taborsky (2011) and Bender et al. (2010), is that there are other risk exposures in the funds that will be unaccounted. If these risks are small both in absolute and relative terms they may not be material at the multi-asset level, but some of these active risks could be substantial.

In addition to asset classes or indexes not being a suitable description of an actively managed fund, asset class returns exhibit multi-collinearity, which reduces the confidence in estimated exposures. A typical investment management firm doesn't have complete holdings data for peer universes to which they are benchmarked in some cases and nearly always compared to by their clients. Therefore, we need to estimate exposures using regression analysis. However, the risk and performance contributions coming from different asset classes are hard to distinguish in a regression due to significant correlations between asset class returns. This makes the estimates imprecise and inaccurate, because the different risk contributors (factors) are not adequately identified.

Figure 2.1 shows the historical weekly return correlations of a representative list of multi-asset-class index returns during the 3-year period ending June 2017. Notice that over 40% of the correlations are greater than 0.4. Particularly, clusters of equity, investment grade (IG) fixed income and below-IG fixed income asset classes have return correlations above 0.7.

	U.S. Large & Mid Cap Equity	U.S. Small Cap Equity	Developed ex-U.S. Equity UH	EM Equity UH	Global Equity UH	Global Commodities UH	Global Infrastructure UH	Developed REITs UH	U.S. Core Fixed Income	Global.Credit.H	U.S. Short Govt	U.S. Short Credit	U.S. Long Govt Credit	U.S. TIPS	Cash	U.S. High Yield	Global High Yield H	EMD (hard)	Bank Loans	Global Convertibles H	Conscious Currency
U.S. Large & Mid Cap Equity	1	0.91	0.76	0.66	0.94	0.30	0.66	0.57	-0.23	-0.07	-0.37	-0.23	-0.16	-0.06	-0.03	0.58	0.61	0.46	0.45	0.88	0.07
U.S. Small Cap Equity	0.91	1	0.72	0.61	0.86	0.31	0.57	0.55	-0.23	-0.07	-0.35	-0.20	-0.17	-0.03	-0.03	0.62	0.62	0.42	0.44	0.86	0.07
Developed ex-U.S. Equity UH	0.76	0.72	1	0.83	0.93	0.42	0.82	0.67	-0.07	0.11	-0.21	-0.03	-0.02	0.10	0.06	0.64	0.70	0.62	0.52	0.84	-0.07
EM Equity UH	0.66	0.61	0.83	1	0.84	0.46	0.74	0.63	0.04	0.20	-0.07	0.10	0.05	0.20	0.20	0.64	0.70	0.73	0.49	0.73	0.04
Global Equity UH	0.94	0.86	0.93	0.84	1	0.40	0.79	0.67	-0.14	0.04	-0.29	-0.12	-0.08	0.04	0.04	0.66	0.71	0.61	0.52	0.91	0.01
Global Commodities UH	0.30	0.31	0.42	0.46	0.40	1	0.42	0.27	-0.06	-0.01	-0.06	0.04	-0.08	0.14	0.08	0.51	0.49	0.33	0.37	0.27	-0.03
Global Infrastructure UH	0.66	0.57	0.82	0.74	0.79	0.42	1	0.78	0.18	0.32	0.03	0.15	0.23	0.28	0.13	0.56	0.60	0.70	0.42	0.69	-0.03
Developed REITs UH	0.57	0.55	0.67	0.63	0.67	0.27	0.78	1	0.38	0.49	0.22	0.34	0.40	0.40	0.15	0.44	0.47	0.60	0.29	0.58	-0.03
U.S. Core Fixed Income	-0.23	-0.23	-0.07	0.04	-0.14	-0.06	0.18	0.38	1	0.94	0.86	0.85	0.96	0.86	0.19	-0.01	-0.03	0.35	-0.16	-0.17	0.00
Global.Credit.H	-0.07	-0.07	0.11	0.20	0.04	-0.01	0.32	0.49	0.94	1	0.72	0.82	0.95	0.82	0.20	0.20	0.21	0.50	0.06	0.04	0.05
U.S. Short Govt	-0.37	-0.35	-0.21	-0.07	-0.29	-0.06	0.03	0.22	0.86	0.72	1	0.91	0.72	0.75	0.21	-0.19	-0.21	0.15	-0.26	-0.34	-0.07
U.S. Short Credit	-0.23	-0.20	-0.03	0.10	-0.12	0.04	0.15	0.34	0.85	0.82	0.91	1	0.73	0.77	0.23	0.08	0.08	0.33	0.02	-0.15	-0.10
U.S. Long Govt Credit	-0.16	-0.17	-0.02	0.05	-0.08	-0.08	0.23	0.40	0.96	0.95	0.72	0.73	1	0.80	0.17	0.03	0.02	0.36	-0.12	-0.09	0.04
U.S. TIPS	-0.06	-0.03	0.10	0.20	0.04	0.14	0.28	0.40	0.86	0.82	0.75	0.77	0.80	1	0.16	0.13	0.11	0.43	-0.08	-0.02	-0.06
Cash	-0.03	-0.03	0.06	0.20	0.04	0.08	0.13	0.15	0.19	0.20	0.21	0.23	0.17	0.16	1	0.03	0.05	0.10	0.08	0.01	0.02
U.S. High Yield	0.58	0.62	0.64	0.64	0.66	0.51	0.56	0.44	-0.01	0.20	-0.19	0.08	0.03	0.13	0.03	1	0.98	0.67	0.78	0.68	0.08
Global High Yield H	0.61	0.62	0.70	0.70	0.71	0.49	0.60	0.47	-0.03	0.21	-0.21	0.08	0.02	0.11	0.05	0.98	1	0.73	0.80	0.72	0.09
EMD (hard)	0.46	0.42	0.62	0.73	0.61	0.33	0.70	0.60	0.35	0.50	0.15	0.33	0.36	0.43	0.10	0.67	0.73	1	0.47	0.56	0.07
Bank Loans	0.45	0.44	0.52	0.49	0.52	0.37	0.42	0.29	-0.16	0.06	-0.26	0.02	-0.12	-0.08	0.08	0.78	0.80	0.47	1	0.56	0.02
Global Convertibles H	0.88	0.86	0.84	0.73	0.91	0.27	0.69	0.58	-0.17	0.04	-0.34	-0.15	-0.09	-0.02	0.01	0.68	0.72	0.56	0.56	1	0.09
Conscious Currency	0.07	0.07	-0.07	0.04	0.01	-0.03	-0.03	-0.03	0.00	0.05	-0.07	-0.10	0.04	-0.06	0.02	0.08	0.09	0.07	0.02	0.09	1

Figure 2.1: Historical weekly asset class correlations (3-year period ending June 2017)

These high correlations are a signal that multi-collinearity is likely to be an issue when estimating exposures. The challenges with using asset classes to estimate peer exposures is one motivation for a factor model. We construct a multi-asset factor model based on 2 goals: 1) We wish to explain as much multi-asset portfolio performance as possible with as few factors as possible. 2) We want the factors to have low correlation.

2.2 Data and Code

Weekly multi-asset retail fund returns were obtained from Morningstar[12] for the 3-year period ending June 2017. Morningstar categorizes these funds into 5 U.S. fund allocation (USFA) peer groups based on the equity allocation in their portfolios: 15% – 30% equity, 30% – 50% equity, 50% – 70% equity, 70% – 85% equity and 85%+ equity. The number of funds in each category and the classification of each fund into these categories by Morningstar can potentially change every period. In our sample, the average number of funds in each of the 5 risk categories were between 40 and 200, with the highest representation in the 50% - 70% equity category.

Weekly asset class index returns for 28 multi-asset variables were obtained via Confluence and Thomson Reuters DataStream[13]. These include domestic and international equity, government, corporate and below-investment grade fixed income, real assets such as developed REITs, global

[12] Source: Morningstar, Inc. © 2017 Morningstar. All Rights Reserved. The information contained herein: (1) is proprietary to Morningstar and/or its content providers; (2) may not be copied or distributed; and (3) is not warranted to be accurate, complete or timely. Neither Morningstar nor its content providers are responsible for any damages or losses arising from any use of this information. Past performance is no guarantee of future results.

[13] Source: Thomson Reuters Datastream. (2017). Disclosure: The TR Data was provided “as is” and neither Thomson Reuters not any third-party suppliers shall be liable for any inaccuracy in the TR Data, the way in which it is used in this dissertation or any reliance on the TR Data by any third party.

commodities and global infrastructure as well as major currency returns. A detailed list of indexes is provided in Appendix E.

The $factorAnalytics$ R package has the tools necessary to fit a multi-factor time series model and perform risk attribution. R package $nloptr$ with the function $slsqp$ was used to find numeric solutions to the sequential quadratic programming step in deriving the optimal risk budget portfolios.

2.3 MULTI-ASSET FACTOR MODEL

To recap from chapters 1 and 2, a time-series factor model for N assets, over T periods, using K factors, estimated via time series regression, can be expressed as:

$$\boldsymbol{r}_i = \boldsymbol{1}_T\,\hat{\alpha}_i + \boldsymbol{F}\,\widehat{\boldsymbol{\beta}}_i + \hat{\boldsymbol{\varepsilon}}_i\,, \qquad i = 1, 2, \ldots, N \tag{2.1}$$

where, $\boldsymbol{r}_i$ is the $T \times 1$ vector of asset i's total return, $\hat{\alpha}_i$ is the estimated intercept, $\boldsymbol{F}$ is a $T \times K$ matrix of observed factor returns, $\widehat{\boldsymbol{\beta}}_i$ is a $K \times 1$ vector of asset i's estimated factor exposures and $\hat{\boldsymbol{\varepsilon}}_i$is the $T \times 1$ vector of asset i's estimated asset-specific return.

In this paper, we construct the factors from observed asset class index returns as shown in Table 2.1. These risk factors are intuitively developed in an attempt to represent predominant sources of risk and return for multi-asset portfolios. The specific indexes used are listed in Appendix E.

On the equity side, we include several factors: a global equity factor (which is the broad equity market in excess of cash), a size factor (difference between small and large stocks), an emerging market factor, a developed ex-U.S. factor, a global infrastructure factor and a developed real estate factor, each defined in excess of the global equity factor. These can be thought of as zero-dollar investments or self-financing portfolios (long the asset class and short the market).

To represent fixed income risk using factors we draw on the paper by Litterman & Scheinkman (1991) which used principal component analysis to show that that the 1st two factors (corresponding to the level and slope of the yield curve) explain close to 90% of the variation in Treasury yields of varying maturities (1 month to 30 year). These 2 factors have been shown to correspond to the level (Duration) and slope (Term) of the yield curve. We also include a credit factor to capture default risk. The last factor for currencies is based on an index that's an equal weighted portfolio of 3 currency factors: carry, value and term.

Table 2.1: Description of risk factors constructed from asset class indexes

Factor[14]	**Construction from indexes**
Global Equity UH	Global Equity UH – Cash
US Size	U.S. Small cap Equity – U.S. Large cap Equity
Excess Dev. ex-US UH	Developed ex-U.S. Equity UH – Global Equity UH
Excess EM UH	Emerging Markets Equity UH – Global Equity UH
Commodity	Global Commodities
Excess Global Infra	Global Infrastructire UH – Global Equity UH
Excess Dev.REITs	Developed REITs UH – Global Equity UH
Duration	Aggregate U.S. Treasury
Term	10+ year U.S. Treasury – 1-3 year U.S. Treasury
Excess Credit H	Global Aggregate Credit H – U.S. Treasury
Conscious Currency	Russell Conscious Currency Index

[14] "UH" and "H" refer to unhedged and hedged returns from the perspective of an USD investor.

Earlier we motivated the use of factors to reduce correlations that exist between asset class returns. The return correlations for these newly constructed factors are shown in Figure 2.2. Less than 13% of the correlations have a magnitude greater than 0.4 (compare with over 40% for asset classes). In particular, we see a material reduction in the correlations between the equity factors. This will lead to a significant tightening of confidence bands when we estimate exposures later in the paper.

	Global Equity UH	US Size	Excess Dev. ex-US UH	Excess EM UH	Commodity	Excess Global Infra	Excess Dev.REITs	Duration	Term	Excess Credit H	Conscious Currency
Global Equity UH	1	0.24	0.09	0.24	0.40	-0.27	-0.36	-0.26	-0.21	0.59	0.01
US Size	0.24	1	0.04	0.03	0.16	-0.22	-0.04	-0.15	-0.17	0.24	0.04
Excess Dev. ex-US UH	0.09	0.04	1	0.26	0.17	0.32	0.14	0.15	0.12	0.01	-0.22
Excess EM UH	0.24	0.03	0.26	1	0.33	0.17	0.08	0.17	0.10	0.14	0.05
Commodity	0.40	0.16	0.17	0.33	1	0.06	-0.14	-0.14	-0.17	0.26	-0.03
Excess Global Infra	-0.27	-0.22	0.32	0.17	0.06	1	0.59	0.55	0.51	-0.41	-0.07
Excess Dev.REITs	-0.36	-0.04	0.14	0.08	-0.14	0.59	1	0.68	0.63	-0.47	-0.06
Duration	-0.26	-0.15	0.15	0.17	-0.14	0.55	0.68	1	0.95	-0.65	-0.02
Term	-0.21	-0.17	0.12	0.10	-0.17	0.51	0.63	0.95	1	-0.56	0.02
Excess Credit H	0.59	0.24	0.01	0.14	0.26	-0.41	-0.47	-0.65	-0.56	1	0.12
Conscious Currency	0.01	0.04	-0.22	0.05	-0.03	-0.07	-0.06	-0.02	0.02	0.12	1

Figure 2.2: Historical weekly factor returns correlations (3-years ending June 2017)

2.4 Estimating Peer Exposures and Risk Attribution

We exploit the Bias-Variance tradeoff, which refers to sacrificing some accuracy from using only a subset of the most relevant factors (due to omitted variable bias) to get more precise estimate of exposures (with lower standard error and tighter confidence bands). Subset selection is performed via Lasso regression (using the Mallows's Cp criterion), which constrains the absolute size of the coefficients while trying to explain as much of the return variability as possible. Once we have identified a subset of the most significant factors, we perform OLS regression to estimate factor exposures and derive a 95% normal confidence band for each exposure.

We make use of the estimated factor model to perform a risk attribution; using an Euler decomposition of the portfolio volatility based on Meucci (2007) and detailed in chapter 3. To summarize, the standard deviation of asset i, σ_i, can be decomposed as:

$$\sigma_i = \sum_{k=1}^{K+1} cSd_{i,k} = \sum_{k=1}^{K+1}\left(\beta_{i,k}^{*} * mSd_{i,k}\right) \qquad (2.2)$$

where, $cSd_{i,k}$ and $mSd_{i,k}$ are the component and marginal contributions to risk from the k^{th} factor. While the component contribution is the total contribution to risk from factor k, the marginal contribution to risk is the effect on the asset's standard deviation due to an incremental change in its exposure to the k^{th} factor, holding all else constant. Note that the residual is considered the $K+1^{\text{th}}$ risk factor, where the exposure to the residual is the residual standard deviation, and the residual factor returns are assumed to be $iid \sim(0,1)$.

A potential issue with this analysis is that in using the historical peer returns, the estimated exposures are inherently backward-looking and the data is subject to survivorship bias in the funds. Using exponentially weighted regressions on 3 years of weekly data ending June 2017 (with a half-life of 1 year) that give more importance to more recent data, results in a more relevant estimate of "current" peer risk exposures.

Figure 2.3 shows the estimated factor exposures and regression adjusted-R^2. We observe the adjusted-R^2 is large for all 5 groups, with the residual contributing less than 1% of the total risk. The standard errors, t-stats and confidence intervals for the estimated exposures (given in Appendix F) are indicative of reasonably good fits. Figure 2.4 shows the estimated risk contributions from each factor (including the unexplained residual), and Figure 2.5 shows the same factor-risk contributions as a percentage of total portfolio risk.

We make some key inferences regarding the average U.S. retail fund's allocations. They have,

1) Domestic bias in equity relative to global market cap, evidenced by the large positive exposure to the global equity factor and negative exposures to the non-U.S. factors,
2) Positive exposures to equity size, commodities, infrastructure, bond duration and credit factors,
3) Significant risk concentrated in the global equity factor in the more aggressive peer groups and in the duration factor for the more conservatively allocated peer groups.

	US FA 15-30	US FA 30-50	US FA 50-70	US FA 70-85	US FA 85-100
(Intercept)	0.000	0.000	0.000	0.000	0.000
Global Equity UH	0.252	0.388	0.617	0.796	0.938
US Size	0.041	0.064	0.081	0.116	0.151
Excess Dev. ex-US UH	-0.096	-0.134	-0.281	-0.309	-0.338
Excess EM UH	-0.015	-0.035	-0.080	-0.083	-0.104
Commodity	0.046	0.042	0.028	0.030	0.016
Excess Global Infra	0.028	0.038	0.017	0.034	
Excess Dev.REITs	0.013		-0.007	-0.007	-0.024
Duration	0.550	0.515	0.288	0.110	
Term	-0.041	-0.038	-0.019		0.016
Excess Credit H	0.395	0.416	0.159		-0.098
Conscious Currency	-0.001	-0.009	-0.004	-0.014	-0.028
Adj. R-squared	97.1	98.1	99.4	99.3	99.3

Figure 2.3: Estimated peer average factor exposures (3-years ending June 2017)

	US FA 15-30	US FA 30-50	US FA 50-70	US FA 70-85	US FA 85-100
Residual	0.04%	0.03%	0.02%	0.02%	0.03%
Global Equity UH	2.69%	4.45%	7.25%	9.41%	11.07%
US Size	0.07%	0.12%	0.15%	0.22%	0.30%
Excess Dev. ex-US UH	-0.03%	-0.02%	0.11%	0.10%	0.12%
Excess EM UH	-0.05%	-0.09%	-0.10%	-0.11%	-0.12%
Commodity	0.27%	0.23%	0.13%	0.15%	0.08%
Excess Global Infra	0.00%	-0.04%	-0.04%	-0.08%	
Excess Dev.REITs	-0.01%		0.02%	0.02%	0.10%
Duration	0.32%	-0.07%	-0.25%	-0.11%	
Term	-0.07%	0.00%	0.03%		-0.04%
Excess Credit H	0.31%	0.41%	0.17%		-0.11%
Conscious Currency	0.00%	0.00%	0.00%	0.00%	0.00%
Total Risk (TE)	3.54%	5.02%	7.49%	9.62%	11.43%

Figure 2.4: Factor-risk contributions of peer average returns (3-years ending June 2017)

	US FA 15-30	US FA 30-50	US FA 50-70	US FA 70-85	US FA 85-100
Residual	1.1	0.7	0.2	0.2	0.2
Global Equity UH	76.1	88.5	96.6	97.8	97.0
US Size	1.9	2.4	2.0	2.3	2.6
Excess Dev. ex-US UH	-0.9	-0.4	1.5	1.0	1.0
Excess EM UH	-1.4	-1.7	-1.4	-1.2	-1.0
Commodity	7.6	4.7	1.8	1.5	0.7
Excess Global Infra	-0.1	-0.8	-0.5	-0.8	
Excess Dev.REITs	-0.2		0.3	0.3	0.8
Duration	9.0	-1.4	-3.3	-1.2	
Term	-2.1	0.0	0.4		-0.4
Excess Credit H	8.8	8.1	2.3		-1.0
Conscious Currency	0.0	0.0	0.0	0.0	0.0

Figure 2.5: Factor-risk contributions as a percentage of total risk for peer average returns

For comparison, we also perform an asset class regression on the peer returns to directly estimate a representative portfolio allocation for each peer group category as shown in Figure 2.6 below. We use lasso regression for subset selection and then perform a constrained least squares regression with the constraints that the weights are non-negative and sum to 1. The annualized tracking error to the actual peer average returns are low and suggests a reasonably good fit.

But as mentioned earlier, the multi-collinearity issue might be confounding the exposures. For example, when performing iterative subset selection of the asset classes we observed that asset classes such as long bonds, emerging market debt and global convertibles have similar risk attributes (due to exposure to common factors such as duration, credit and equity market factors) that make it difficult to pinpoint with confidence the exact vehicle used in the portfolio implementation, purely based on realized portfolio returns.

Asset Class	US FA 15-30	US FA 30-50	US FA 50-70	US FA 70-85	US FA 85-100
Intercept	-0.9%	-1.0%	-1.4%	-1.6%	-1.3%
U.S. Large & Mid Cap Equity	12.7%	20.7%	42.3%	48.2%	56.7%
U.S. Small Cap Equity	1.5%	3.5%	6.9%	10.7%	12.7%
Developed ex-U.S. Equity UH	1.0%	4.5%	6.6%	11.0%	15.6%
EM Equity UH	1.0%	1.0%	1.6%	2.9%	3.3%
Global Commodities UH	2.9%	2.6%	2.4%	3.3%	2.2%
Global Infrastructure UH	2.0%	2.4%	1.3%	2.8%	0.0%
Developed REITs UH	2.7%	2.5%	0.0%	0.0%	0.0%
U.S. Core Fixed Income	8.0%	8.4%	12.7%	0.0%	0.0%
Global Credit H	0.0%	0.0%	0.0%	0.0%	0.0%
U.S. Short Govt	41.3%	27.9%	15.3%	11.0%	0.0%
U.S. Short Credit	0.0%	0.0%	0.0%	0.0%	0.0%
U.S. Long Govt/Credit	4.2%	3.5%	1.6%	2.3%	0.0%
U.S. TIPS	3.5%	2.7%	0.0%	0.0%	0.0%
Cash	0.0%	0.0%	0.0%	0.0%	0.0%
U.S. High Yield	11.8%	12.5%	4.8%	0.0%	0.0%
Global High Yield H	0.0%	0.0%	0.0%	0.0%	0.0%
EMD (hard)	3.6%	4.1%	1.5%	3.6%	0.0%
Bank Loans	0.0%	0.0%	0.0%	0.0%	0.0%
Global Convertibles H	3.7%	3.8%	3.0%	4.2%	9.6%
TE to realized peer returns	0.34%	0.42%	0.47%	0.72%	0.88%

Figure 2.6: Estimated asset class weights for peer average returns (3-years ending June 2017)

Apart from understanding the factor risk decompositions for any given portfolio risk category, it's important for multi-asset portfolio managers to estimate the changes in the *level* of the total portfolio risk through time, perhaps in response to changing macroeconomic regimes. Figure 2.7

shows the rolling 1-yr market beta of the peer average returns relative to the global equity return. We note that the peer's total market risk exposures are dynamic through time, varying approximately $\pm 5\%$ in each category over the last 10 years.

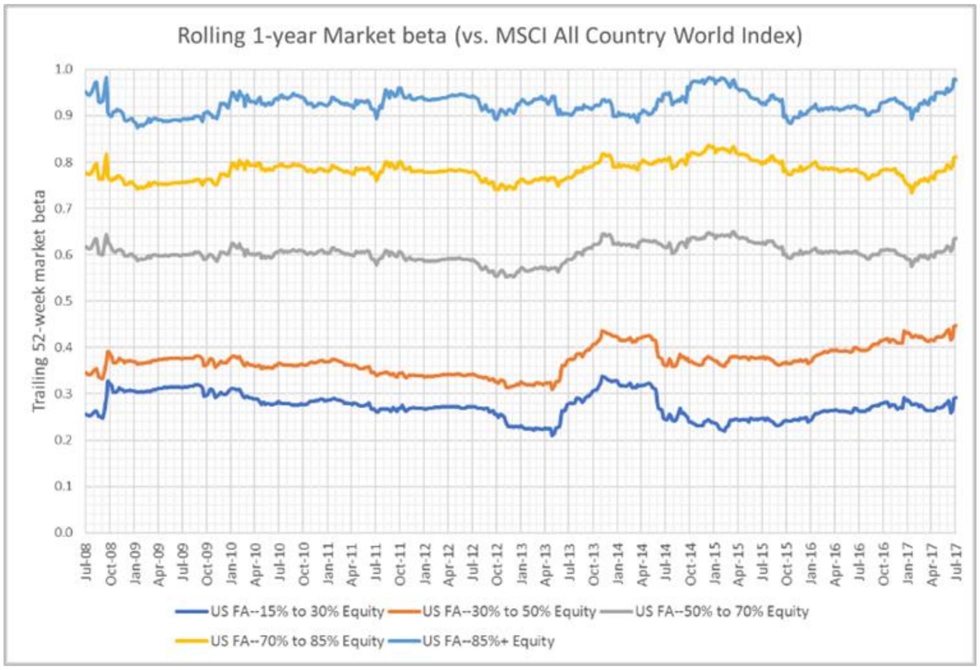

Figure 2.7: Rolling 1-year market beta of the peer average returns

2.5 RISK BUDGETING

Risk budgeting or risk parity refers to the idea of portfolio having equal risk contributions coming from different sources (asset classes or risk factors). Why should we consider risk parity? One can argue that risk parity is simply continuing in the tradition of Markowitz (1952, 1956) for optimal portfolio construction and the benefits of diversification. To quote Lee, Spellar and Bouchey (2013), "If one accepts the premise that the risk adjusted returns of all asset classes are equivalent, then a portfolio holding diversified asset classes that each contribute equally to risk should produce superior risk adjusted returns relative to a traditional 60/40 portfolio."

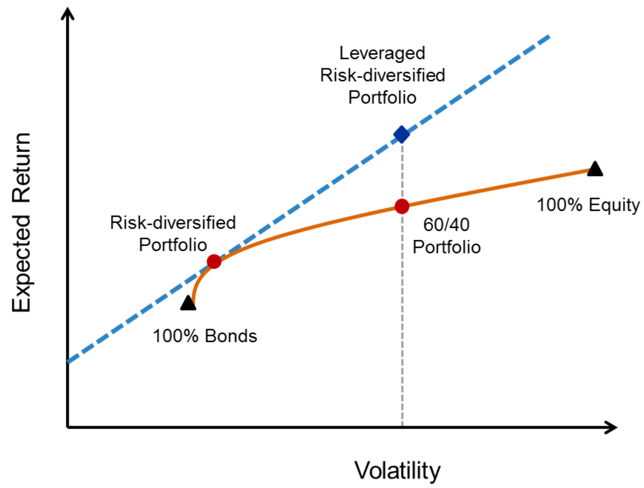

Figure 2.8: Illustration for diversification benefits of risk parity

Maillard, Roncalli & Teïletche (2010) provide a detailed discussion on the arguments for risk parity. They show that mean-variance framework has drawbacks (concentration of assets in a limited subset of assets, and overly sensitive to the input parameters, particularly expected returns), while alternative methods such as portfolio resampling and robust asset allocation have other disadvantages. In practice, a large fraction of investors prefer more heuristic solutions with computational simplicity and robustness to expected return forecasts, not complex estimation methods. The minimum variance portfolio is robust and computationally simpler, but results in high asset concentrations and potentially low returns. The equal-weighted portfolio is known to be efficient out of sample, but has limited diversification of risks. Thus, they argue for risk parity or equal asset-risk contribution portfolios, and derive closed-form solutions for simple cases. The optimal solution becomes endogenous in parameters as the complexity of the problem increases, requiring sequential quadratic programming for deriving the solution.

Using the asset classes defined earlier, we construct the equal asset-class risk portfolio and compare its performance against the equal-weighted portfolio and the minimum variance portfolio;

the results are shown in Figure 2.9. The equal-asset risk portfolio has a better Sharpe ratio than the equal weighted portfolio, while the minimum variance portfolio had a remarkably low volatility that boosted its Sharpe ratio despite poor expected returns.

	Ann. Return	Cum. Return	Ann. Vol.	Sharpe Ratio
Equal asset class risk	1.9%	5.7%	2.4%	0.779
Equal asset class weights	3.3%	10.3%	6.8%	0.484
Minimum Var (L only)	1.0%	3.1%	0.8%	1.346

Figure 2.9: Performance comparison of optimal portfolios (3-yrs ending June 2017)

Figure 2.10 shows the asset class weights for these 3 optimal portfolios. As mentioned earlier, the minimum variance portfolio is highly concentrated in the lowest volatility asset in the sample (short government bonds) and is not close to any realistic multi-asset portfolio. The equal asset risk portfolio makes a tradeoff between the equal-weighted and minimum variance portfolios. In fact, Maillard, Roncalli & Teïletche (2010) show that the variance of the equal-asset risk portfolio will always lie between those of the minimum variance portfolio and the equal-weighted portfolio.

	Equal asset class risk	Equal asset class weights	Minimum Var (L only)
U.S. Large & Mid Cap Equity	2.4%	7.1%	1.1%
U.S. Small Cap Equity	2.0%	7.1%	0.0%
Developed ex-U.S. Equity UH	1.8%	7.1%	0.0%
EM Equity UH	1.3%	7.1%	0.0%
Global Commodities UH	2.9%	7.1%	0.0%
Global Infrastructure UH	1.7%	7.1%	0.0%
Developed REITs UH	1.7%	7.1%	0.0%
U.S. Core Fixed Income	9.4%	7.1%	0.0%
U.S. Short Govt	57.7%	7.1%	95.3%
U.S. Long Govt/Credit	3.4%	7.1%	0.0%
U.S. TIPS	3.5%	7.1%	0.0%
U.S. High Yield	4.5%	7.1%	0.4%
EMD (hard)	4.0%	7.1%	0.0%
Global Convertibles H	3.8%	7.1%	3.2%

Figure 2.10: Comparing asset class allocations implied by 3 different strategies

Next, we transition from equal-asset risk to equal-factor risk in our analysis. Lohre, Opfer & Orszag (2014) provide a detailed discussion on the properties of equal factor risk portfolios, using perfectly uncorrelated statistical factors. We diverge from the literature here to consider our imperfect (correlations not equal to 0), yet simple and intuitive multi-asset factors constructed earlier. Unlike the statistical factors, the time series factors constructed from asset class indexes, retain their meaning always, while having the added benefit of being computationally simple as well as easy to interpret and explain to clients and financial advisors.

Given these results and observations, we came up with the idea of adding a zero-investment active equal-factor risk sleeve to the traditional portfolios (represented by the peer average portfolios here) to try to create a more diversified portfolio at the margin with better performance metrics. An equal-factor-risk portfolio sleeve helps bridge the gap between pure risk budgeting (with unrealistic allocations and inability to scale up risk without the use of leverage) and a traditional portfolio (with concentrated factor risks). This zero-investment sleeve only depends on the factor covariance matrix and is independent of the baseline portfolio. Figure 2.11 shows that the constructed sleeve enhances a portfolio's Sharpe ratio across all risk categories, and its effects can be scaled up depending on the tracking error budget.

		Ann. Return	Cum. Return	Ann. Vol.	Sharpe Ratio
Peer Average Portfolio	US FA 15-30	2.2%	6.9%	3.2%	0.700
	US FA 30-50	3.1%	9.7%	4.6%	0.677
	US FA 50-70	4.5%	14.0%	7.1%	0.624
	US FA 70-85	4.7%	14.7%	9.2%	0.507
	US FA 85-100	5.9%	18.9%	11.0%	0.541
Peer Average Portfolio with 0.5% TE active sleeve	US FA 15-30	2.1%	6.6%	2.9%	0.741
	US FA 30-50	3.0%	9.4%	4.2%	0.720
	US FA 50-70	4.4%	13.7%	6.7%	0.654
	US FA 70-85	4.6%	14.5%	8.8%	0.526
	US FA 85-100	5.9%	18.7%	10.5%	0.559
Peer Average Portfolio with 1% TE active sleeve	US FA 15-30	2.2%	6.8%	2.4%	0.924
	US FA 30-50	3.1%	9.7%	3.5%	0.899
	US FA 50-70	4.5%	14.1%	5.9%	0.768
	US FA 70-85	4.8%	15.0%	7.9%	0.603
	US FA 85-100	6.0%	19.3%	9.6%	0.628
Peer Average Portfolio with 2% TE active sleeve	US FA 15-30	2.5%	7.6%	3.3%	0.741
	US FA 30-50	3.4%	10.4%	4.7%	0.708
	US FA 50-70	4.7%	14.7%	7.3%	0.646
	US FA 70-85	4.9%	15.5%	9.4%	0.525
	US FA 85-100	6.2%	19.7%	11.1%	0.557

Figure 2.11: Adding an active factor-risk parity sleeve to peer average portfolios

2.6 Conclusion

A multi-asset factor model is a useful tool for estimating peer exposures, especially in areas where holdings based information is lacking. Historically, factor returns have had low, robust correlations when compared to asset class returns and hence less susceptible to multi-collinearity issues. We constructed a multi-asset time-series factor model using readily available asset class index returns, making a trade-off between intuitive (but low-correlated factors) vs. unintuitive but uncorrelated statistical factors. By defining factors using long-short portfolios of asset classes, we ensure that the factor definitions are the same over time unlike statistical factors. Applying the multi-asset factor model to Morningstar U.S. multi-asset fund allocation peer-average returns, we are able to infer that the peers have 1) domestic bias in equities relative to market cap weights, 2) positive exposure to equity size, commodities, infrastructure, bond duration and credit factors, and 3) significant total portfolio risk concentrated in the global equity factor. Furthermore, the peers are dynamic over time and regular assessments are warranted. The caveat in this application is that in using the historical peer data, the estimated exposures are inherently backward-looking and the data is subject to survivorship bias in the funds.

We then use the multi-asset factor model to construct total and active (vs. estimated peer average portfolio) equal-asset risk and equal-factor risk portfolios, with a long-only constraint for a fairer comparison. The equal asset-risk portfolio had a better Sharpe ratio compared to the peer-average portfolios and the equal-weighted portfolio in the last 3 years. Lastly, we construct active zero-investment factor portfolios (for different risk or tracking error budget) and find that they enhance performance for all 5 risk categories of the estimated peer average portfolios. By using exponentially weighted regression that gives more importance to more recent data, the estimated risk attribution is made more relevant for a forward-looking investor.

Chapter 3. FITTING TIME SERIES FACTOR MODELS

The purpose of this vignette is to demonstrate the use of the $fitTsfm$ function and related control, analysis and plot functions in the $factorAnalytics$ package.

3.1 Overview

3.1.1 *Load Package*

The latest version of the $factorAnalytics$ package used in this vignette is hosted in the publicly available GitHub repository https://github.com/sangeeuw/factorAnalytics. There are plans for further updates to the package before its moved back to R-Forge and released on CRAN later this year. The package can be installed from GitHub as shown below.

```
library(devtools)
install_github("sangeeuw/factorAnalytics")
```

```
# load the package and its dependencies
library(factorAnalytics)
options(digits=3)
```

3.1.2 *Summary of Related Functions*

Here's a list of the functions and methods demonstrated in this vignette:

- $fitTsfm\ (asset.names,\ factor.names,\ data,\ fit.method,\ variable.selection,\ \ldots)$: Fits a time series factor model for one or more asset returns or excess returns using time series regression. Least squares (LS), discounted least squares (DLS) and robust regression fitting are possible. Variable selection methods include $stepwise$, $subsets$ and

$lars$. An object of class $tsfm$ containing the fitted objects, estimated coefficients, R^2 and residual volatility is returned.

- $coef\ (object,\ \ldots)$: Returns a $data.frame$ containing the coefficients (intercept and factor betas) for all assets fit by the $tsfm$ object.
- $fitted\ (object,\ \ldots)$: Returns an xts data object of fitted asset returns from the factor model for all assets.
- $residuals\ (object,\ \ldots)$: Returns an xts data object of residuals from the fitted factor model for all assets.
- $fmCov\ (object,\ use,\ \ldots)$: Returns the $N \times N$ symmetric covariance matrix for asset returns based on the fitted factor model. use specifies how missing values are to be handled.
- $fmSdDecomp\ (object,\ factor.cov,\ use,\ \ldots)$: Returns a list containing the standard deviation of asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. use specifies how missing values are to be handled.
- $fmVaRDecomp\ (object,\ factor.cov,\ p,\ type,\ use,\ \ldots)$: Returns a list containing the value-at-risk (VaR) for asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. $type$ specifies if VaR computation should be non-parametric (sample quantile) or based on a Normal distribution. And, p specifies the confidence level.

- $fmEsDecomp\ (object,\ factor.cov,\ p,\ type,\ use,\ \ldots)$: Returns a list containing the expected shortfall (ES) for asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. $type$ specifies if ES computation should be non-parametric (sample quantile) or based on a Normal distribution. And, p specifies the confidence level.
- $plot\ (x,\ \ldots)$: The plot method for class $tsfm$ can be used for plotting factor model characteristics of a group of assets (default) or an individual asset. The user can select the type of plot either from the menu prompt or directly via argument $which$. In case multiple plots are needed, the menu is repeated after each plot (enter 0 to exit). User can also input a numeric vector of plot options via $which$.
- $predict\ (object,\ newdata,\ \ldots)$: The predict method for class $tsfm$ returns a vector or matrix of predicted values for a new data sample or simulated values.
- $summary\ (object,\ se.type,\ \ldots)$: The summary method for class $tsfm$ returns an object of class $summary.tsfm$ containing the summaries of the fitted lm, $lmRob$ or $lars$ objects and the chosen type (HC/HAC) of standard errors and t-statistics to display. Printing the factor model summary object displays the call, coefficients (with standard errors and t-statistics), R^2 and residual volatility (under the homo-skedasticity assumption) for all assets.

3.1.3 *Data*

The following examples primarily use the $managers$ dataset from the $PerformanceAnalytics$ package. It's an xts data object with 132 observations on 10 variables; frequency is monthly.

```
data(managers)
colnames(managers)

##  [1] "HAM1"        "HAM2"        "HAM3"        "HAM4"        "HAM5"
##  [6] "HAM6"        "EDHEC.LS.EQ" "SP500.TR"    "US.10Y.TR"   "US.3m.TR"

range(index(managers))

## [1] "1996-01-31" "2006-12-31"
```

In the examples below, the monthly returns for the six hypothetical asset managers (HAM1 through HAM6) will be the explained asset returns. Columns 7 through 9, composed of the EDHEC Long-Short Equity hedge fund index, the S&P 500 total returns, and the total return series for the US Treasury 10-year bond will serve as explanatory factors. The last column (US 3-month T-bill) can be considered as the risk-free rate. The series have unequal histories in this sample and $fitTsfm$ removes asset-wise incomplete cases (asset's return data combined with respective factors' return data) before fitting a factor model.

```
asset.names <- colnames(managers[,1:6])
factor.names <- colnames(managers[,7:9])
mkt.name <- "SP500.TR"
rf.name <- "US.3m.TR"
```

Typically, factor models are fit using excess returns. If the asset and factor returns are not in excess return form, $rf.name$ can be specified to convert returns into excess returns. Similarly, market returns can be specified via $mkt.name$ to add market-timing factors to the factor model. The $CommonFactors$ dataset in the $factorAnalytics$ package also provides a collection of common factors as both monthly ($factors.M$) and quarterly ($factors.Q$) time series. Refer to the help file for the dataset for more information.

```
data(CommonFactors)
names(factors.Q)

## [1] "SP500"         "GS10TR"        "USD.Index"     "Term.Spread"
```

```
## [5] "Credit.Spread" "dVIX"          "TED.Spread"    "OILPRICE"
## [9] "TB3MS"

range(index(factors.Q))

## [1] "1997-03-31" "2014-03-31"
```

3.2 Fitting a Time Series Factor Model

In a time-series or macroeconomic factor model, observable economic time series such as industrial production growth rate, interest rates, market returns and inflation are used as common factors that contribute to asset returns. For example, the famous single index model by Sharpe (1964) uses the market excess return as the common factor (captures economy-wide or market risk) for all assets and the unexplained returns in the error term represents the non-market firm specific risk. On the other hand, Chen et al. (1986) uses a multi-factor model to find that surprise inflation, the spread between long and short-term interest rates and between high and low-grade bonds are significantly priced, while the market portfolio, aggregate consumption risk and oil price risk are not priced separately. Chapter 15 from Zivot and Jia-hui (2006) serves as a good reference for a description of the different multi-factor models, estimation methods and relevant examples using S-PLUS.

Let's look at the arguments for $fitTsfm$.

```
args(fitTsfm)

## function (asset.names, factor.names, mkt.name = NULL, rf.name = NULL,
##     data = data, fit.method = c("LS", "DLS", "Robust"), variable.selection = c("none",
##         "stepwise", "subsets", "lars"), control = fitTsfm.control(...),
##     ...)
## NULL
```

The default model fitting method is LS regression and the default variable selection method is $none$ (that is, all factors are included in the model). The different model fitting and variable selection options are described in sections 3.2.3 and 3.2.4.

The default for $rf.name$ and $mkt.name$ are $NULL$. If $rf.name$ is not specified by the user, perhaps because the data is already in excess return form, then no risk-free rate adjustment is made. Similarly, if $mkt.name$ is not specified, market-timing factors are not added to the model. All other optional control parameters passed through the ellipsis are processed and assimilated internally by $fitTsfm.control$. More on that in section 3.2.5.

3.2.1 *Single Index Model*

Here's an implementation of the single index model for the 6 hypothetical assets described in section 3.1.3 earlier. Since $rf.name$ was included, excess returns are computed and used for all variables during model fitting.

```
# Single Index Model using SP500
fit.singleIndex <- fitTsfm(asset.names=asset.names, factor.names="SP500.TR",
                           rf.name="US.3m.TR", data=managers)
```

The resulting object, $fit.singleIndex$, has the following attributes.

```
class(fit.singleIndex)

## [1] "tsfm"

names(fit.singleIndex)

##  [1] "asset.fit"          "alpha"              "beta"
##  [4] "r2"                 "resid.sd"           "call"
##  [7] "data"               "asset.names"        "factor.names"
## [10] "mkt.name"           "fit.method"         "variable.selection"
```

The component $asset.fit$ contains a list of lm objects[15], one for each asset. The estimated coefficients[16] are in $alpha$ and $beta$. R^2 and residual standard deviations are in $r2$ and $resid.sd$ respectively. The remaining components contain the input choices and the data.

```
fit.singleIndex # print the fitted "tsfm" object

##
## Call:
## fitTsfm(asset.names = asset.names, factor.names = "SP500.TR",
##     rf.name = "US.3m.TR", data = managers)
##
## Model dimensions:
## Factors  Assets Periods
##       1       6     132
##
## Regression Alphas:
##                  HAM1    HAM2    HAM3    HAM4    HAM5    HAM6
## (Intercept) 0.00577 0.00909 0.00622 0.00403 0.00173 0.00784
##
```

[15] The fitted objects can be of class lm, $lmRob$ or $lars$ depending on the fit and variable selection methods.

[16] Refer to the summary method in section 2.2.6 for standard errors, degrees of freedom, t-statistics etc.

```
## Factor Betas:
##           HAM1  HAM2  HAM3  HAM4  HAM5  HAM6
## SP500.TR 0.39 0.338 0.552 0.691 0.321 0.324
##
## R-squared values:
##   HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.4339 0.1673 0.4341 0.3148 0.0829 0.2601
##
## Residual Volatilities:
##   HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0193 0.0334 0.0274 0.0443 0.0441 0.0206
```

Figure 3.1 shows the single factor linear fits for the assets. (Plot options are explained later in section 3.4.)

```
# plot asset returns vs factor returns for the single factor models
plot(fit.singleIndex, which=12, f.sub=1)
```

3.2.2 *Market Timing Models*

In the following example, we fit the Henriksson & Merton (1981) market timing model, using the S&P 500 as the market. Market timing accounts for the price movement of the general stock market relative to fixed income securities. The function $fitTsfm.MT$, a wrapper to $fitTsfm$, includes $down.market = \max(0, R_f - R_m)$ as a factor. To test market timing ability, this factor can be added to the single index model as shown below. The coefficient of this down-market factor can be interpreted as the number of "free" put options on the market provided by the manager's market-timing skills. That is, a negative value for the regression estimate would imply a negative value for market timing ability of the manager. Note: the user needs to specify which column in $data$ corresponds to the market returns using argument $mkt.name$.

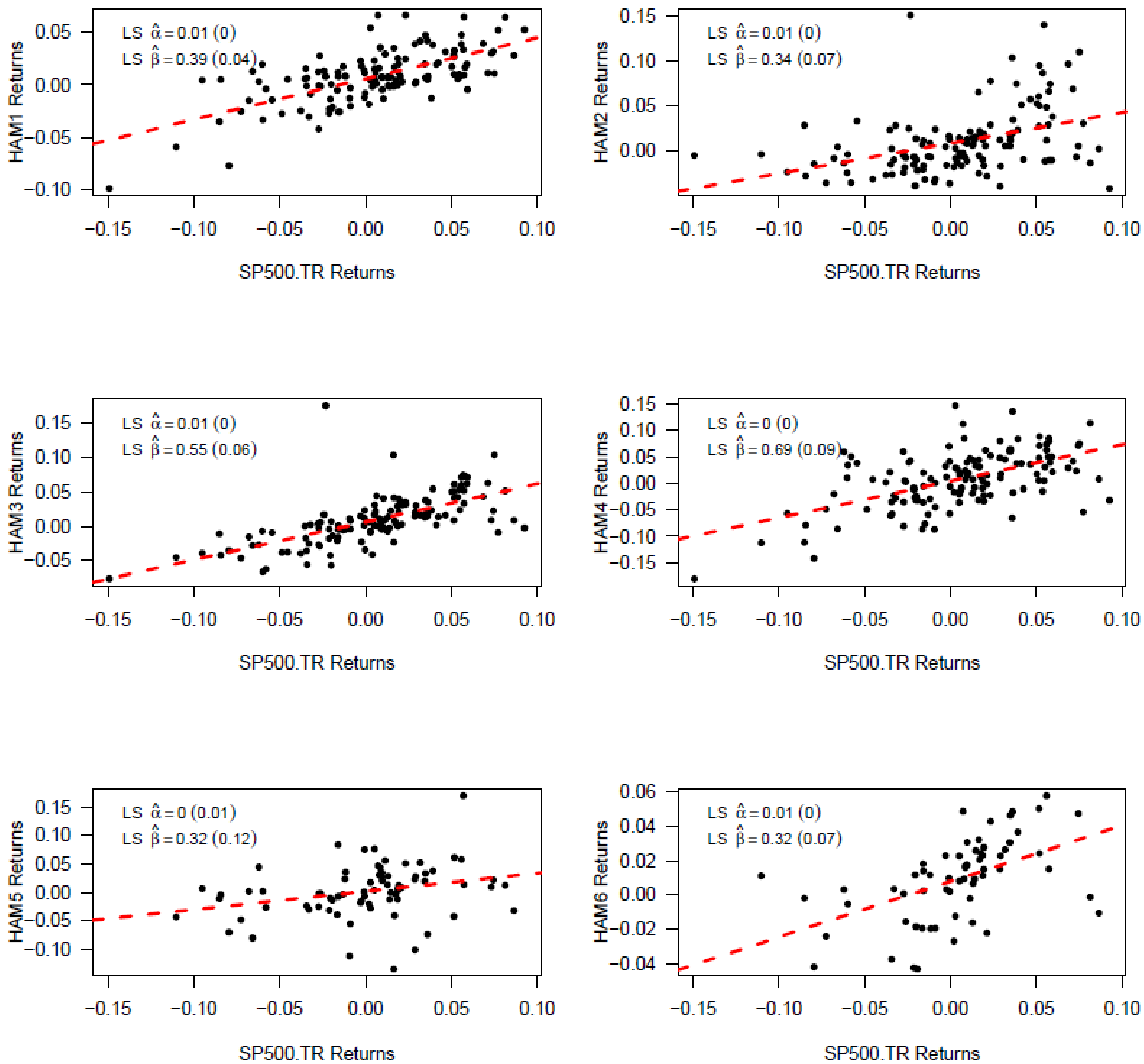

Figure 3.1: Single Index Model: Asset returns vs. Factor Returns

```
# Henriksson-Merton's market timing model
fit.mktTiming <-  fitTsfmMT(asset.names=asset.names, mkt.name="SP500.TR",
                            rf.name="US.3m.TR", data=managers)
t(fit.mktTiming$beta)

##                   HAM1  HAM2    HAM3   HAM4    HAM5  HAM6
## SP500.TR        0.450 0.119  0.5615  0.954  0.3564 0.275
## down.market -0.125 0.457 -0.0191 -0.549 -0.0845 0.106
```

```
fit.mktTiming$r2

##    HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.4382 0.1960 0.4341 0.3340 0.0834 0.2631

fit.mktTiming$resid.sd

##    HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0193 0.0330 0.0275 0.0438 0.0444 0.0207
```

3.2.3 *Fit Methods*

The default fit method is LS regression. The next example performs LS regression using all 3 available factors in the dataset. Notice that the R^2 values have improved considerably when compared to the single index model as well as the market-timing model.

```
fit.ols <- fitTsfm(asset.names=asset.names, factor.names=factor.names,
                   rf.name="US.3m.TR", data=managers)
fit.ols$beta

##      EDHEC.LS.EQ SP500.TR US.10Y.TR
## HAM1       0.268    0.287   -0.2302
## HAM2       1.547   -0.195    0.0504
## HAM3       1.251    0.131    0.1437
## HAM4       1.222    0.273   -0.1391
## HAM5       1.621   -0.184    0.2712
## HAM6       1.250   -0.175   -0.1739

fit.ols$r2

##  HAM1  HAM2  HAM3  HAM4  HAM5  HAM6
## 0.501 0.514 0.657 0.413 0.232 0.564

fit.ols$resid.sd

##    HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0189 0.0253 0.0216 0.0427 0.0409 0.0161
```

Other options include discounted least squares (DLS) and robust regression ($Robust$). DLS is least squares regression using exponentially discounted weights and accounts for time variation in coefficients. Robust regression is resistant to outliers.

```
fit.robust <- fitTsfm(asset.names=asset.names, factor.names=factor.names,
                      rf.name="US.3m.TR", data=managers, fit.method="Robust")
fit.robust$beta

##      EDHEC.LS.EQ SP500.TR US.10Y.TR
## HAM1       0.157    0.277   -0.1635
## HAM2       1.151   -0.116   -0.0524
## HAM3       0.781    0.240    0.0500
## HAM4       1.613    0.209   -0.0829
## HAM5       1.341   -0.117    0.1920
## HAM6       1.255   -0.180   -0.1778

fit.robust$r2

##  HAM1  HAM2  HAM3  HAM4  HAM5  HAM6
## 0.313 0.231 0.461 0.316 0.240 0.470

fit.robust$resid.sd

##   HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0179 0.0202 0.0152 0.0370 0.0285 0.0156
```

Notice the lower R^2 values and smaller residual volatilities with robust regression. Figure 3.2 and Figure 3.3 give a graphical comparison of the fitted returns for asset "HAM3" and residual volatilities from the factor model fits. Figure 3.3 depicts the smaller influence that the volatility of Jan 2000 has on the robust regression.

```
par(mfrow=c(1,2))
plot(fit.ols, which=5, xlim=c(0,0.045), sub="LS")
plot(fit.robust, which=5, xlim=c(0,0.045), sub="Robust")
```

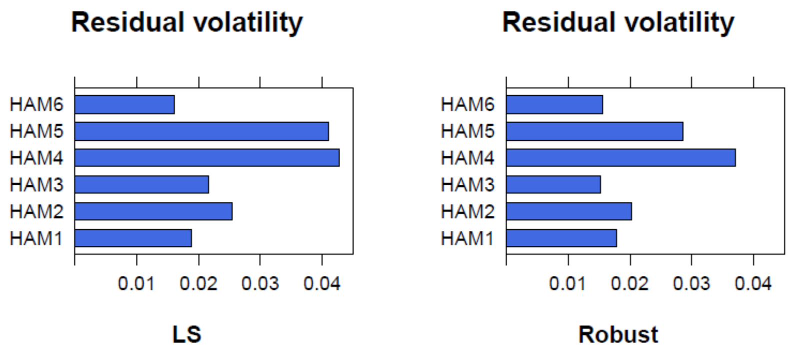

Figure 3.2: Residual Volatility: LS (left) vs. Robust (right)

```
par(mfrow=c(2,1))
plot(fit.ols, plot.single=TRUE, which=1, asset.name="HAM3")
mtext("LS", side=3)
plot(fit.robust, plot.single=TRUE, which=1, asset.name="HAM3")
mtext("Robust", side=3)
```

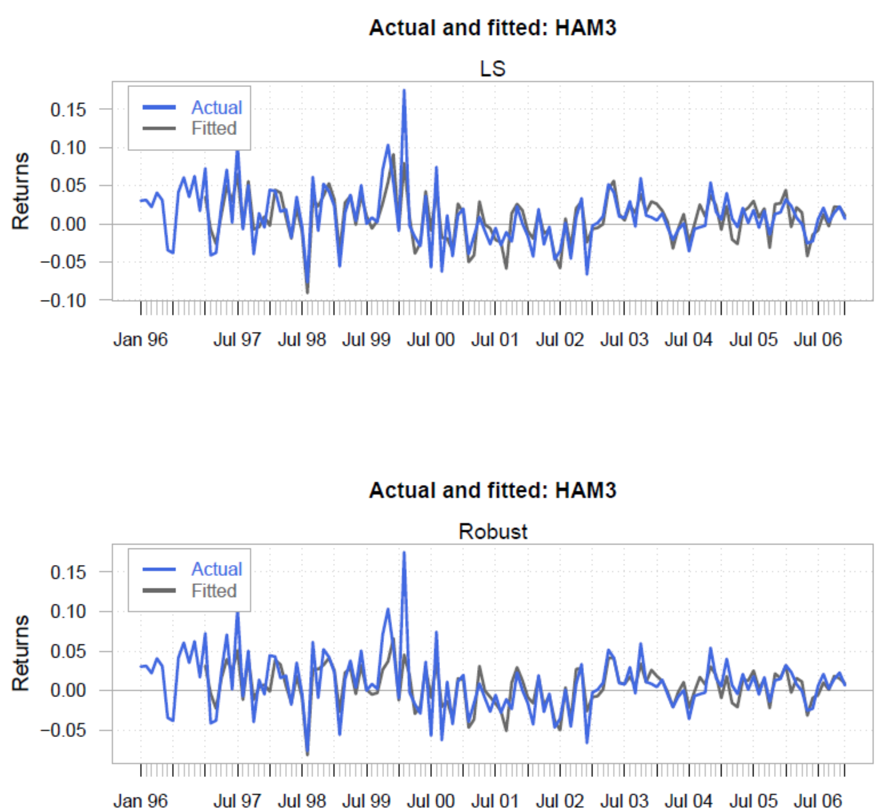

Figure 3.3: HAM 3 returns: LS (top) vs. Robust (bottom)

3.2.4 *Variable Selection*

Though the R^2 values improved by adding more factors in $fit.ols$ (compared to the single index model), one might prefer to employ variable selection methods such as $stepwise$, $subsets$ or $lars$ to avoid over-fitting. The method can be selected via the $variable.selection$ argument. The default $none$, uses all the factors and performs no variable selection.

Specifying $stepwise$ selects traditional stepwise[17] least squares or robust regression using $step$ or $step.lmRob$ respectively. Starting from the given initial set of factors, factors are added (or subtracted) only if the regression fit, as measured by the Bayesian Information Criterion (BIC) or Akaike Information Criterion (AIC)[18], improves.

Specifying $subsets$ enables subsets selection using $regsubsets$. The best performing subset of any given size or within a range of subset sizes is chosen. Different methods such as exhaustive search (default), forward or backward stepwise, or sequential replacement can be employed. Finally, $lars$ corresponds to least angle regression using lars with variants $lasso$ (default), lar, $forward.stagewise$ or $stepwise$. The next example uses the $lars$ variable selection method. The default type and criterion used are $lasso$ and the Cp statistic.

[17] The direction for stepwise search can be one of "forward", "backward" or "both". See the help file for more details.

[18] AIC is the default. When the additive constant can be chosen so that AIC is equal to Mallows's Cp, this is done. The optional control parameter k can be used to switch to BIC instead.

```
fit.lars <- fitTsfm(asset.names=asset.names, factor.names=factor.names,
                    data=managers, rf.name="US.3m.TR",
                    variable.selection="lars")
fit.lars

##
## Call:
## fitTsfm(asset.names = asset.names, factor.names = factor.names,
##     rf.name = "US.3m.TR", data = managers, variable.selection = "lars")
##
## Model dimensions:
## Factors  Assets Periods
##       3       6     132
##
## Regression Alphas:
##         HAM1    HAM2     HAM3      HAM4     HAM5    HAM6
## [1,] 0.00537 0.00151 -0.00124 -0.000983 -0.00284 0.00404
##
## Factor Betas:
##               HAM1   HAM2  HAM3  HAM4  HAM5   HAM6
## EDHEC.LS.EQ  0.268  1.340 1.251 1.129 1.176  1.250
## SP500.TR     0.287 -0.105 0.131 0.239     . -0.175
## US.10Y.TR   -0.230      . 0.144     . 0.242 -0.174
##
## R-squared values:
##  HAM1  HAM2  HAM3  HAM4  HAM5  HAM6
## 0.501 0.506 0.657 0.405 0.221 0.564
##
## Residual Volatilities:
##   HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0188 0.0253 0.0215 0.0426 0.0407 0.0159
```

Using the same set of factors for comparison, let's fit another model using the $subsets$ variable selection method. Here, the best subset of size 2 for each asset is chosen by specifying $nvmin =$

$nvmax = 2$. Note that when $nvmin < nvmax$, the best subset is chosen from a range of subset sizes $[nvmin, nvmax]$. Default is $nvmin = 1$.

```
(fit.sub <- fitTsfm(asset.names=asset.names, factor.names=factor.names,
                    data=managers, rf.name="US.3m.TR",
                    variable.selection="subsets", nvmin=2, nvmax=2))

##
## Call:
## fitTsfm(asset.names = asset.names, factor.names = factor.names,
##     rf.name = "US.3m.TR", data = managers, variable.selection = "subsets",
##     nvmin = 2, nvmax = 2)
##
## Model dimensions:
## Factors  Assets Periods
##       3       6     132
##
## Regression Alphas:
##                HAM1    HAM2      HAM3     HAM4     HAM5    HAM6
## (Intercept) 0.00614 0.00063 -0.000903 -0.00183 -0.00346 0.00366
##
## Factor Betas:
##               HAM1   HAM2  HAM3  HAM4  HAM5   HAM6
## EDHEC.LS.EQ      .  1.545 1.245 1.228 1.294  1.221
## SP500.TR     0.372 -0.199 0.119 0.285     . -0.119
## US.10Y.TR   -0.232      .     .     . 0.338      .
##
## R-squared values:
##  HAM1  HAM2  HAM3  HAM4  HAM5  HAM6
## 0.467 0.514 0.651 0.410 0.224 0.542
##
## Residual Volatilities:
##   HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0188 0.0253 0.0217 0.0426 0.0409 0.0163
```

Comparing the coefficients and R^2 values from the two models, we find that the method using more factors produces higher R^2 values as expected. However, when both $lars$ and $subsets$ chose the same number of factors, $lars$ fits have slightly higher R^2 values. Figure 3.4 and Figure 3.5 display the factor betas from the two fits.

Remarks:

- Variable selection methods $stepwise$ and $subsets$ can be combined with any of the fit methods, LS, DLS or $Robust$. If variable selection method selected is $lars$, $fit.method$ will be ignored.
- Refer to the next section on $fitTsfm.control$ for more details on the control arguments that can be passed to the different variable selection methods.

```
plot(fit.sub, which=2, f.sub=1:3)
```

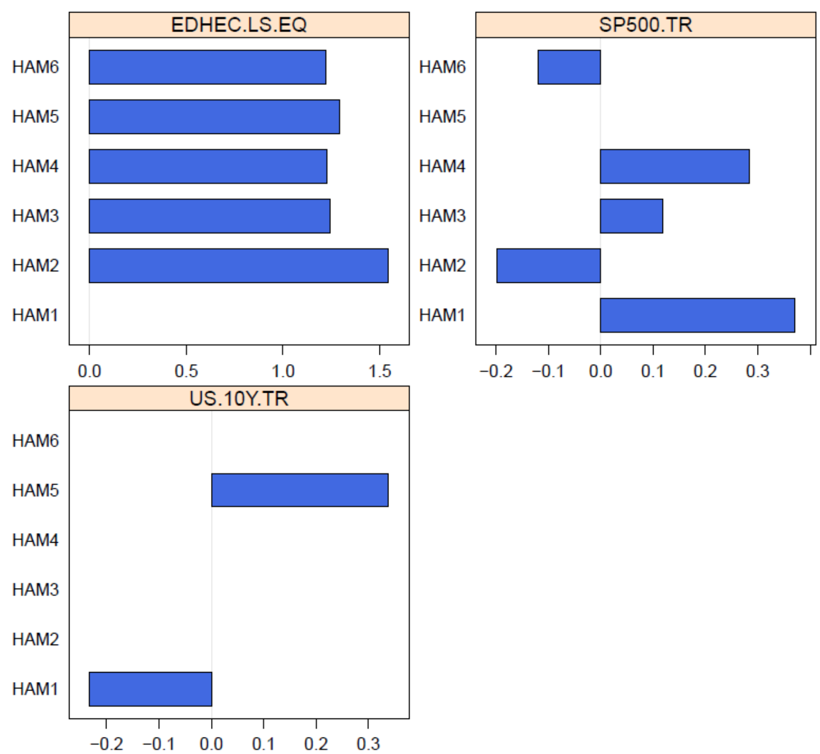

Figure 3.4: Factor betas: $fit.sub$

```
plot(fit.lars, which=2, f.sub=1:3)
```

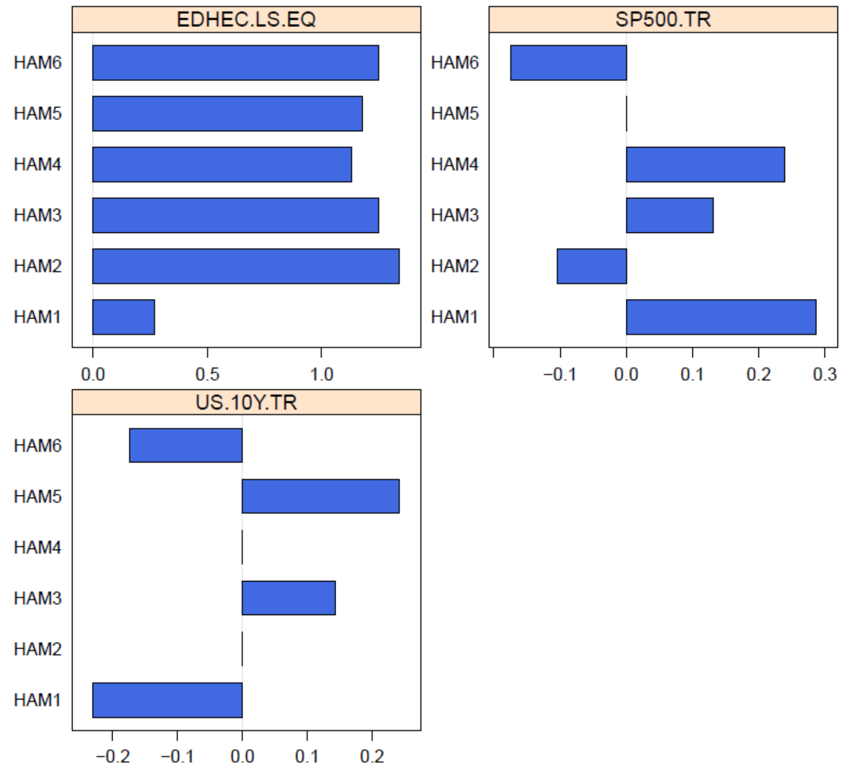

Figure 3.5: Factor betas: $fit.lars$

3.2.5 *Control Function for $fitTsfm$*

Since $fitTsfm$ calls many different regression fitting and variable selection methods, it made sense to collect all the optional controls for these functions and process them via $fitTsfm.control$. This function is meant to be used internally by $fitTsfm$ when arguments are passed to it via the ellipsis. The use of control parameters was demonstrated with $nvmin$ and $nvmax$ in the $fit.sub$ example earlier.

For easy reference, here's a list of control parameters accepted and passed by $fitTsfm$ to their respective model fitting (or) model selection functions in other packages. See the corresponding help files for more details on each parameter.

- lm: "weights","model","x","y","qr"
- $lmRob$: "weights","model","x","y","nrep","efficiency","mxr","mxf","mxs","trace"
- $step$: "scope","scale","direction","trace","steps","k"
- $regsubsets$: "weights","nvmax","force.in","force.out","method","really.big"
- $lars$: "type","normalize","eps","max.steps","trace"
- $cv.lars$: "K","type","normalize","eps","max.steps","trace"

There are 3 other significant arguments that can be passed through the ... argument to $fitTsfm$.

- $decay$: Determines the decay factor for DLS fit method, which corresponds to exponentially weighted least squares, with weights adding to 1.
- $nvmin$: The lower limit for the range of subset sizes from which the best model (BIC) is found when performing "subsets" selection. Note that the upper limit was already passed to $regsubsets$ function. By specifying $nvmin = nvmax$, users can obtain the best model of a given size (meaningful to those who want a parsimonious model, or to compare with a different model of the same size, or perhaps to avoid over-fitting/ data dredging etc.).
- $lars.criterion$: An option (one of Cp or cv) to assess model selection for the $lars$ variable selection method. Cp is Mallows's Cp statistic and cv is K-fold cross-validated mean squared prediction error.

3.2.6 *S3 Generic Methods*

Many useful generic accessor functions are available for $tsfm$ fit objects.

```
methods(class="tsfm")

##  [1] coef          fitted        fmCov         fmEsDecomp    fmSdDecomp
##  [6] fmVaRDecomp   plot          portEsDecomp  portSdDecomp  portVaRDecomp
## [11] portVolDecomp predict       print         repRisk       residuals
## [16] riskDecomp    summary
## see '?methods' for accessing help and source code
```

$coef()$ returns a matrix of estimated model coefficients including the intercept. $fitted()$ returns an xts data object of the part of observed asset returns explained by the factor model. $residuals()$ returns an xts data object with the part of observed asset returns not explained by the factor model. $predict()$ uses the fitted factor model to estimate asset returns given a set of new or simulated factor return data.

$summary()$ prints standard errors and t-statistics for all estimated coefficients in addition to R^2 values and residual volatilities. Argument $se.type$, one of $Default$, HC or HAC, allows for heteroskedasticity and auto-correlation consistent estimates and standard errors whenever possible. A $summary.tsfm$ object is returned which contains a list of summary objects returned by lm, $lmRob$ or $lars$ for each asset fit. Note: Standard errors are currently not available for the $lars$ variable selection method, as there seems to be no consensus on a statistically valid method of calculating standard errors for the lasso predictions.

Factor model covariance and risk decomposition functions are explained in section 3.3 and the plot method is discussed separately in section 3.4. Here are some examples using the time series factor models fitted earlier.

```
# all estimated coefficients from the LS fit using all 3 factors
coef(fit.ols)

##      (Intercept) EDHEC.LS.EQ SP500.TR US.10Y.TR
## HAM1    0.005371       0.268    0.287   -0.2302
## HAM2    0.000512       1.547   -0.195    0.0504
## HAM3   -0.001240       1.251    0.131    0.1437
## HAM4   -0.001503       1.222    0.273   -0.1391
## HAM5   -0.004447       1.621   -0.184    0.2712
```

```
## HAM6    0.004038       1.250   -0.175   -0.1739

# compare returns data with fitted and residual values for HAM1 from fit.lars
HAM1.ts <- merge(fit.lars$data[,1], fitted(fit.lars)[,1], residuals(fit.lars)[,1])
colnames(HAM1.ts) <- c("HAM1.return","HAM1.fitted","HAM1.residual")
tail(HAM1.ts)

##            HAM1.return HAM1.fitted HAM1.residual
## 2006-07-31    -0.01863     0.00131      -0.01994
## 2006-08-31     0.01169     0.00878       0.00291
## 2006-09-30     0.00224     0.00870      -0.00646
## 2006-10-31     0.03889     0.01735       0.02154
## 2006-11-30     0.00740     0.01152      -0.00412
## 2006-12-31     0.00709     0.01563      -0.00854

# summary for fit.sub computing HAC standard erros
summary(fit.sub, se.type="HAC")

##
## Call:
## fitTsfm(asset.names = asset.names, factor.names = factor.names,
##     rf.name = "US.3m.TR", data = managers, variable.selection = "subsets",
##     nvmin = 2, nvmax = 2)
##
## Factor Model Coefficients:
##
```

```
## Asset1: HAM1
## (HAC Standard Errors & T-stats)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.00614    0.00181    3.40   0.0009 ***
## SP500.TR     0.37163    0.04930    7.54  7.4e-12 ***
## US.10Y.TR   -0.23242    0.07091   -3.28   0.0013 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.467, Residual Volatility: 0.0188
```

```
##
## Asset2: HAM2
## (HAC Standard Errors & T-stats)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.00063    0.00239    0.26    0.792
## EDHEC.LS.EQ  1.54468    0.24583    6.28  5.9e-09 ***
## SP500.TR    -0.19897    0.10595   -1.88    0.063 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.514, Residual Volatility: 0.0253
##
## Asset3: HAM3
## (HAC Standard Errors & T-stats)
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.000903   0.001871   -0.48  0.63029
## EDHEC.LS.EQ  1.244687   0.328379    3.79  0.00024 ***
## SP500.TR     0.119303   0.124063    0.96  0.33822
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.651, Residual Volatility: 0.0217
##
```

```
## Asset4: HAM4
## (HAC Standard Errors & T-stats)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.00183    0.00428   -0.43  0.66977
## EDHEC.LS.EQ  1.22790    0.34069    3.60  0.00046 ***
## SP500.TR     0.28467    0.12115    2.35  0.02046 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.41, Residual Volatility: 0.0426
##
## Asset5: HAM5
## (HAC Standard Errors & T-stats)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.00346    0.00396   -0.87     0.38
## EDHEC.LS.EQ  1.29442    0.28326    4.57  1.9e-05 ***
## US.10Y.TR    0.33791    0.21106    1.60     0.11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.224, Residual Volatility: 0.0409
##
## Asset6: HAM6
## (HAC Standard Errors & T-stats)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.00366    0.00286    1.28     0.20
## EDHEC.LS.EQ  1.22084    0.21548    5.67  4.2e-07 ***
## SP500.TR    -0.11910    0.10905   -1.09     0.28
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-squared: 0.542, Residual Volatility: 0.0163
```

3.3 FACTOR MODEL COVARIANCE AND RISK DECOMPOSITION

3.3.1 *Factor Model Covariance*

Following Zivot & Jia-hui (2006), $R_{i,t}$, the return on asset i $(i = 1, 2, \ldots, N)$ at time t $(t = 1, 2, \ldots, T)$, is fitted with a factor model of the form,

$$R_{i,t} = \alpha_i + \boldsymbol{\beta}_i{}' \boldsymbol{f}_t + \varepsilon_{i,t} \tag{3.1}$$

where, α_i is the intercept, $\boldsymbol{f}_t$ is a $K \times 1$ vector of factor returns at time t, $\boldsymbol{\beta}_i$ is a $K \times 1$ vector of factor exposures for asset i and the error terms $\varepsilon_{i,t}$ are serially uncorrelated across time and contemporaneously uncorrelated across assets so that $\varepsilon_{i,t} \sim iid\ (0, \sigma_i^2)$. Thus, the variance of asset i's return is given by,

$$var(R_{i,t}) = \boldsymbol{\beta}_i{}' var\ (\boldsymbol{f}_t) \boldsymbol{\beta}_i + \sigma_i^2 \tag{3.2}$$

And the $N \times N$ covariance matrix of asset returns is,

$$var(\boldsymbol{R}) = \boldsymbol{\Omega} = \boldsymbol{B}\ var(\boldsymbol{F})\ \boldsymbol{B} + \boldsymbol{D} \tag{3.3}$$

where, $\boldsymbol{R}$ is the $N \times T$ matrix of asset returns, $\boldsymbol{B}$ is the $N \times K$ matrix of factor betas, $\boldsymbol{F}$ is the $K \times T$ matrix of factor returns and $\boldsymbol{D}$ is a diagonal matrix with σ_i^2 along the diagonal.

$fmCov$ computes the factor model covariance from a fitted factor model. The covariance of factor returns is the sample covariance matrix by default, but the option exists for the user to specify their own. Options for handling missing observations include $pairwise.complete.obs$ (default), $everything$, $all.obs$, $complete.obs$ and $na.or.complete$.

```
fmCov(fit.sub)

##           HAM1     HAM2     HAM3     HAM4     HAM5     HAM6
## HAM1 0.000661 0.000257 0.000411 0.000527 0.000286 0.000231
## HAM2 0.000257 0.001297 0.000710 0.000807 0.000631 0.000545
## HAM3 0.000411 0.000710 0.001334 0.001024 0.000732 0.000601
## HAM4 0.000527 0.000807 0.001024 0.003051 0.000855 0.000689
## HAM5 0.000286 0.000631 0.000732 0.000855 0.002348 0.000529
## HAM6 0.000231 0.000545 0.000601 0.000689 0.000529 0.000720

# factor model return correlation plot
plot(fit.sub, which=8)
```

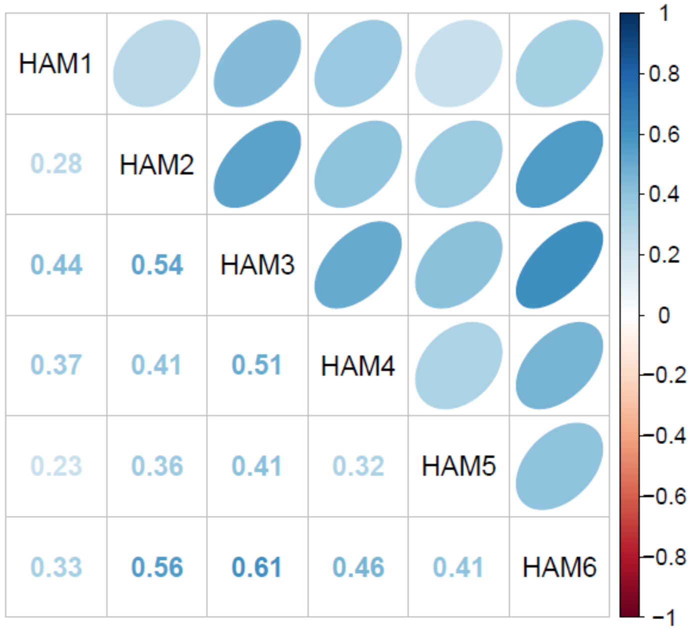

Figure 3.6: Factor model return correlation (pairwise complete observations)

3.3.2 *Standard Deviation Decomposition*

Following Meucci (2007), the standard deviation of asset i's return can be decomposed into the factor risk contributions using the factor model in equation 2.1 as shown below.

$$R_{i,t} = \boldsymbol{\beta}_i^{*\prime} \boldsymbol{f}_t^{*} \tag{3.4}$$

where, $\boldsymbol{\beta}_i^* = (\boldsymbol{\beta}_i' \ \sigma_i)$ and $\boldsymbol{f}_t^* = (\boldsymbol{f}_t' \ z_t)$, with $z_t \sim iid\ (0, 1)$ and σ_i is asset i's residual standard deviation. In other words, the residual is considered the $K + 1^{\text{th}}$ risk factor, where the exposure to the residual is the residual standard deviation, and the residual factor returns are assumed to be $iid \sim (0,1)$. By Euler's theorem, the standard deviation of asset i, σ_i, can be decomposed as:

$$\sigma_i = \sum_{k=1}^{K+1} cSd_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^* * mSd_{i,k}\right) \tag{3.5}$$

where, $cSd_{i,k}$ and $mSd_{i,k}$ are the component and marginal contributions to risk from the k^{th} factor. While the component contribution is the total contribution to risk from factor k, the marginal contribution to risk is the effect on the asset's standard deviation due to an incremental change in its exposure to the k^{th} factor, holding all else constant. Computing the component and marginal risk contributions is straight forward. The formulas are given below and details are in Meucci (2007).

$$\sigma_i = \sqrt{\boldsymbol{\beta}_i^{*\prime} cov(\boldsymbol{F}^*)\ \boldsymbol{\beta}_i^*} \tag{3.6}$$

$$\boldsymbol{mSd}_i = \frac{cov\left(\boldsymbol{F}^*\right) \boldsymbol{\beta}_i^*}{\sigma_i} \tag{3.7}$$

$$\boldsymbol{cSd}_i = \boldsymbol{\beta}_i^* \odot \boldsymbol{mSd}_i \tag{3.8}$$

The covariance term is approximated by the sample covariance and $\odot$ represents element-wise multiplication. $fmSdDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total standard deviation and component, marginal and percentage component contributions for each asset are returned.

```
decomp <- fmSdDecomp(fit.sub)
names(decomp)

## [1] "Sd.fm" "mSd"   "cSd"   "pcSd"

# get the factor model standard deviation for all assets
decomp$Sd.fm

##   HAM1   HAM2   HAM3   HAM4   HAM5   HAM6
## 0.0257 0.0360 0.0365 0.0552 0.0485 0.0268

# get the component contributions to Sd
decomp$cSd
```

```
##      EDHEC.LS.EQ SP500.TR US.10Y.TR Residuals
## HAM1      0.0000  0.01054  0.001365   0.01381
## HAM2      0.0218 -0.00354  0.000000   0.01772
## HAM3      0.0202  0.00339  0.000000   0.01291
## HAM4      0.0154  0.00689  0.000000   0.03289
## HAM5      0.0137  0.00000  0.000322   0.03447
## HAM6      0.0194 -0.00256  0.000000   0.00996

# get the marginal factor contributions to Sd
decomp$mSd

##      EDHEC.LS.EQ SP500.TR US.10Y.TR Residuals
## HAM1      0.0101   0.0284 -0.005874     0.733
## HAM2      0.0141   0.0178 -0.002272     0.701
## HAM3      0.0162   0.0284 -0.002941     0.594
## HAM4      0.0126   0.0242 -0.002367     0.772
## HAM5      0.0106   0.0165  0.000953     0.843
## HAM6      0.0159   0.0215 -0.002621     0.609
```

```
# get the percentage component contributions to Sd
decomp$pcSd

##      EDHEC.LS.EQ SP500.TR US.10Y.TR Residuals
## HAM1         0.0    41.00     5.309      53.7
## HAM2        60.6    -9.82     0.000      49.2
## HAM3        55.4     9.29     0.000      35.3
## HAM4        28.0    12.48     0.000      59.6
## HAM5        28.2     0.00     0.664      71.1
## HAM6        72.4    -9.54     0.000      37.1

# plot the percentage component contributions to Sd
plot(fit.sub, which=9, f.sub=1:3)
```

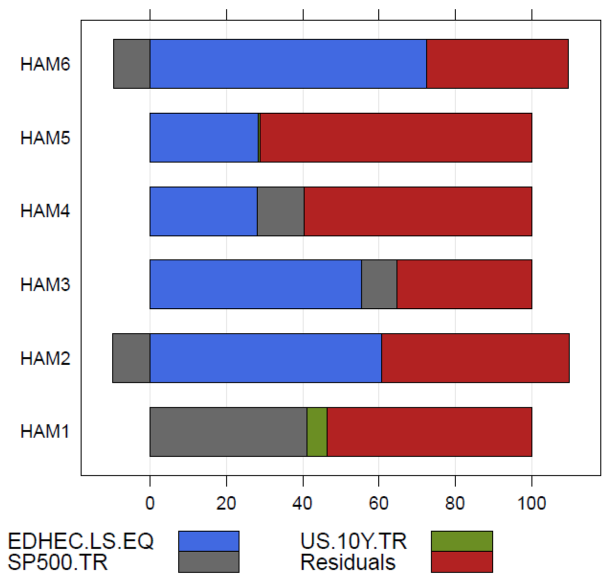

Figure 3.7: Percentage factor contribution to SD

3.3.3 *Value-at-Risk Decomposition*

Euler decomposition of return standard deviation shown above can also be applied to other risk measures such as value-at-risk (VaR) and expected shortfall (ES). The VaR version of equation 3.5 is given below. The value-at-risk of asset i can be decomposed as:

$$VaR_i = \sum_{k=1}^{K+1} cVaR_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^* * mVaR_{i,k}\right) \tag{3.9}$$

The marginal contribution to VaR_i is defined as the expectation of $\boldsymbol{F}^*$, conditional on the loss being equal to VaR_i. This is approximated as described in Epperlein and Smillie (2006) using a triangular smoothing kernel. $type$ gives the option to estimate VaR_i non-parametrically using the sample quantile (default) or assuming a normal distribution.

$fmVaRDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total VaR and component, marginal and percentage component contributions for each asset are returned.

```
# factor model VaR decomp using estimated factor return covariance (default)
# using tail probability = 10% and parametric (normal) VaR estimation
decomp1 <- fmVaRDecomp(fit.sub, p=0.10, type="normal")
names(decomp1)

## [1] "VaR.fm"     "n.exceed"   "idx.exceed" "mVaR"       "cVaR"
## [6] "pcVaR"

# get the factor model value-at-risk for all assets
decomp1$VaR.fm

##    HAM1    HAM2    HAM3    HAM4    HAM5    HAM6
## -0.0312 -0.0373 -0.0382 -0.0613 -0.0534 -0.0272

# print the number of VaR exceedences for all assets
decomp1$n.exceed

## HAM1 HAM2 HAM3 HAM4 HAM5 HAM6
##    6    3   13   11    7    4

# plot the percentage component contributions to VaR
plot(fit.sub, which=11, f.sub=1:3)
```

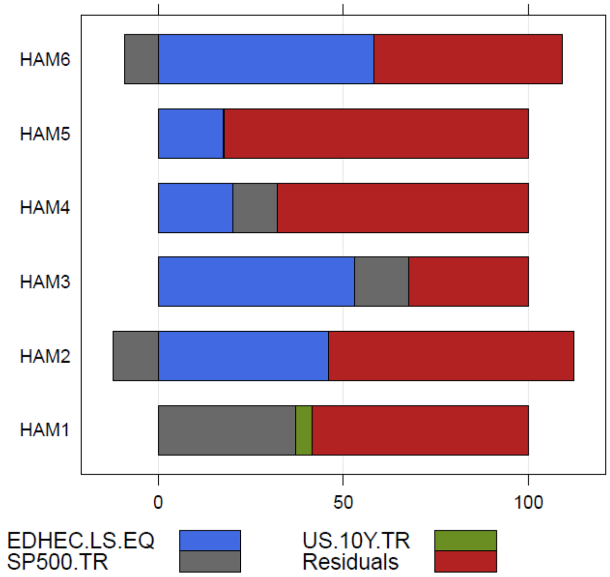

Figure 3.8: Percentage factor contribution to VaR

3.3.4 *Expected Shortfall Decomposition*

The expected shortfall (ES) version of equation 3.5 is given below. The expected shortfall of asset i can be decomposed as:

$$ES_i = \sum_{k=1}^{K+1} cES_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^* * mES_{i,k}\right) \tag{3.10}$$

The marginal contribution to ES_i is defined as the expectation of $\boldsymbol{F}^*$, conditional on the loss being less than or equal to ES_i. This is estimated as a sample average of the observations in that data window. Once again, $type$ gives the option to estimate ES_i non-parametrically using the sample quantile (default) or assuming a normal distribution.

$fmESDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total ES and component, marginal and percentage component contributions for each asset are returned.

```r
# using normal distr. for computing ES (default is non-param. sample quantile)
decomp2 <- fmEsDecomp(fit.sub, type="normal")
names(decomp2)
```

```
## [1] "ES.fm" "mES"   "cES"   "pcES"
```

```r
# get the factor model expected shortfall for all assets
decomp2$ES.fm
```

```
##    HAM1    HAM2    HAM3    HAM4    HAM5    HAM6
## -0.0548 -0.0831 -0.0840 -0.1234 -0.1087 -0.0626
```

```r
# get the component contributions to Sd
decomp2$cES
```

```
##      EDHEC.LS.EQ SP500.TR US.10Y.TR Residuals
## HAM1      0.0000 -0.02376  -0.00255   -0.0285
## HAM2     -0.0550  0.00838   0.00000   -0.0365
## HAM3     -0.0497 -0.00765   0.00000   -0.0266
## HAM4     -0.0396 -0.01575   0.00000   -0.0680
## HAM5     -0.0362  0.00000  -0.00104   -0.0714
## HAM6     -0.0477  0.00593   0.00000   -0.0208
```

```r
# get the marginal factor contributions to ES
decomp2$mES
```

```
##      EDHEC.LS.EQ SP500.TR US.10Y.TR Residuals
## HAM1     -0.0273  -0.0639   0.01096     -1.51
## HAM2     -0.0356  -0.0421   0.00351     -1.44
## HAM3     -0.0399  -0.0641   0.00492     -1.23
## HAM4     -0.0322  -0.0553   0.00378     -1.60
## HAM5     -0.0280  -0.0393  -0.00307     -1.75
## HAM6     -0.0391  -0.0498   0.00444     -1.27
```

```
# get the percentage component contributions to ES
decomp2$pcES

##        EDHEC.LS.EQ SP500.TR US.10Y.TR Residuals
## HAM1           0.0    43.37     4.650      52.0
## HAM2          66.2   -10.08     0.000      43.9
## HAM3          59.2     9.11     0.000      31.7
## HAM4          32.1    12.77     0.000      55.1
## HAM5          33.3     0.00     0.956      65.7
## HAM6          76.2    -9.49     0.000      33.3

# plot the percentage component contributions to ES
plot(fit.sub, which=10, f.sub=1:3)
```

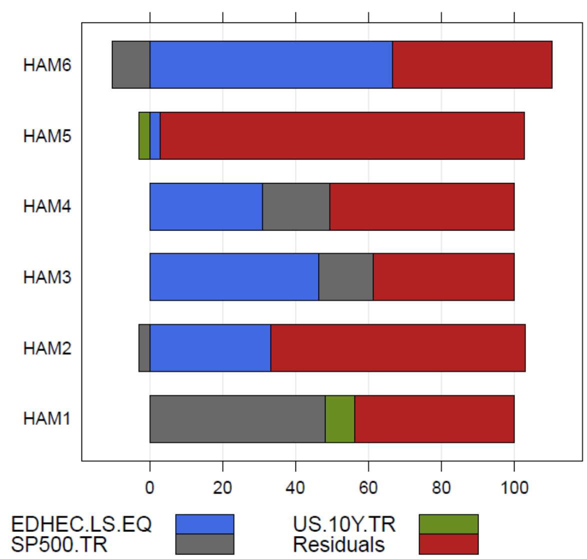

Figure 3.9: Percentage contribution to ES

3.4 Plot

Some types of individual asset (Figure 3.3) and group plots (all other plots in this chapter) have already been demonstrated. Let's look at all available arguments for plotting a $tsfm$ object.

```
## S3 method for class "tsfm"
plot (x, which=NULL, f.sub=1:2, a.sub=1:6, plot.single=FALSE, asset.name,
      colorset=c("royalblue","dimgray","olivedrab","firebrick",
                 "goldenrod","mediumorchid","deepskyblue","chocolate",
                 "darkslategray"),
      legend.loc="topleft", las=1, lwd=2, maxlag=15, ...)
```

3.4.1 *Group Plots*

This is the default option for plotting. Simply running $plot\ (fit)$, where fit is any $tsfm$ object, will bring up the following menu for group plots.

```
plot(fit.sub)

# Make a plot selection (or 0 to exit):

#  1: Factor model coefficients: Alpha
#  2: Factor model coefficients: Betas
#  3: Actual and Fitted asset returns
#  4: R-squared
#  5: Residual Volatility
#  6: Scatterplot matrix of residuals, with histograms, density overlays,
#     correlations and significance stars
#  7: Factor Model Residual Correlation
#  8: Factor Model Return Correlation
#  9: Factor Contribution to SD
# 10: Factor Contribution to ES
# 11: Factor Contribution to VaR
# 12: Asset returns vs factor returns (single factor model)
#
# Selection:
```

Note: Only a subset of assets and factors selected by $a.sub$ and $f.sub$ are plotted. The first 2 factors and first 6 assets are shown by default. The last option for plotting asset returns vs. factor returns is only applicable for single factor models.

```
# Examples of group plots: looping disabled & no. of assets displayed = 4.
plot(fit.sub, which=3, a.sub=1:4, legend.loc=NULL, lwd=1)
```

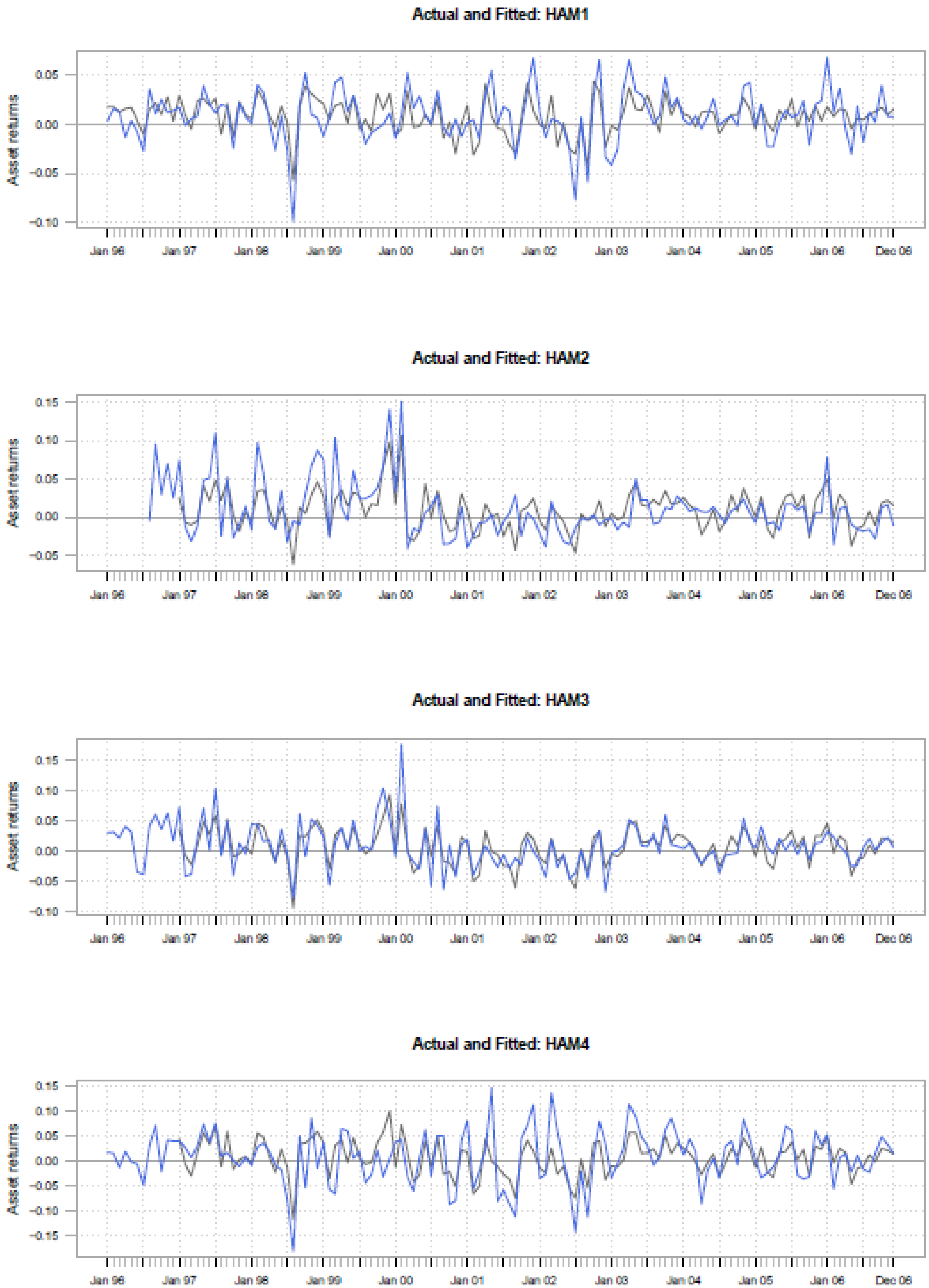

Figure 3.10: Actual and fitted returns for the 1st 4 assets

```
plot(fit.sub, which=6) # residual scatter plot matrix with correlations
```

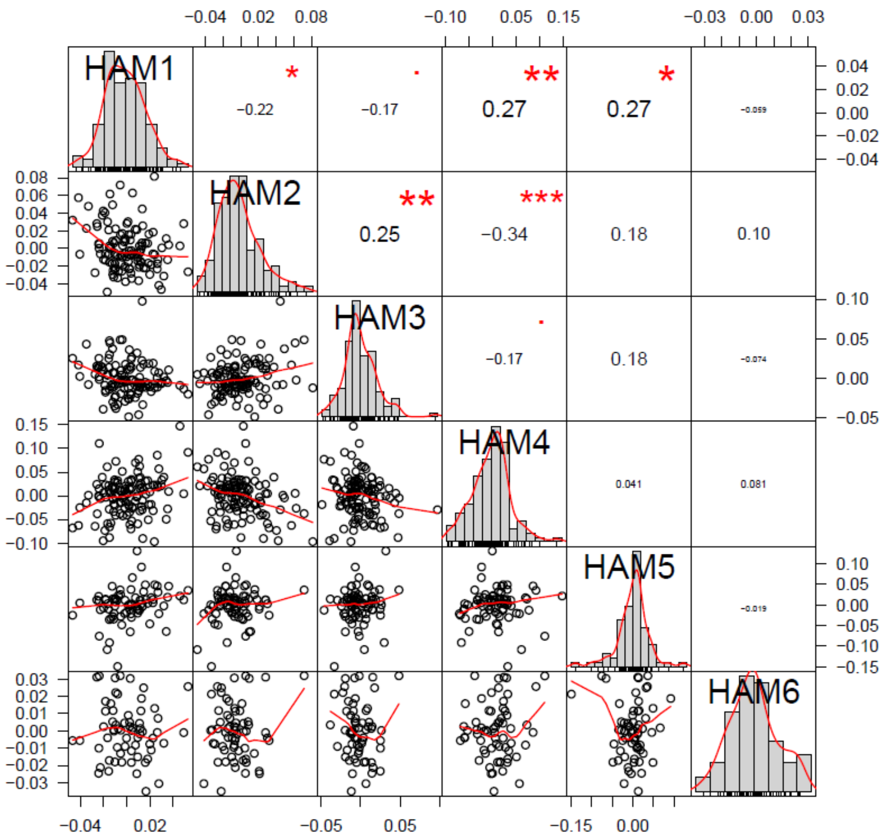

Figure 3.11: Residual scatterplot matrix with histograms, density overlays, correlations and significance stars

3.4.2 *Menu and Looping*

If the plot type argument *which* is not specified, a menu prompts for user input. In case multiple plots are needed, the menu is repeated after each plot (enter 0 to exit). User can also input a numeric vector of plot options via argument *which*.

3.4.3 *Individual Plots*

Setting $plot.single = TRUE$ enables individual asset plots. If there is more than one asset fit by the fitted object x, $asset.name$ is also necessary. In case the $tsfm$ object x contains only a single asset's fit, $plot.tsfm$ can infer $asset.name$ without user input. Here's the individual plot menu.

```
plot(fit.sub, plot.single=TRUE, asset.name="HAM1")

# Make a plot selection (or 0 to exit):
#  1: Actual and fitted asset returns
#  2: Actual vs fitted asset returns
#  3: Residuals vs fitted asset returns
#  4: Sqrt. of modified residuals vs fitted
#  5: Residuals with standard error bands
#  6: Time series of squared residuals
#  7: Time series of absolute residuals
#  8: SACF and PACF of residuals
#  9: SACF and PACF of squared residuals
# 10: SACF and PACF of absolute residuals
# 11: Non-parametric density of residuals with normal overlaid
# 12: Non-parametric density of residuals with skew-t overlaid
# 13: Histogram of residuals with non-parametric density and normal overlaid
# 14: QQ-plot of residuals
# 15: CUSUM test-Recursive residuals
# 16: CUSUM test-LS residuals
# 17: Recursive estimates (RE) test of LS regression coefficients
# 18: Rolling regression over a 24-period observation window
# 19: Asset returns vs factor returns (single factor model)
#
# Selection:
```

Note: CUSUM plots (options 15, 16 and 17) are applicable only for $fit.method = "LS"$. Modified residuals, rolling regression and single factor model plots (options 4, 18 and 19) are not applicable for $variable.selection = "lars"$.

Here are a few more examples which don't need interactive user input.

```
plot(fit.sub, plot.single=TRUE, asset.name="HAM1", which=5, ylim=c(-0.06,0.06))
```

```
plot(fit.sub, plot.single=TRUE, asset.name="HAM1", which=10)
```

```
plot(fit.sub, plot.single=TRUE, asset.name="HAM1", which=14)
grid()
```

```
plot(fit.sub, plot.single=TRUE, asset.name="HAM1", which=11)
```

```
plot(fit.sub, plot.single=TRUE, asset.name="HAM1", which=12)
```

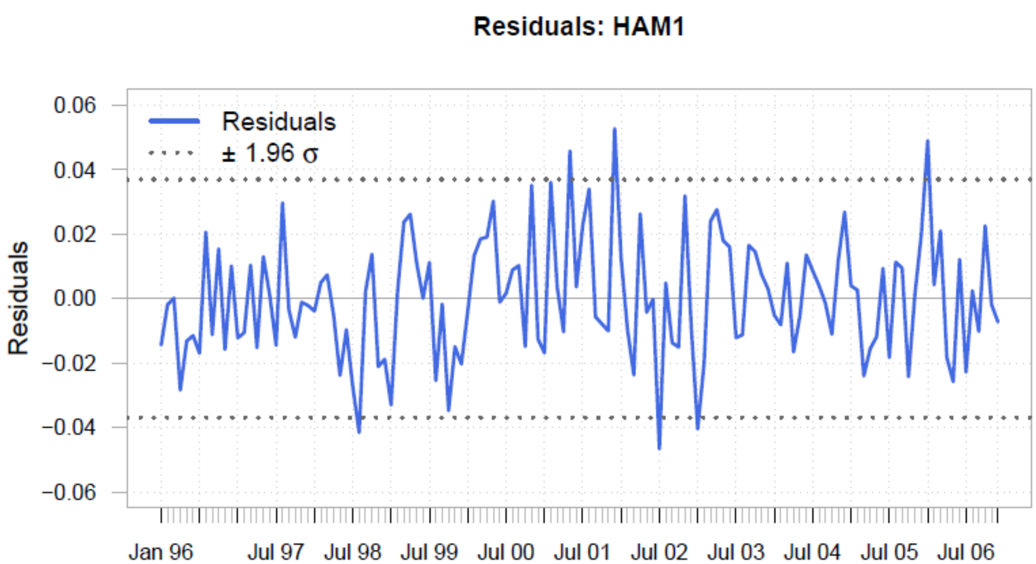

Figure 3.12: Time series plot of residuals with standard error bands: HAM1

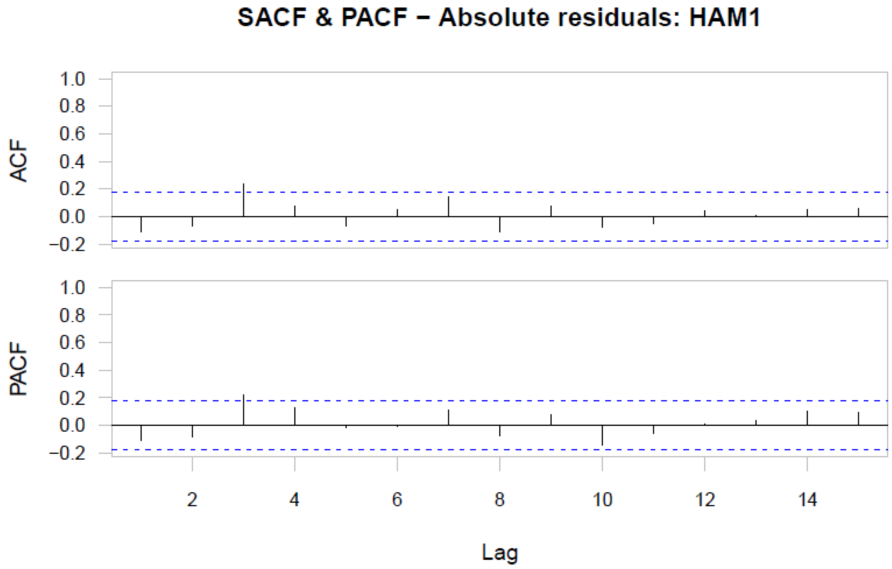

Figure 3.13: SACF and PACF of absolute residuals: HAM1

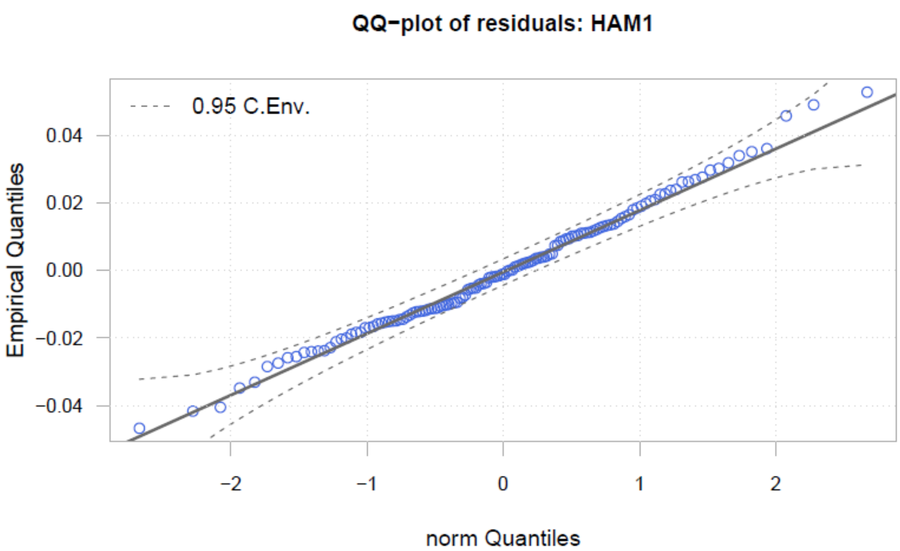

Figure 3.14: QQ-plot of residuals: HAM1

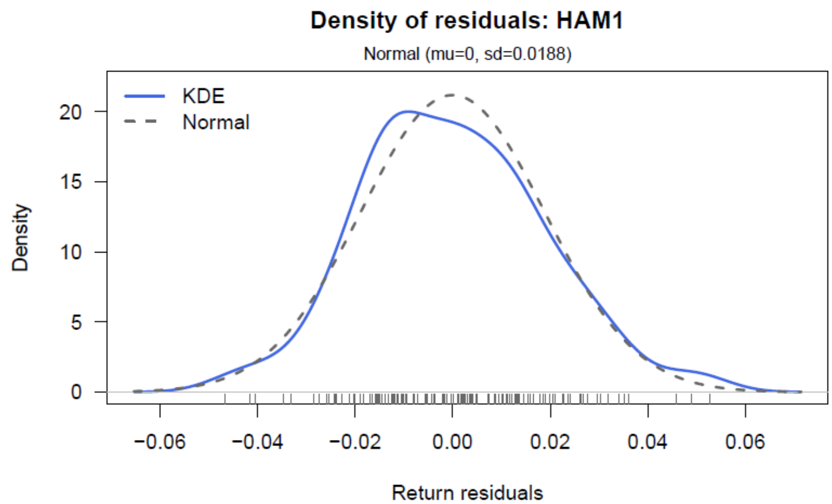

Figure 3.15: Non-parametric density of residuals with normal overlaid for HAM1

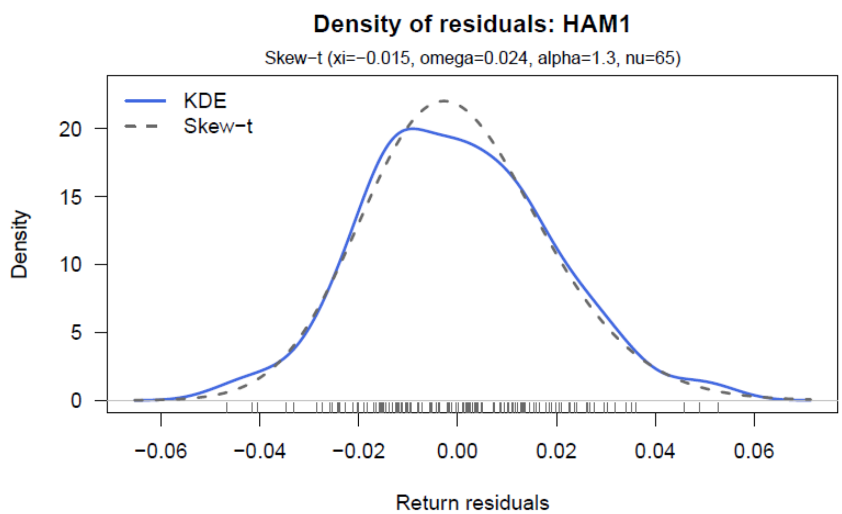

Figure 3.16: Non-parametric density of residuals with skew-t overlaid for HAM1

Chapter 4. FITTING STATISTICAL FACTOR MODELS

The purpose of this vignette is to demonstrate the use of the $fitSfm$ function and related control, analysis and plot functions in the $factorAnalytics$ package.

4.1 Overview

4.1.1 *Load Package*

The latest version of the $factorAnalytics$ package used in this vignette is hosted in the publicly available GitHub repository https://github.com/sangeeuw/factorAnalytics. There are plans for further updates to the package before its moved back to R-Forge and released on CRAN later this year. The package can be installed from GitHub as shown below.

```
library(devtools)
install_github("sangeeuw/factorAnalytics")
```

```
# load the package and its dependencies
library(factorAnalytics)
options(digits=3)
```

4.1.2 *Summary of Related Functions*

Here's a list of the functions and methods demonstrated in this vignette:

- $fitSfm\ (asset.names,\ factor.names,\ data,\ fit.method,\ variable.selection,\ \ldots)$: Fits a statistical factor model for one or more asset returns using Principal Component Analysis (PCA). When the number of assets exceeds the number of time periods, Asymptotic Principal Component Analysis (APCA) is performed. Additionally, for APCA, user can specify a method, one of Connor & Korajczyk (1993) or Bai & Ng (2002), to determine the number of factors and/or choose to use the Connor and Korajczyk (1988) refinement to the APCA procedure. The returned object is of class sfm and contains the fitted lm object, estimated factor realizations, factor loadings, R^2, residual volatility, factor model return covariance and the factor mimicking portfolio weights.

- $coef\ (object,\ \ldots)$: Extracts the coefficient matrix (intercept and factor betas) for all assets fit by the sfm object.
- $fitted\,(object,\ \ldots)$: Returns an xts data object of fitted asset returns from the factor model for all assets.
- $residuals\,(object,\ \ldots)$: Returns an xts data object of residuals from the fitted factor model for all assets.
- $fmCov\,(object,\ use,\ \ldots)$: Returns the $N \times N$ symmetric covariance matrix for asset returns based on the fitted factor model. use specifies how missing values are to be handled.
- $fmSdDecomp\,(object,\ factor.cov,\ use,\ \ldots)$: Returns a list containing the standard deviation of asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. use specifies how missing values are to be handled.
- $fmVaRDecomp\,(object,\ factor.cov,\ p,\ type,\ use,\ \ldots)$: Returns a list containing the value-at-risk (VaR) for asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical

factor returns. $type$ specifies if VaR computation should be non-parametric (sample quantile) or based on a Normal distribution. And, p specifies the confidence level.

- $fmEsDecomp\ (object,\ factor.cov,\ p,\ type,\ use,\ \ldots)$: Returns a list containing the expected shortfall (ES) for asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. $type$ specifies if ES computation should be non-parametric (sample quantile) or based on a Normal distribution. And, p specifies the confidence level.
- $plot\ (x,\ \ldots)$: The plot method for class sfm can be used for plotting factor model characteristics of a group of assets (default) or an individual asset. The user can select the type of plot either from the menu prompt or directly via argument $which$. In case multiple plots are needed, the menu is repeated after each plot (enter 0 to exit). User can also input a numeric vector of plot options via $which$.
- $predict\ (object,\ newdata,\ \ldots)$: The predict method for class sfm returns a vector or matrix of predicted values for a new data sample or simulated values.
- $summary\ (object,\ se.type,\ \ldots)$: The summary method for class sfm returns an object of class $summary.sfm$ containing the summaries of the fitted lm objects, factor loadings, residual volatilities (under the homo-skedasticity assumption), R^2 values and factor-mimicking portfolio weights. Printing the factor model summary object displays the call, coefficients (with standard errors and t-statistics), R^2 and residual volatility for all assets and top long and short positions for each factor-mimicking portfolio.

4.1.3 *Data*

The examples in this chapter primarily use the $StockReturns$ dataset from the $factorAnalytics$ package. It contains two $data.frame$ objects $r.M$ and $r.W$. Originally used in Berndt (1991), $r.M$ has 120 observations of 15 variables (U.S. stock returns) and the frequency is monthly. Whereas, $r.W$ has 182 weekly observations of 1618 variables (U.S. stock returns).

```
# load the Rdata object
data(StockReturns)
# view class and dimensions
class(r.M)

## [1] "data.frame"

dim(r.M)

## [1] 120  15

# variable names
colnames(r.M)

##  [1] "CITCRP" "CONED"  "CONTIL" "DATGEN" "DEC"    "DELTA"  "GENMIL"
##  [8] "GERBER" "IBM"    "MOBIL"  "PANAM"  "PSNH"   "TANDY"  "TEXACO"
## [15] "WEYER"

# range of observations
range(rownames(r.M))

## [1] "1978-01-01" "1987-12-01"
```

```
class(r.W)

## [1] "data.frame"

dim(r.W)

## [1]  182 1618

range(rownames(r.W))

## [1] "1997-01-08" "2000-06-28"
```

The yield curve example in section 4.3, uses the $TreasuryYields$ dataset in the $factorAnalytics$ package. This contains an xts data object $tr.yields$. The data was obtained from the companion website to Ruppert (2010).

```
data(TreasuryYields)
head(tr.yields)

##            X1mo X3mo X6mo X1yr X2yr X3yr X5yr X7yr X10yr X20yr X30yr
## 1990-01-02   NA 7.83 7.89 7.81 7.87 7.90 7.87 7.98  7.94    NA  8.00
## 1990-01-03   NA 7.89 7.94 7.85 7.94 7.96 7.92 8.04  7.99    NA  8.04
## 1990-01-04   NA 7.84 7.90 7.82 7.92 7.93 7.91 8.02  7.98    NA  8.04
## 1990-01-05   NA 7.79 7.85 7.79 7.90 7.94 7.92 8.03  7.99    NA  8.06
## 1990-01-08   NA 7.79 7.88 7.81 7.90 7.95 7.92 8.05  8.02    NA  8.09
## 1990-01-09   NA 7.80 7.82 7.78 7.91 7.94 7.92 8.05  8.02    NA  8.10

range(index(tr.yields))

## [1] "1990-01-02" "2008-10-31"
```

$tr.yields$, contains U.S. Treasury bond yields for 11 different maturities (1, 3, and 6 months and 1, 2, 3, 5, 7, 10, 20, and 30 years). Daily yields were taken from a U.S. Treasury website for the period January 2, 1990, to October 31, 2008. Daily yields are missing from some maturities. For example, the 20-year constant maturity series were discontinued at the end of calendar year 1986 and reinstated on October 1, 1993. Omitting the missing values of the differenced data, leaves 819 days of observations. Excluding the one-month and 20-year maturities would leave us with a longer series.

4.2 Fitting a Statistical Factor Model

In statistical factor models, factor realizations are not directly observable (unlike times series factor models) and the factor loadings are not known (unlike fundamental factor models). Both factor returns and exposures must be extracted from the asset returns data using statistical methods such as factor analysis or Principal Component Analysis (PCA). Chapter 15 from Zivot & Jia-hui (2006) serves as a good reference for a description of the different multi-factor models, estimation methods and relevant examples using S-PLUS.

PCA uses the eigen decomposition of the covariance (or correlation) matrix of asset returns to find the first K principal components that explain the largest portion of the sample covariance matrix of returns. Factor loadings are then estimated using time series regression. Factor analysis involves maximum likelihood optimization to estimate the factor loadings and the residual covariance matrix, constructing the factor realizations and choosing a rotation of the coordinate system for a more meaningful interpretation of the factors.

In $fitSfm$, PCA is applied to extract the factor realizations when the number of time series observations, T, is greater than the number of assets, N. When $N > T$, the sample covariance matrix for asset returns is singular and Asymptotic Principal Component Analysis (APCA) based on Connor & Korajczyk (1988) is performed.

Let's look at the arguments for $fitSfm$.

```
args(fitSfm)

## function (data, k = 1, max.k = NULL, refine = TRUE, sig = 0.05,
##     check = FALSE, corr = FALSE, ...)
## NULL
```

A time series of asset returns is input via argument $data$. If $data$ is not of class xts, it's row names must provide an xts compatible time index. Specifying $check = TRUE$, issues a warning if any asset is found to have identical observations. And before model fitting, incomplete cases in $data$ are removed.

For both PCA and APCA, any number of factors less than $\min(N, T)$ can be chosen explicitly via argument k. Alternately for APCA, a method to determine the number of factors can be specified: $k = "bn"$ corresponds to Bai & Ng (2002) and $k = "ck"$ corresponds to Connor & Korajczyk (1993). User can specify the maximum number of factors, $max.k$ to consider with these methods. If not, it is assumed to be either 10 or $T - 1$, whichever is smaller.

For the "ck" method, sig specifies the desired level of significance. Argument $refine$ specifies whether a refinement of the APCA procedure from Connor and Korajczyk (1988) that may improve efficiency is to be used.

When $corr = TRUE$, the correlation matrix of returns is used for finding the principal components instead of the covariance matrix. This is typically decided by practitioners on a case-by-case basis. The variable with the highest variance dominates the PCA when the covariance matrix is used. However, this may be justified if a volatile asset is more interesting for some reason and volatility information shouldn't be discarded. On the other hand, using the correlation matrix standardizes the variables and makes them comparable, avoiding penalizing variables with less dispersion.

Finally, if the median of the 1st principal component is negative, all it's factor realizations are automatically inverted to enable more meaningful interpretation.

4.2.1 *Principal Components Analysis*

The following example fits a statistical factor model with two principal components for $r.M$, the monthly returns on fifteen U.S. stocks described in section 4.1.3. Since the number of observations is larger than the number of assets in this case, $fitSfm$ will choose to perform PCA.

```
fit.pca <- fitSfm(r.M, k=2)
```

The resulting object, $fit.pca$, has the following attributes.

```
class(fit.pca)

## [1] "sfm"

names(fit.pca)

##  [1] "asset.fit"   "k"           "factors"     "loadings"    "alpha"
##  [6] "r2"          "resid.sd"    "residuals"   "Omega"       "eigen"
## [11] "mimic"       "call"        "data"        "asset.names"
```

The component k contains the number of factors, either as input or determined by "ck" or "bn" methods. The N (or, T for APCA) eigenvalues of the sample covariance matrix are in $eigen$. The $T \times K$ xts object of estimated factor realizations is in $factors$.

The component $asset.fit$ contains an object of class "mlm" or "lm" from the time-series OLS regression of asset returns on estimated factors. The estimated factor loadings[19] are in $loadings$ and regression alphas are in $alpha$. The $T \times N$ xts object of residuals from the OLS regression are in $residuals$. R^2 and residual standard deviations are in $r2$ and $resid.sd$ respectively.

The $N \times N$ return covariance matrix estimated by the factor model is in $Omega$[20]. The $N \times K$ matrix of factor mimicking portfolio weights are given in $mimic$[21]. The remaining components contain the input choices and the data. The fitted factor model is printed below.

[19] Refer to the summary method in section 4.2.3 for standard errors, degrees of freedom, t-statistics etc.

[20] Section 4.4.1 on Factor Model Covariance gives a detailed derivation.

[21] The summary method in section 4.2.3 helps to make this more tangible by summarizing the largest and smallest weights for each factor mimicking portfolio.

```
fit.pca # print the fitted "sfm" object

##
## Call:
## fitSfm(data = r.M, k = 2)
##
## Model dimensions:
## Factors  Assets Periods
##       2      15     120
##
## Factor Loadings:
##       F.1              F.2
##  Min.   :0.044   Min.   :-0.824
##  1st Qu.:0.139   1st Qu.:-0.067
##  Median :0.250   Median : 0.012
##  Mean   :0.231   Mean   :-0.002
##  3rd Qu.:0.308   3rd Qu.: 0.142
##  Max.   :0.417   Max.   : 0.365
##
## R-squared values:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.031   0.214   0.427   0.398   0.573   0.925
##
## Residual Volatilities:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0415  0.0538  0.0721  0.0693  0.0779  0.1090
```

The screeplot of eigenvalues is illustrated in Figure 4.1 (option 1 on the plot menu; refer to section 4.5 for a list of all the plot options). The first principal component explains about 35% of the total variance, and the first two components explain about 50% of the total variance. By specifying $eig.max = 0.9$, we are requesting the first set of components that explain at least 90% of the total variance.

```
plot(fit.pca, which=1, eig.max=0.9)
```

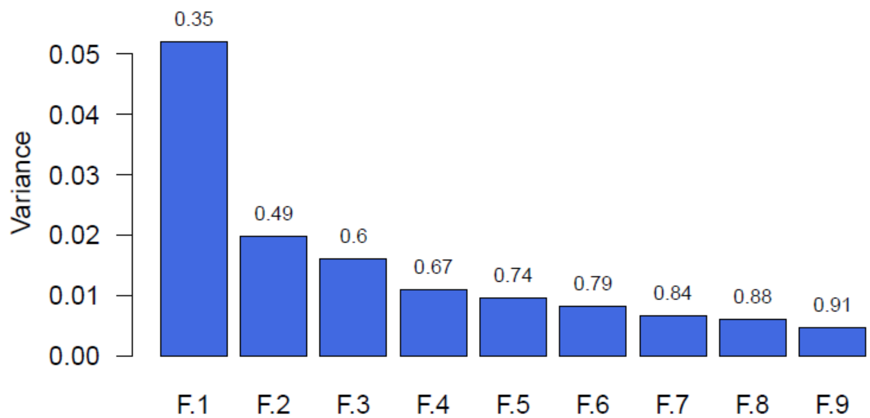

Figure 4.1: Screeplot of eigen values: $fit.pca$

The time series of estimated factor returns are displayed in Figure 4.2.

```
plot(fit.pca, which=2)
```

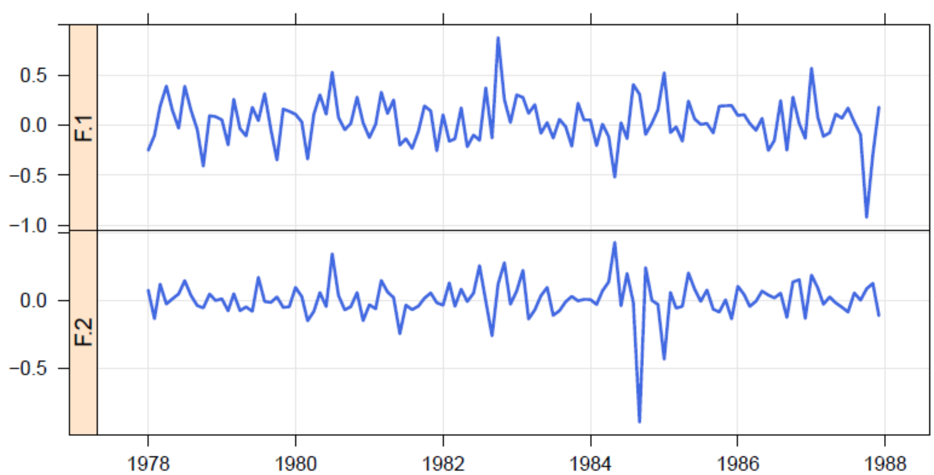

Figure 4.2: Time series of estimated factors: $fit.pca$

The estimated factor loadings for all assets are shown in Figure 4.3. Note that the first factor has all positive loadings. The second factor has both positive and negative loadings.

```
plot(fit.pca, which=3, a.sub=1:15)
```

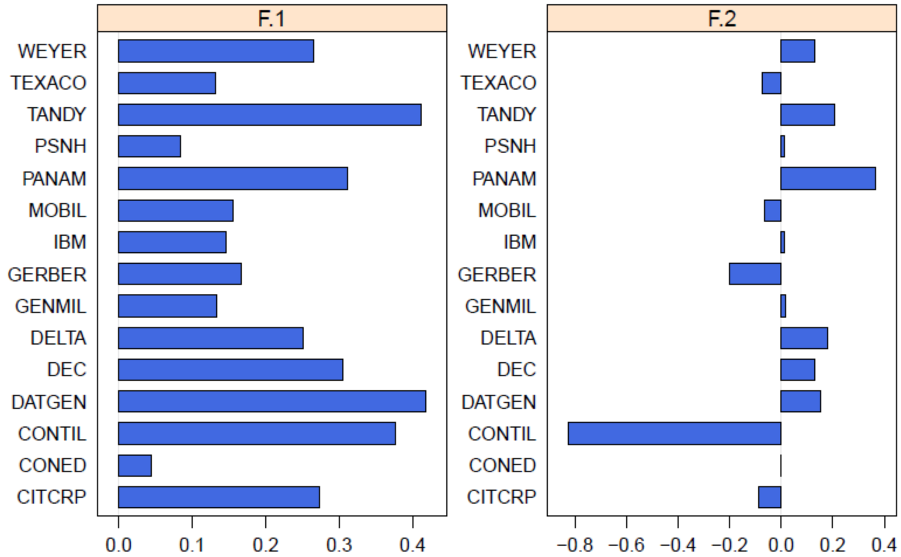

Figure 4.3: Estimated factor loadings: $fit.pca$

Figure 4.4 displays the top three ($n.top$) assets with the largest and smallest weights in each factor mimicking portfolio. For the first factor, assets DATGEN, TANDY and CONTIL have the highest weights and assets CONED, PSNH and TEXACO have the lowest weights. Since all the weights in the first portfolio are positive, this might be construed as a market-wide factor.

```
# Factor mimicking portfolio weights from PCA fit
t(fit.pca$mimic)

##      CITCRP   CONED  CONTIL DATGEN     DEC   DELTA  GENMIL GERBER     IBM
## F.1 0.0786  0.0128  0.109   0.12  0.0878  0.0721  0.0382 0.0482  0.0422
## F.2 2.3217 -0.0324 22.365  -4.12 -3.5049 -4.8626 -0.4649 5.3906 -0.3367
##       MOBIL   PANAM    PSNH  TANDY TEXACO   WEYER
## F.1 0.0447  0.0895  0.0242  0.119 0.0381  0.0763
## F.2 1.6931 -9.9234 -0.2840 -5.624 1.9488 -3.5637

plot(fit.pca, which=12, n.top=3)
```

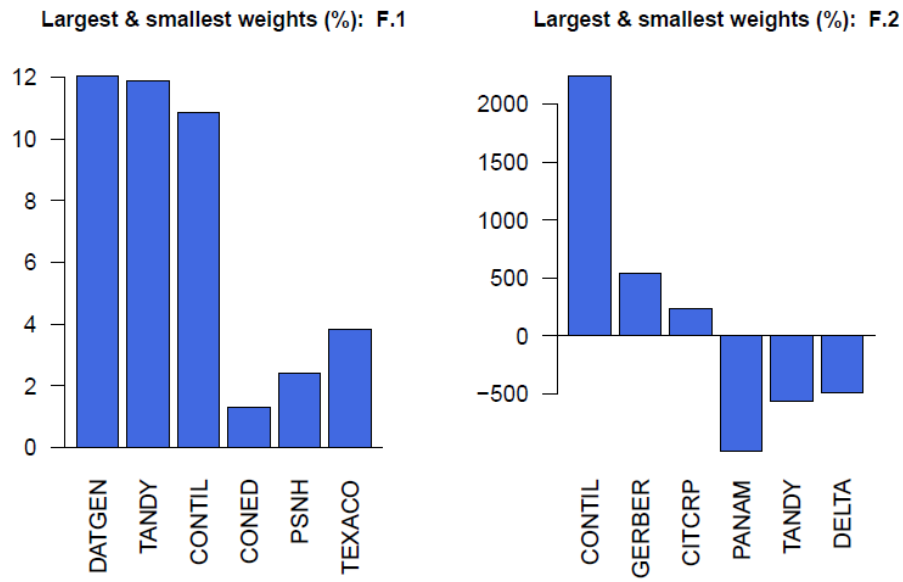

Figure 4.4: Top 3 largest and smallest weights in the factor mimicking portfolios

Figure 4.5 gives the correlations between assets with $n.top$ largest and smallest weights in the factor mimicking portfolio for the first principal component.

```
plot(fit.pca, which=13, f.sub=1, n.top=3)
```

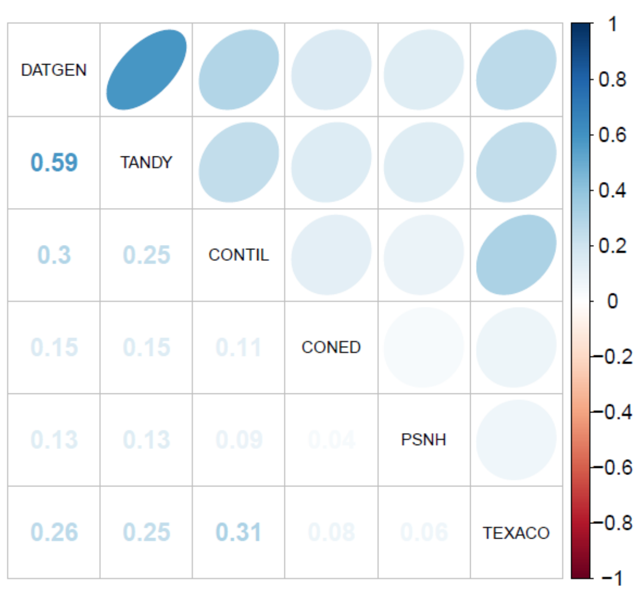

Figure 4.5: Correlation between assets with the top 3 largest and smallest positions in factor $F.1$'s factor mimicking portfolio

4.2.2 *Asymptotic Principal Components Analysis*

The following example fits a statistical factor model with two principal components for $r.W$, the weekly returns on 1618 U.S. stocks described earlier. Since the number of observations is smaller than the number of assets in this case, $fitSfm$ would choose to perform APCA. The primary difference is that the $T \times T$ covariance matrix is used instead.

```
fit.apca <- fitSfm(r.W, k=15)
```

Since the optional argument $refine = TRUE$ by default, the APCA refinement will be used. This procedure involves rescaling the returns using the residual variances obtained from one iteration of the APCA procedure, re-computing the $T \times T$ covariance matrix and performing a second iteration of the APCA procedure using this covariance matrix. This refinement may improve efficiency. Figure 4.6 and Figure 4.7 give the screeplot of eigenvalues and the estimated time series of the first 4 factor realizations respectively.

```
plot(fit.apca, which=1, eig.max=0.4, las=2)
```

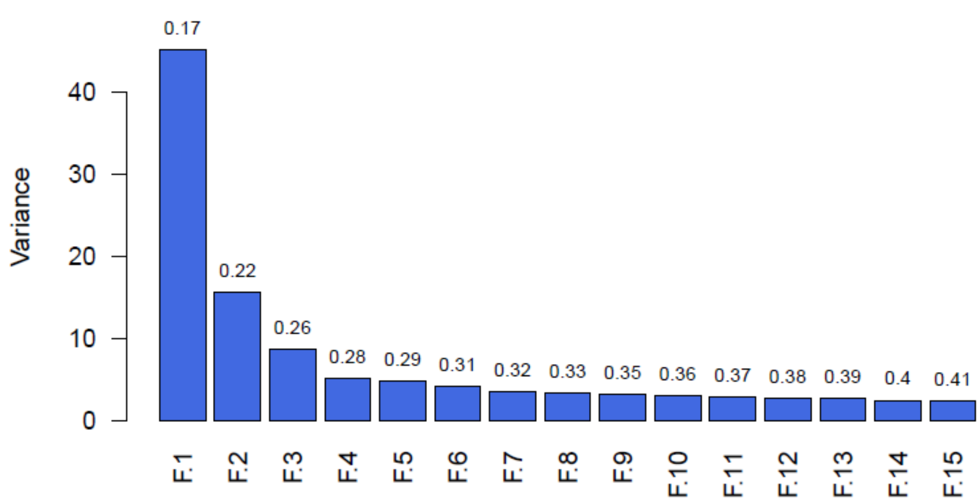

Figure 4.6: Screeplot of eigen values: $fit.apca$

```
plot(fit.apca, f.sub=1:4, which=2)
```

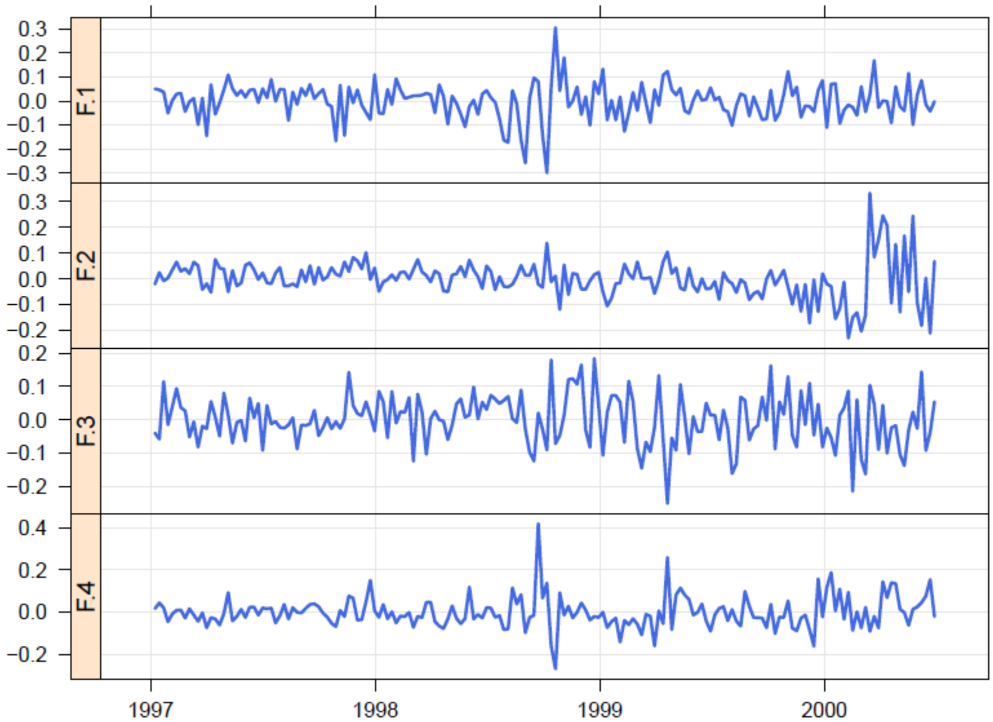

Figure 4.7: Time series of first 4 factor returns: $fit.apca$

Note that the number of factors was known or pre-specified in $fit.apca$ above. In practice, the number of factors is unknown and must be determined from the data. Two such procedures are available via $fitSfm$ via the argument k: "bn" corresponds to Bai & Ng (2002) and "ck" corresponds to Connor & Korajczyk (1993). The maximum number of factors to be considered with these methods is specified via $max.k$. By default, it is assumed to be either 10 or $T-1$, whichever is smaller. For the "ck" method, sig specifies the desired level of significance.

Here are some examples using the "ck" or "bn" method for performing APCA with the weekly return data for 1618 U.S. stocks. We find that both these methods select 2 factors and hence output the same factor model in this case.

```
# APCA with the Bai & Ng method
fit.apca.bn <- fitSfm(r.W, k="bn")
summary(fit.apca.bn$loadings)

##       F.1                F.2
##  Min.   :-0.177   Min.   :-1.119
##  1st Qu.: 0.218   1st Qu.:-0.181
##  Median : 0.317   Median : 0.005
##  Mean   : 0.332   Mean   :-0.078
##  3rd Qu.: 0.419   3rd Qu.: 0.096
##  Max.   : 0.950   Max.   : 0.411

# APCA with the Connor-Korajczyk method
fit.apca.ck <- fitSfm(r.W, k="ck", sig=0.05)
fit.apca.ck$k

## [1] 2
```

Finally, since the number of assets is large, it helps to look at the histograms of R^2 values and residual volatilities for all assets as shown in Figure 4.8 and Figure 4.9 respectively.

```
plot(fit.apca, which=4, legend.loc="topright")
```

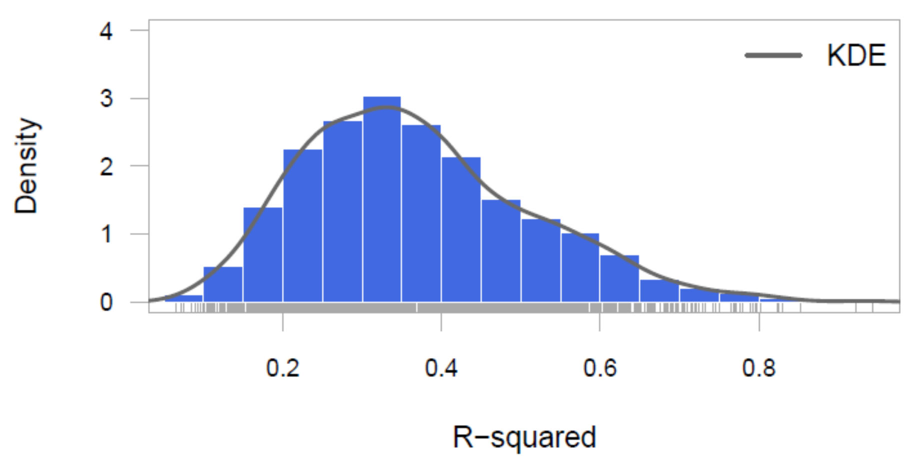

Figure 4.8: Histogram of R^2 values: $fit.apca$

```
plot(fit.apca, which=5, legend.loc="topright")
```

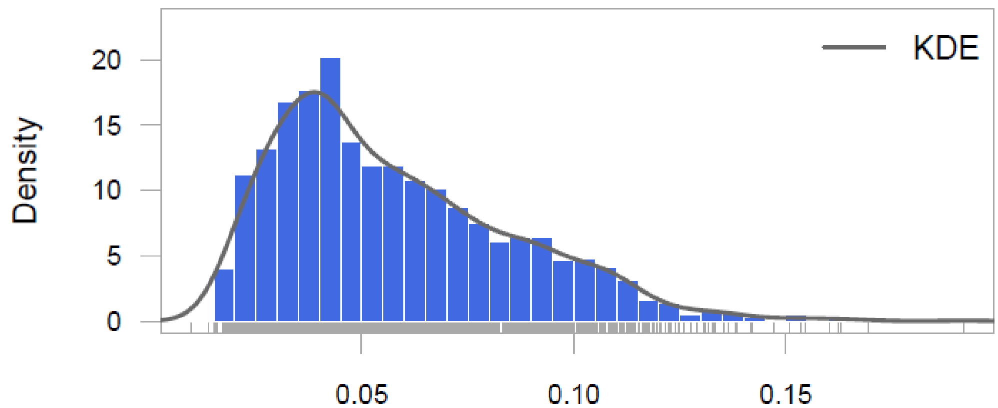

Figure 4.9: Histogram of residual volatilities: $fit.apca$

4.2.3 *S3 Generic Methods*

Many useful generic accessor functions are available for sfm fit objects.

```
methods(class="sfm")

##  [1] coef        fitted      fmCov       fmEsDecomp  fmSdDecomp
##  [6] fmVaRDecomp plot        predict     print       residuals
## [11] summary
## see '?methods' for accessing help and source code
```

$coef()$ returns a matrix of estimated model coefficients including the intercept. $fitted()$ returns an xts data object of the part of observed asset returns explained by the factor model. $residuals()$ returns an xts data object with the part of observed asset returns not explained by the factor model. $predict()$ uses the fitted factor model to estimate asset returns given a set of new or simulated factor return data.

```
## S3 method for class "sfm"
summary.sfm (object, se.type="Default", n.top=3, ...)
```

$summary()$ prints the call, coefficients (with standard errors and t-statistics), R^2 and residual volatilities for all assets, and $n.top$ long and short positions for each factor-mimicking portfolio. Argument $se.type$, one of $Default$, HC or HAC, allows for heteroskedasticity and auto-correlation consistent estimates and standard errors. A $summary.tsfm$ object is returned which contains a list of summaries of the fitted lm objects, factor loadings, residual volatilities (under homo-skedasticity assumption), R^2 values and factor-mimicking portfolio weights.

Factor model covariance and risk decomposition functions are explained in section 4.4 and the plot method is discussed separately in section 4.5. Here are some examples using the statistical factor models fitted earlier.

```
# all estimated coefficients from PCA example
coef(fit.pca)

##         (Intercept)    F.1      F.2
## CITCRP     0.001761 0.2727 -0.08549
## CONED      0.016733 0.0444  0.00119
## CONTIL    -0.008949 0.3769 -0.82358
## DATGEN    -0.010411 0.4172  0.15182
## DEC        0.006514 0.3049  0.12907
## DELTA      0.000199 0.2502  0.17906
## GENMIL     0.011167 0.1326  0.01712
## GERBER     0.011475 0.1672 -0.19851
## IBM        0.003690 0.1464  0.01240
## MOBIL      0.010565 0.1552 -0.06235
## PANAM     -0.011994 0.3107  0.36542
## PSNH      -0.007648 0.0841  0.01046
## TANDY      0.006845 0.4119  0.20710
## TEXACO     0.007307 0.1323 -0.07177
## WEYER     -0.002030 0.2649  0.13123
```

```
# compare returns data with fitted and residual values for CITCRP: fit.pca
CITCRP.ts <- merge(fit.pca$data[,1], fitted(fit.pca)[,1],
                   residuals(fit.pca)[,1])
colnames(CITCRP.ts) <- c("CITCRP.return","CITCRP.fitted","CITCRP.residual")
tail(CITCRP.ts)
```

```
##            CITCRP.return CITCRP.fitted CITCRP.residual
## 1987-07-01         0.041       0.05562        -0.01462
## 1987-08-01         0.033       0.00565         0.02735
## 1987-09-01        -0.086      -0.02429        -0.06171
## 1987-10-01        -0.282      -0.25650        -0.02550
## 1987-11-01        -0.136      -0.08875        -0.04725
## 1987-12-01         0.064       0.06053         0.00347
```

```
# summary for fit.pca with HAC standard erros
sum.pca <- summary(fit.pca, se.type="HAC", n.top=3)
names(sum.pca)
```

```
## [1] "call"      "se.type"   "sum.list"  "mimic.sum"
```

```
# print the summary for the 1st asset
sum.pca$sum.list[[1]]
```

```
##
## Call:
## lm(formula = CITCRP ~ f)
##
## Residuals:
##          CITCRP
## Min    -0.13961
## 1Q     -0.03487
## Median  0.00822
## 3Q      0.03419
## Max     0.14293
```

```
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.00176    0.00461    0.38    0.703
## object$factorsF.1  0.27271    0.02206   12.36   <2e-16 ***
## object$factorsF.2 -0.08549    0.04724   -1.81    0.073 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0505 on 117 degrees of freedom
## Multiple R-squared:  0.618,Adjusted R-squared:  0.611
## F-statistic: 94.6 on 2 and 117 DF,  p-value: <2e-16
```

```
# print the summary for the factor mimicking portfolio weights
sum.pca$mimic.sum
```

```
## $F.1
##   Top.Long.Name Top.Long.Weight Top.Short.Name Top.Short.Weight
## 1        DATGEN           0.120          CONED           0.0128
## 2         TANDY           0.119           PSNH           0.0242
## 3        CONTIL           0.109         TEXACO           0.0381
##
## $F.2
##   Top.Long.Name Top.Long.Weight Top.Short.Name Top.Short.Weight
## 1        CONTIL           22.36          PANAM            -9.92
## 2        GERBER            5.39          TANDY            -5.62
## 3        CITCRP            2.32          DELTA            -4.86
```

4.3 TREASURY YIELD CURVE EXAMPLE

The following example uses PCA to model yield curve variations similar to Example 17.2 in Ruppert (2010). The Treasury yields data used was described in section 4.1 earlier. Figure 4.10 plots the time series of the raw data and Figure 4.11 shows the treasury yield curve through time.

```
plot.zoo(tr.yields, main="Treasury yields", col="royalblue")
```

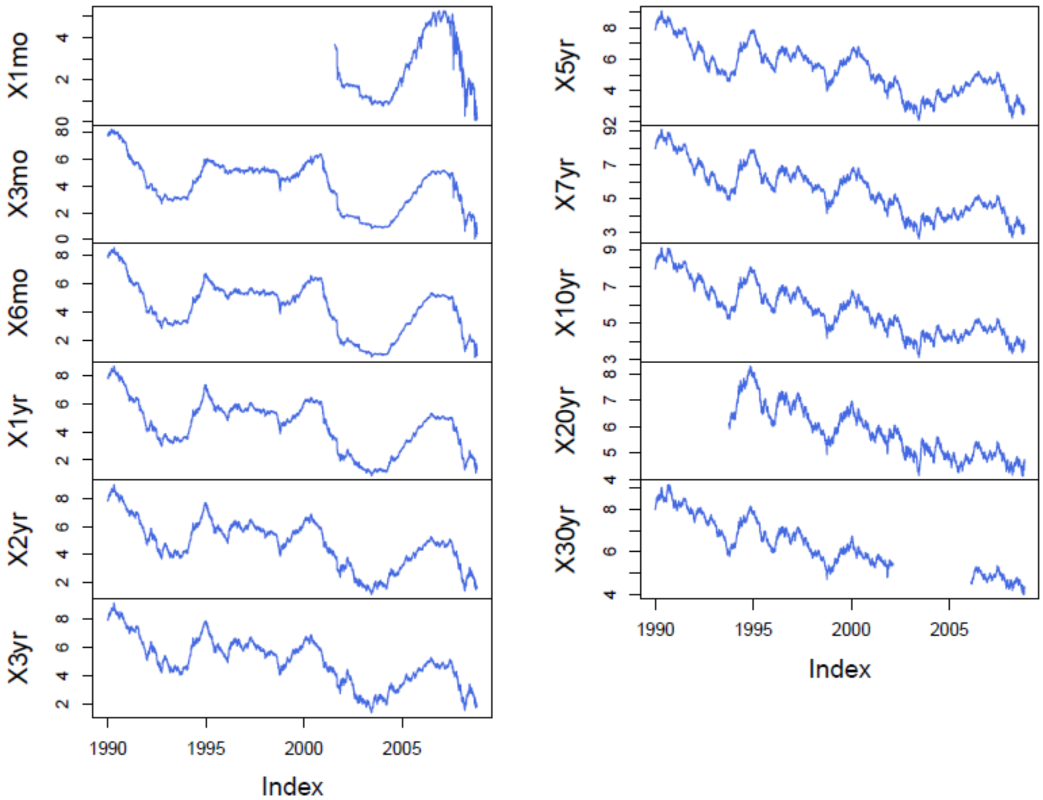

Figure 4.10: Time-series of U.S. Treasury yields

```
dat <- na.omit(tr.yields) # remove NAs
time = c(1/12,.25,.5,1, 2, 3, 5, 7, 10, 20, 30)
plot(time, as.vector(dat[1,]), ylim=c(0,6), type="b", col="royalblue", lwd=2,
     pch=19, ylab="Yield", xlab="T")
```

```
lines(time, as.vector(dat[486,]), type="b", lwd=2, col="olivedrab", pch=19)
lines(time, as.vector(dat[821,]), type="b", lwd=2, col="firebrick", pch=19)
legend("bottomright", c("07/31/01","07/02/07","10/31/08"),
       col=c("royalblue","olivedrab","firebrick"), lwd=2, bty="n")
```

Next, we fit a statistical factor model to the differenced data, with missing values removed. Since all 11 series have the same units and a comparable scale, PCA is performed on the sample correlation matrix.

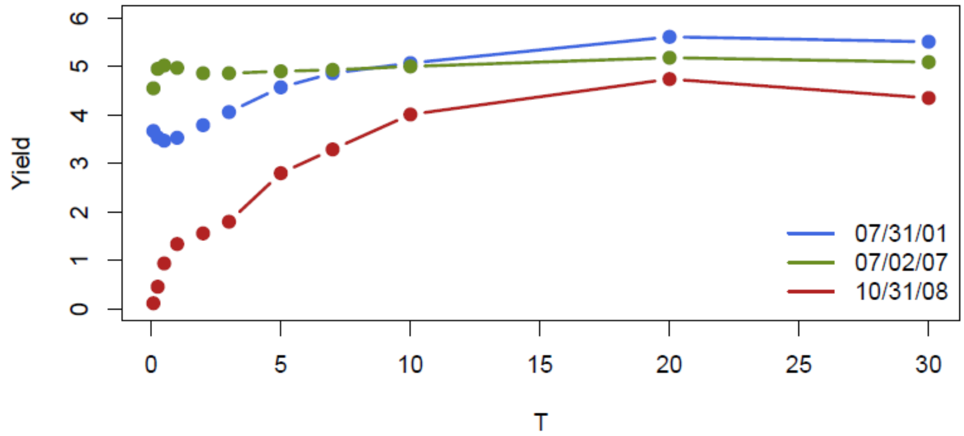

Figure 4.11: Treasury yield curve at 3 different dates

```
diff.yield <- na.omit(diff(tr.yields))
dim(diff.yield)

## [1] 819  11

yield.pca <- fitSfm(diff.yield, k=3, corr=TRUE)
```

Figure 4.12 shows a screeplot of all the eigenvalues. Approximately 94% of the variation is explained by the first 3 principal components and 99% is explained by the first five. So, the choice of $k = 3$ when fitting the model is not inappropriate.

```
plot(yield.pca, which=1, f.sub=1:3, eig.max=1)
```

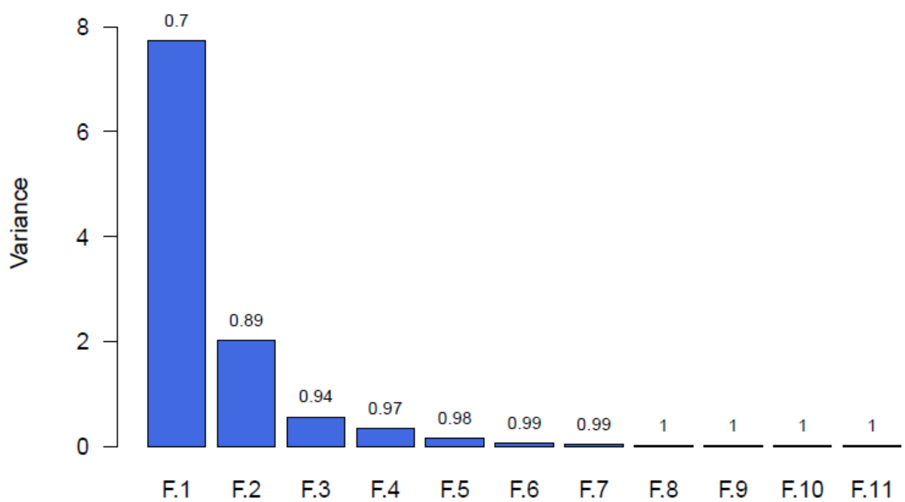

Figure 4.12: Screeplot of eigen values for the first difference of Treasury yields

A summary of the distribution of the factor loadings and the top three assets with the largest and smallest weights in each factor mimicking portfolio are shown below.

```
beta <- yield.pca$loadings
summary(beta)

##       F.1              F.2               F.3
##  Min.   :0.176   Min.   :-0.588   Min.   :-0.375
##  1st Qu.:0.231   1st Qu.:-0.328   1st Qu.:-0.173
##  Median :0.272   Median : 0.053   Median :-0.025
##  Mean   :0.294   Mean   :-0.052   Mean   : 0.031
##  3rd Qu.:0.373   3rd Qu.: 0.204   3rd Qu.: 0.162
##  Max.   :0.397   Max.   : 0.251   Max.   : 0.870

summary(yield.pca)$mimic.sum
## $F.1
##   Top.Long.Name Top.Long.Weight Top.Short.Name Top.Short.Weight
## 1          X5yr           0.107           X1mo           0.0488
## 2          X7yr           0.106           X3mo           0.0595
## 3          X3yr           0.106           X6mo           0.0781
##
## $F.2
##   Top.Long.Name Top.Long.Weight Top.Short.Name Top.Short.Weight
## 1          X3mo           0.942          X30yr           -0.487
## 2          X1mo           0.853          X20yr           -0.477
## 3          X6mo           0.741          X10yr           -0.386
##
## $F.3
##   Top.Long.Name Top.Long.Weight Top.Short.Name Top.Short.Weight
## 1          X1mo           1.754           X6mo           -0.805
## 2         X30yr           0.970           X1yr           -0.803
## 3         X20yr           0.801           X2yr           -0.738
```

Figure 4.13 and Figure 4.14 show the factor loadings as barplots and as line plots respectively.

```
plot(yield.pca, which=3, f.sub=1:3, a.sub=1:11)
```

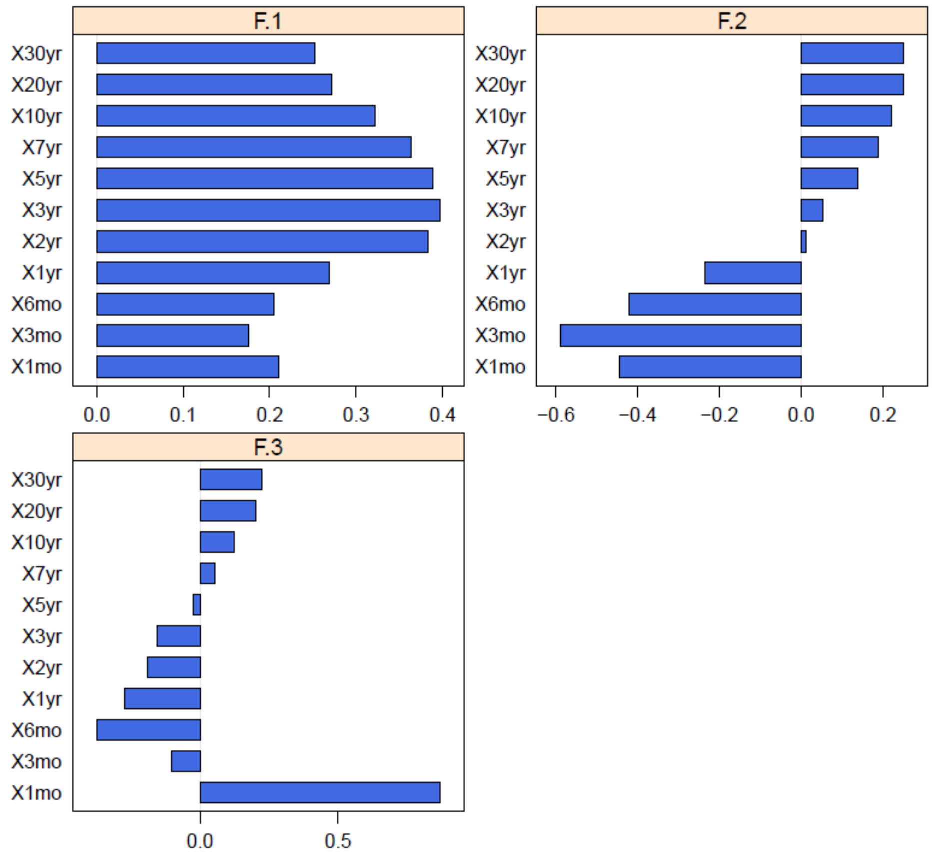

Figure 4.13: Factor loadings on the 3 statistical factors

```
plot(time, beta[,1], ylim=c(-.8,.8), type="b", col="royalblue", lwd=2, pch=19,
     ylab="Factor loading", xlab="T")
lines(time, beta[,2], type="b", lwd=2, col="olivedrab", pch=19)
lines(time, beta[,3], type="b", lwd=2, col="firebrick", pch=19)
legend("bottomright", c("F.1","F.2","F.3"),
       col=c("royalblue","olivedrab","firebrick"), lwd=2, bty="n")
```

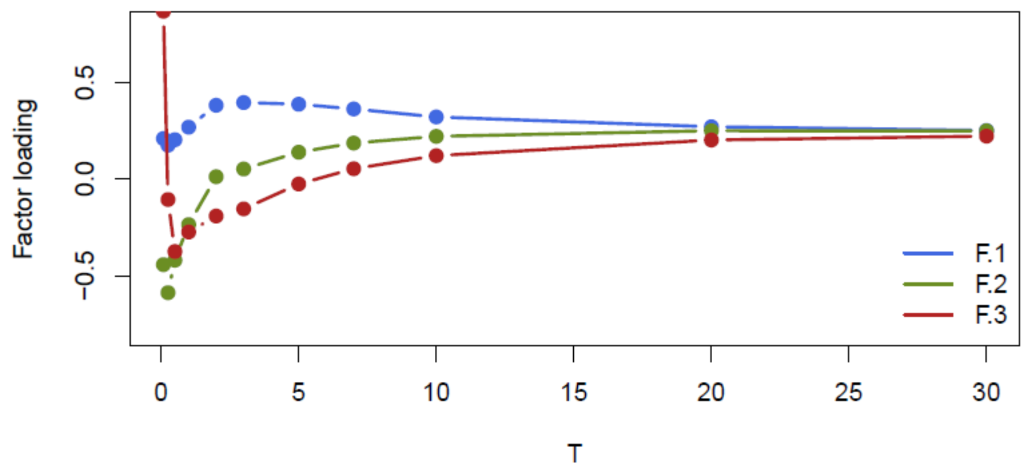

Figure 4.14: Loadings on the three statistical factors across maturities

All the weights in the first portfolio are positive and roughly the same. Any change in the first factor affects all the variables by similar amounts, causing approximately parallel shifts. So, this might be interpreted as a level factor. The factor loadings for the second principal component are increasing with maturity, so any change in this factor affects the slope of the yield curve. Finally, the factor loadings for the third principal component are decreasing and then increasing. Any change in this factor affects the curvature of the yield curve. This is illustrated next in Figure 4.15.

```
mu <- colMeans(dat)
par(mfrow=c(3,1))
for (i in 1:3) {
  plot(time, mu, ylim=c(2,5.3), type="b", col="royalblue", lwd=2, pch=19,
       ylab="Yield", xlab="T")
  lines(time, mu+beta[,i], type="b", lwd=2, lty=2, col="olivedrab", pch=19)
  lines(time, mu-beta[,i], type="b", lwd=2, lty=2, col="firebrick", pch=19)
  legend("bottomright", bty="n",
         c("mean", paste("mean+F.",i,sep=""), paste("mean-F.",i,sep="")),
         col=c("royalblue","olivedrab","firebrick"), lwd=2, lty=c(1,2,2))
}
```

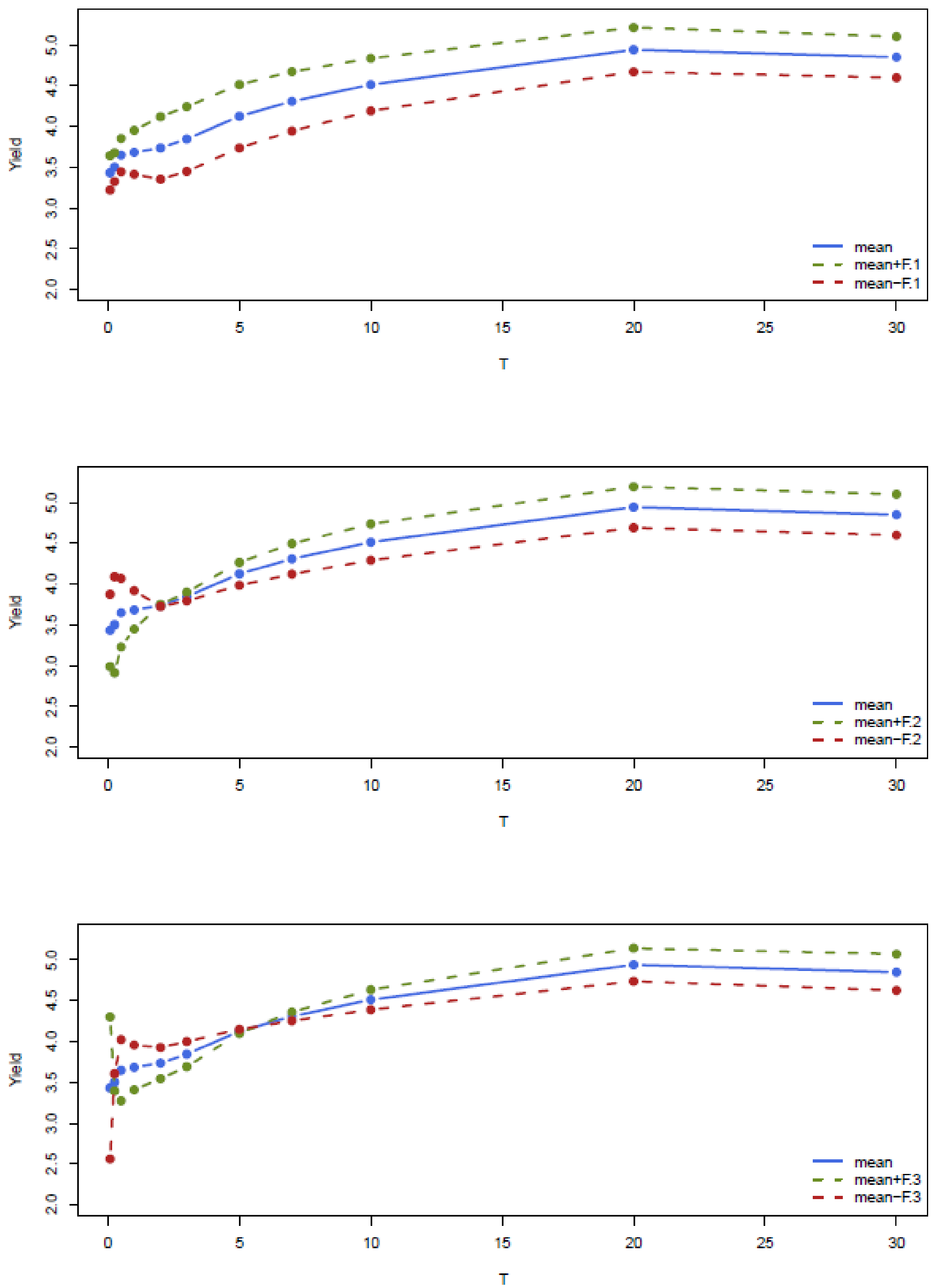

Figure 4.15: Effect of a unit change in the 3 statistical factors on the yield curve: level (shift), slope (tilt) and curvature (bend)

4.4 Factor Model Covariance and Risk Decomposition

4.4.1 *Factor Model Covariance*

Following Zivot & Jia-hui (2006), $R_{i,t}$, the return on asset i $(i = 1, 2, \dots, N)$ at time t $(t = 1, 2, \dots, T)$, is fitted with a factor model of the form,

$$R_{i,t} = \alpha_i + \boldsymbol{\beta}_i{}'\boldsymbol{f}_t + \varepsilon_{i,t} \tag{4.1}$$

where, α_i is the intercept, $\boldsymbol{f}_t$ is a $K \times 1$ vector of factor returns at time t, $\boldsymbol{\beta}_i$ is a $K \times 1$ vector of factor exposures for asset i and the error terms $\varepsilon_{i,t}$ are serially uncorrelated across time and contemporaneously uncorrelated across assets so that $\varepsilon_{i,t} \sim iid\ (0, \sigma_i^2)$. Thus, the variance of asset i's return is given by,

$$var(R_{i,t}) = \boldsymbol{\beta}_i{}'var\ (\boldsymbol{f}_t)\boldsymbol{\beta}_i + \sigma_i^2 \tag{4.2}$$

And the $N \times N$ covariance matrix of asset returns is,

$$var(\boldsymbol{R}) = \boldsymbol{\Omega} = \boldsymbol{B}\ var(\boldsymbol{F})\ \boldsymbol{B} + \boldsymbol{D} \tag{4.3}$$

where, $\boldsymbol{R}$ is the $N \times T$ matrix of asset returns, $\boldsymbol{B}$ is the $N \times K$ matrix of factor betas, $\boldsymbol{F}$ is the $K \times T$ matrix of factor returns and $\boldsymbol{D}$ is a diagonal matrix with σ_i^2 along the diagonal.

$fmCov$ computes the factor model covariance from a fitted factor model. The covariance of factor returns is the sample covariance matrix by default, but the option exists for the user to specify their own. Options for handling missing observations include $pairwise.complete.obs$ (default), $everything$, $all.obs$, $complete.obs$ and $na.or.complete$.

```
Omega <- fmCov(fit.pca)
# return correlation plot for all 15 assets
plot(fit.pca, which=8, a.sub=1:15, tl.cex=0.7)
```

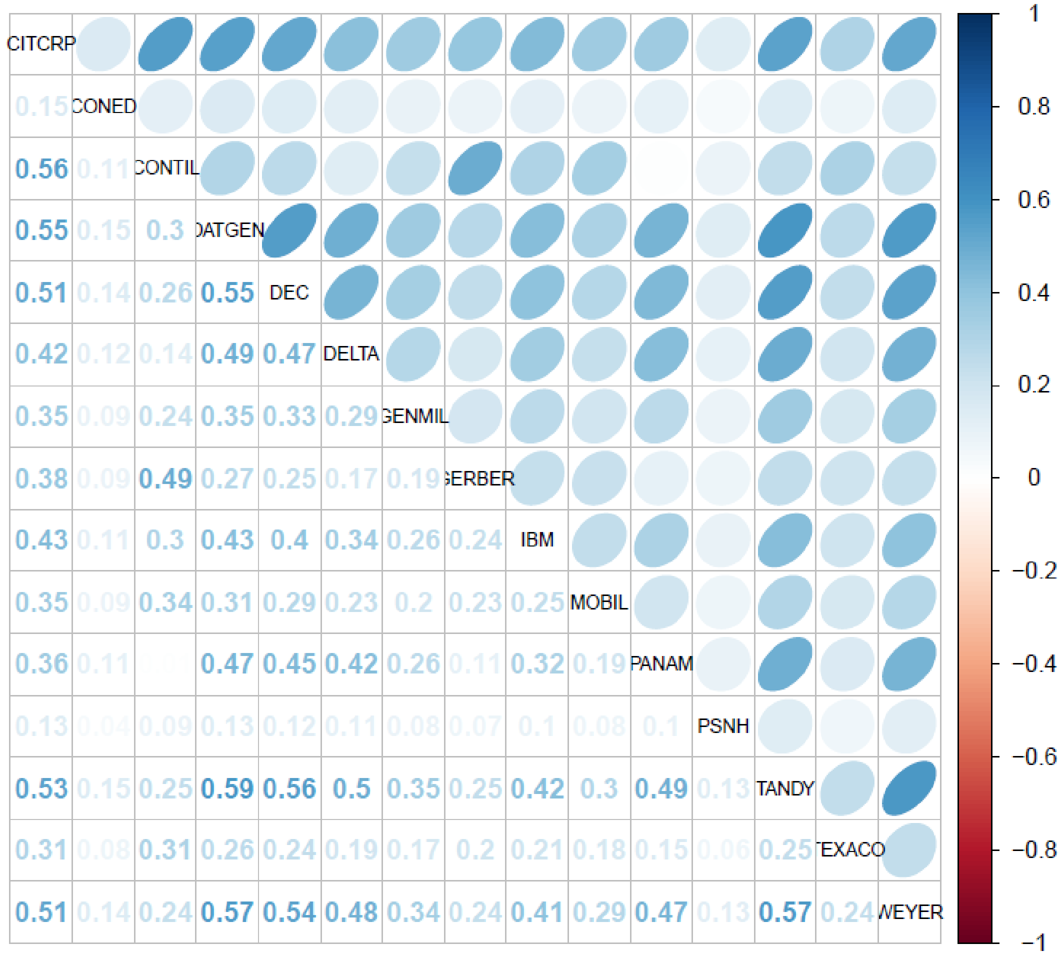

Figure 4.16: Factor model return correlation

4.4.2 *Standard Deviation Decomposition*

Following Meucci (2007), the standard deviation of asset i's return can be decomposed into the factor risk contributions using the factor model in equation 4.1 as shown below.

$$R_{i,t} = \boldsymbol{\beta}_i^{*\prime} \boldsymbol{f}_t^* \tag{4.4}$$

where, $\boldsymbol{\beta}_i^* = (\boldsymbol{\beta}_i' \; \sigma_i)$ and $\boldsymbol{f}_t^* = (\boldsymbol{f}_t' \; z_t)$, with $z_t \sim iid\ (0, 1)$ and σ_i is asset i's residual standard deviation. In other words, the residual is considered the $K+1$th risk factor, where the exposure to the residual is the residual standard deviation, and the residual factor returns are assumed to be iid $\sim(0,1)$. By Euler's theorem, the standard deviation of asset i, σ_i, can be decomposed as:

$$\sigma_i = \sum_{k=1}^{K+1} cSd_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^* * mSd_{i,k} \right) \tag{4.5}$$

where, $cSd_{i,k}$ and $mSd_{i,k}$ are the component and marginal contributions to risk from the k^{th} factor. While the component contribution is the total contribution to risk from factor k, the marginal contribution to risk is the effect on the asset's standard deviation due to an incremental change in its exposure to the k^{th} factor, holding all else constant. Computing the component and marginal risk contributions is straight forward. Formulas are given below and details are in Meucci (2007).

$$\sigma_i = \sqrt{\boldsymbol{\beta}_i^{*\prime} cov(\boldsymbol{F}^*)\, \boldsymbol{\beta}_i^*} \tag{4.6}$$

$$\boldsymbol{mSd}_i = \frac{cov(\boldsymbol{F}^*)\, \boldsymbol{\beta}_i^*}{\sigma_i} \tag{4.7}$$

$$\boldsymbol{cSd}_i = \boldsymbol{\beta}_i^* \odot \boldsymbol{mSd}_i \tag{4.8}$$

The covariance term is approximated by the sample covariance and $\odot$ represents element-wise multiplication. $fmSdDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total standard deviation and component, marginal and percentage component contributions for each asset are returned.

```
decomp <- fmSdDecomp(fit.pca)
names(decomp)

## [1] "Sd.fm" "mSd"   "cSd"   "pcSd"

# get the factor model standard deviation for all assets
decomp$Sd.fm

## CITCRP  CONED CONTIL DATGEN    DEC  DELTA GENMIL GERBER    IBM  MOBIL
## 0.0812 0.0507 0.1508 0.1280 0.0995 0.0964 0.0655 0.0883 0.0594 0.0808
##  PANAM   PSNH  TANDY TEXACO  WEYER
## 0.1324 0.1104 0.1280 0.0803 0.0854
```

```r
# get the component contributions to Sd; print first 6 assets
head(decomp$cSd)
```

```
##              F.1      F.2 Residuals
## CITCRP 0.04807 1.80e-03    0.0314
## CONED  0.00204 5.61e-07    0.0486
## CONTIL 0.04947 8.99e-02    0.0114
## DATGEN 0.07140 3.60e-03    0.0530
## DEC    0.04904 3.34e-03    0.0472
## DELTA  0.03409 6.65e-03    0.0557
```

```r
# plot the percentage component contributions to Sd for all 15 assets
plot(fit.pca, which=9, f.sub=1:2, a.sub=1:15)
```

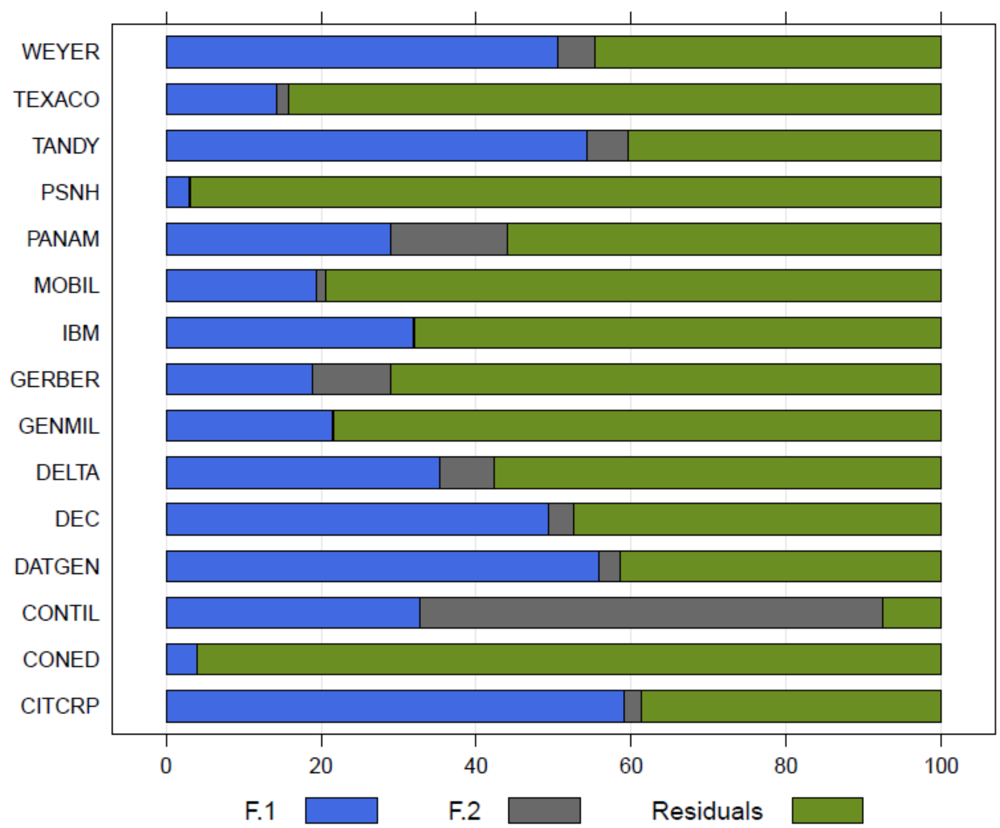

Figure 4.17: Percentage contribution to SD

4.4.3 *Value-at-Risk Decomposition*

Euler decomposition of return standard deviation shown above can also be applied to other risk measures such as value-at-risk (VaR) and expected shortfall (ES). The VaR version of equation 4.5 is given below. The value-at-risk of asset i can be decomposed as:

$$VaR_i = \sum_{k=1}^{K+1} cVaR_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^{*} * mVaR_{i,k}\right) \tag{4.9}$$

The marginal contribution to VaR_i is defined as the expectation of $\boldsymbol{F}^{*}$, conditional on the loss being equal to VaR_i. This is approximated as described in Epperlein and Smillie (2006) using a triangular smoothing kernel. $type$ gives the option to estimate VaR_i non-parametrically using the sample quantile (default) or assuming a normal distribution.

$fmVaRDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total VaR and component, marginal and percentage component contributions for each asset are returned.

```
# factor model VaR decomp using estimated factor return covariance (default)
# using tail probability = 10% and a parametric (normal) VaR estimation
decomp1 <- fmVaRDecomp(fit.apca, p=0.10, type="normal")
names(decomp1)

## [1] "VaR.fm"     "n.exceed"   "idx.exceed" "mVaR"       "cVaR"
## [6] "pcVaR"

# factor model Value-at-Risk; print first 6 assets
head(decomp1$VaR.fm)

##    IATV    ADCT    ADEX     ABM    ACTM     AFL
## -0.1826 -0.1201 -0.1147 -0.0617 -0.1530 -0.0655
```

```
# marginal factor contributions to VaR from 1st 4 factors; display top 6 assets
head(decomp1$mVaR[,1:4])

##          F.1      F.2       F.3      F.4
## IATV -0.0168  0.02472 -0.019789  0.01856
## ADCT -0.0397  0.02272  0.000112  0.00379
## ADEX -0.0312  0.03370 -0.008269 -0.01237
## ABM  -0.0358  0.00941 -0.006161  0.00201
## ACTM -0.0262  0.02560 -0.011434  0.00304
## AFL  -0.0364 -0.03292 -0.010552  0.01145

# plot the 1st 4 factors % component contributions to VaR for the 1st 6 assets
plot(fit.apca, which=11, f.sub=1:4, a.sub=1:6)
```

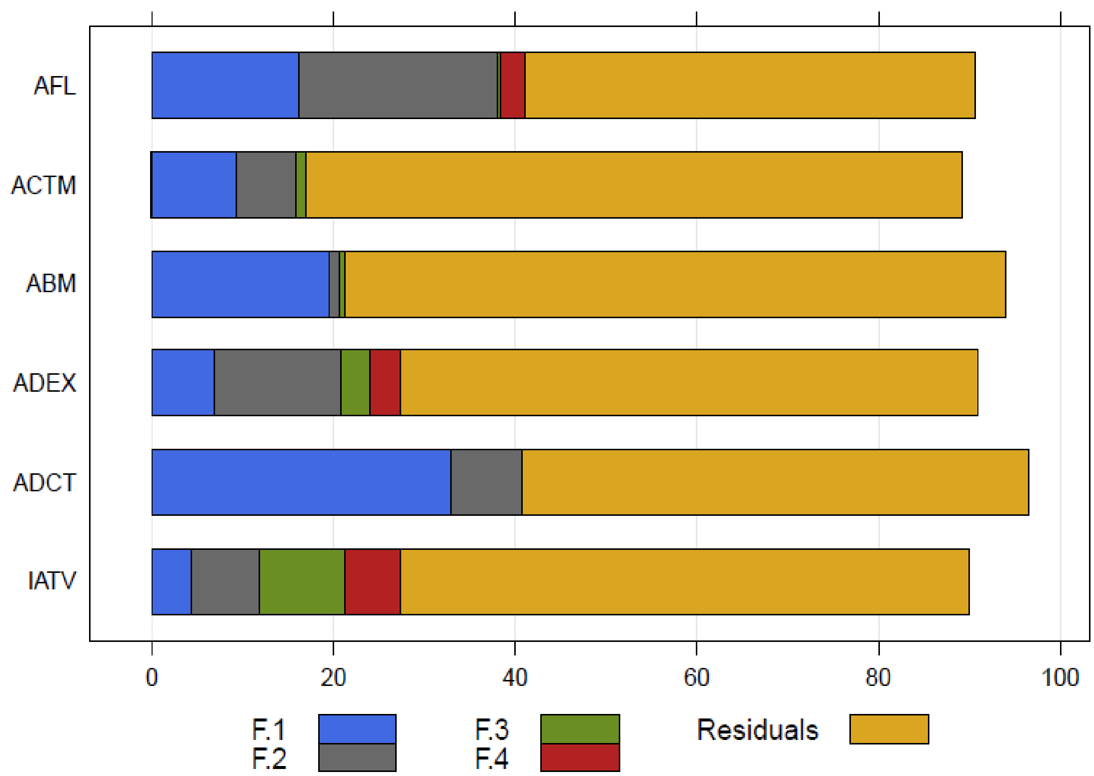

Figure 4.18: Percentage contributions to VaR

4.4.4 *Expected Shortfall Decomposition*

The expected shortfall (ES) version of equation 4.5 is given below. The expected shortfall of asset i can be decomposed as:

$$ES_i = \sum_{k=1}^{K+1} cES_{i,k} = \sum_{k=1}^{K+1}\left(\beta_{i,k}^{*} * mES_{i,k}\right) \tag{4.10}$$

The marginal contribution to ES_i is defined as the expectation of $\boldsymbol{F}^{*}$, conditional on the loss being less than or equal to ES_i. This is estimated as a sample average of the observations in that data window. Once again, $type$ gives the option to estimate ES_i non-parametrically using the sample quantile (default) or assuming a normal distribution.

$fmESDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total ES and component, marginal and percentage component contributions for each asset are returned.

```
# using normal distr. for computing ES (default is non-param. sample quantile)
decomp2 <- fmEsDecomp(fit.apca, type="normal")
names(decomp2)

## [1] "ES.fm" "mES"   "cES"   "pcES"

# factor model Expected Shortfall; print first 6 assets
head(decomp2$ES.fm)

##    IATV    ADCT    ADEX     ABM    ACTM     AFL
## -0.2939 -0.1933 -0.1846 -0.0994 -0.2463 -0.1055

# percentage component contributions to ES from 1st 4 factors; show 1st 6 assets
head(decomp2$pcES[,1:4])
```

```
##        F.1    F.2      F.3    F.4
## IATV  3.11  6.737 4.315813 3.7962
## ADCT 17.38  5.689 0.000139 0.1580
## ADEX 10.69 12.516 0.753523 1.6855
## ABM  14.16  0.975 0.418297 0.0445
## ACTM  7.56  7.221 1.440873 0.1015
## AFL  14.63 11.947 1.227134 1.4451

# plot the 1st 4 factors % component contributions to ES for the 1st 6 assets
plot(fit.apca, which=10, f.sub=1:4, a.sub=1:6)
```

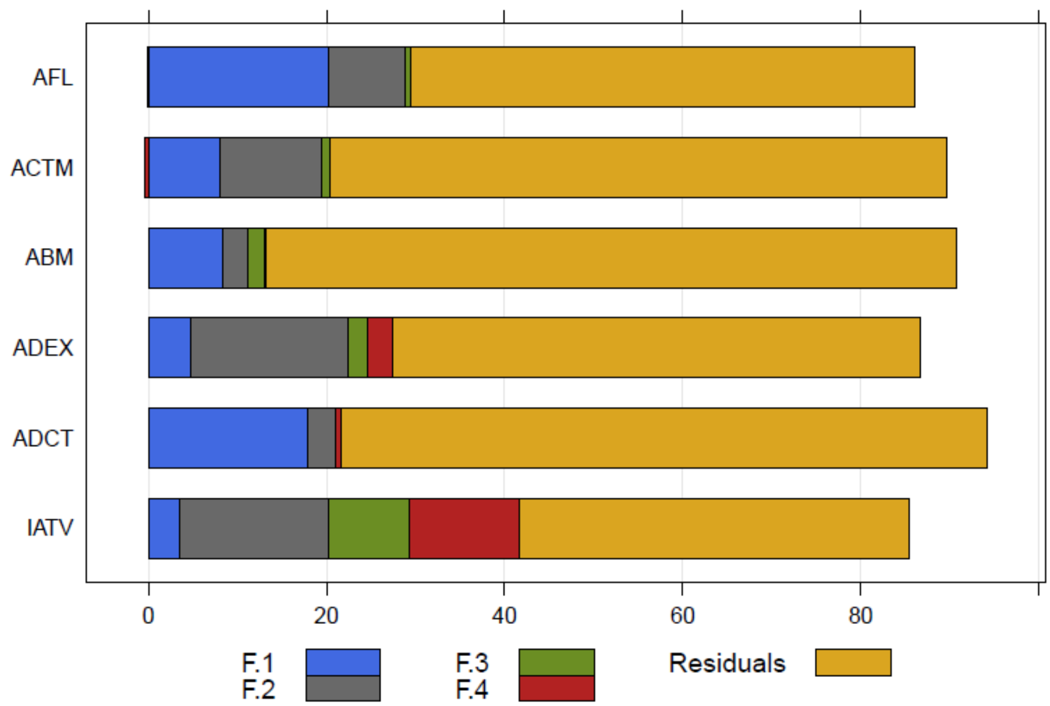

Figure 4.19: Percentage factor contribution to ES

4.5 Plot

Many types of plots for sfm objects have already been demonstrated. Let's look at all available arguments for plotting a sfm object.

```
## S3 method for class "sfm"
plot (x, which=NULL, f.sub=1:2, a.sub=1:6, n.top=3,
      plot.single=FALSE, asset.name,
      colorset=c("royalblue","firebrick","olivedrab","firebrick","goldenrod",
                 "mediumorchid","deepskyblue","chocolate","darkslategray"),
      legend.loc="topleft", las=1, lwd=2, maxlag=15, eig.max=0.9,
      cum.var=TRUE, ...)
```

4.5.1 *Group Plots*

This is the default option for plotting. Simply running $plot\ (fit)$, where fit is any sfm object, will bring up the following menu for group plots.

```
plot(fit.pca)

# Make a plot selection (or 0 to exit):
#
#  1: Screeplot of eigenvalues
#  2: Time series plot of estimated factors
#  3: Estimated factor loadings
#  4: Histogram of R-squared
#  5: Histogram of residual volatility
#  6: Scatterplot matrix of residuals, with histograms, density overlays,
#     correlations and significance stars
#  7: Factor model residual correlation
#  8: Factor model return correlation
#  9: Factor contribution to SD
# 10: Factor contribution to ES
# 11: Factor contribution to VaR
# 12: Factor mimicking portfolio weights - top long and short positions in each
#     factor
# 13: Asset correlations - top long and short positions in each factor
#
# Selection:
```

Note: Only a subset of assets and factors selected by $a.sub$ and $f.sub$ are plotted. The first 2 factors and first 6 assets are shown by default. Argument $cum.var$ applies to group plot 1, and specifies whether the cumulative fraction of the variance is printed above each bar in the screeplot of eigenvalues. Argument $eig.max$ also applies to group plot 1, and displays the largest eigenvalues that cumulatively explain a specified percent of the total variance. Argument $n.top$ applies to group plots 12 and 13, which involve summarizing the factor mimicking portfolios, and specifies the number of top positions to display.

4.5.2 *Menu and Looping*

If the plot type argument $which$ is not specified, a menu prompts for user input. In case multiple plots are needed, the menu is repeated after each plot (enter 0 to exit). User can also input a numeric vector of plot options via argument $which$.

4.5.3 *Individual Plots*

Setting $plot.single = TRUE$ enables individual asset plots. If there is more than one asset fit by the fitted object x, $asset.name$ is also necessary. In case the sfm object x contains only a single asset's fit, $plot.sfm$ can infer $asset.name$ without user input. Here's the individual plot menu.

```
plot(fit.pca, plot.single=TRUE, asset.name="DATGEN")

# Make a plot selection (or 0 to exit):
#
#  1: Actual and fitted asset returns
#  2: Actual vs fitted asset returns
#  3: Residuals vs fitted asset returns
#  4: Sqrt. of modified residuals vs fitted
#  5: Residuals with standard error bands
```

```
#  6: Time series of squared residuals
#  7: Time series of absolute residuals
#  8: SACF and PACF of residuals
#  9: SACF and PACF of squared residuals
# 10: SACF and PACF of absolute residuals
# 11: Non-parametric density of residuals with normal overlaid
# 12: Non-parametric density of residuals with skew-t overlaid
# 13: Histogram of residuals with non-parametric density and normal overlaid
# 14: QQ-plot of residuals
# 15: CUSUM test-Recursive residuals
# 16: CUSUM test-LS residuals
# 17: Recursive estimates (RE) test of LS regression coefficients
# 18: Rolling estimates over a 24-period observation window
#
# Selection:
```

Here are some examples which don't need interactive user input. These are individual plots for the DATGEN asset in the PCA fit illustrated earlier.

```
plot(fit.pca, plot.single=TRUE, asset.name="DATGEN", which=5)
```

```
plot(fit.pca, plot.single=TRUE, asset.name="DATGEN", which=10)
```

```
plot(fit.pca, plot.single=TRUE, asset.name="DATGEN", which=14)
grid()
```

```
plot(fit.pca, plot.single=TRUE, asset.name="DATGEN", which=11)
```

```
plot(fit.pca, plot.single=TRUE, asset.name="DATGEN", which=12)
```

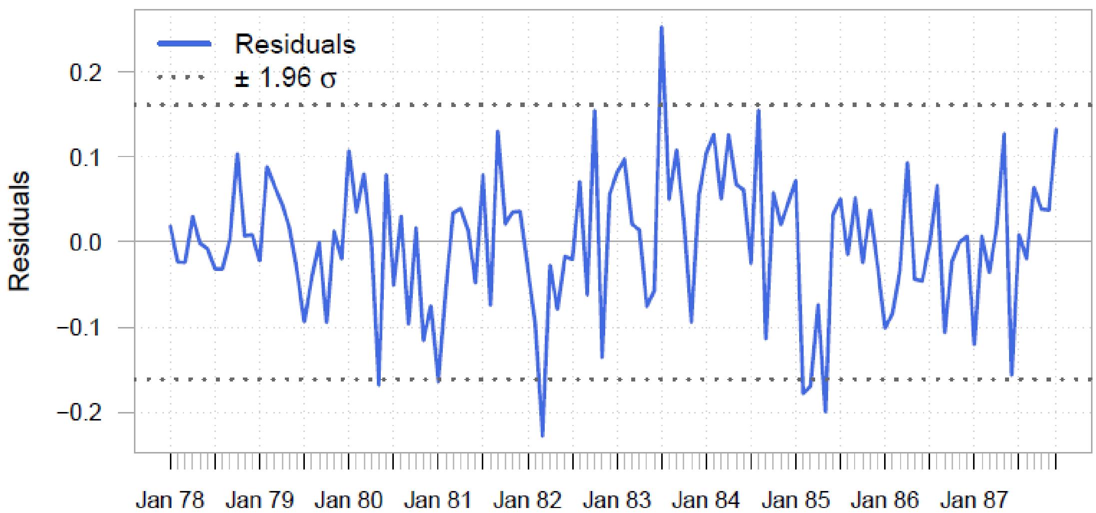

Figure 4.20: Time-series plot of residuals with standard error bands: DATGEN

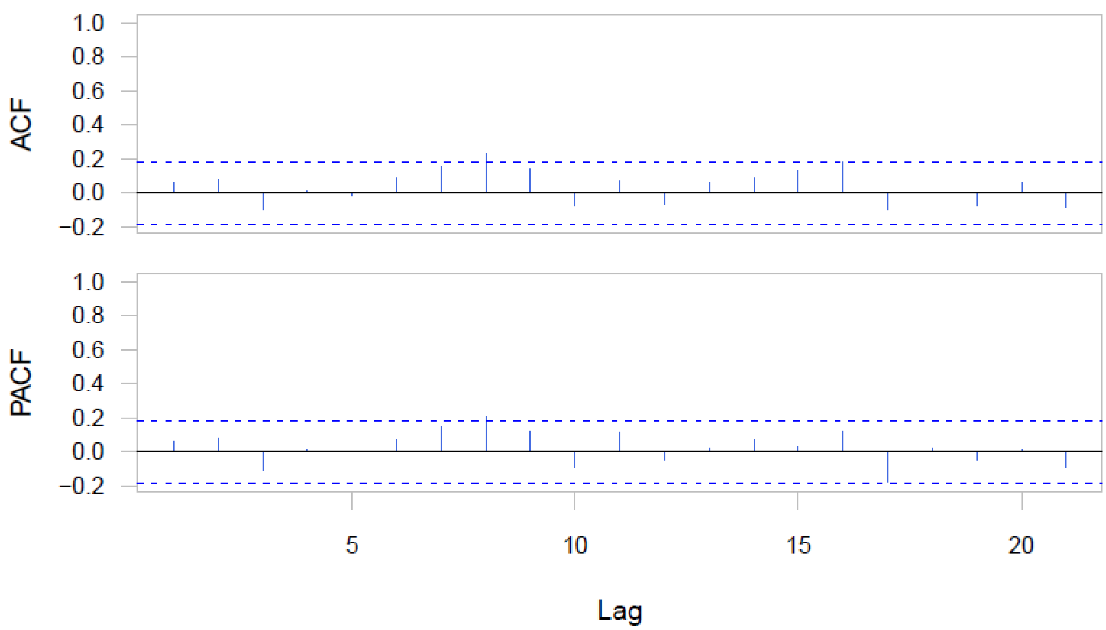

Figure 4.21: SACF and PACF of absolute residuals: DATGEN

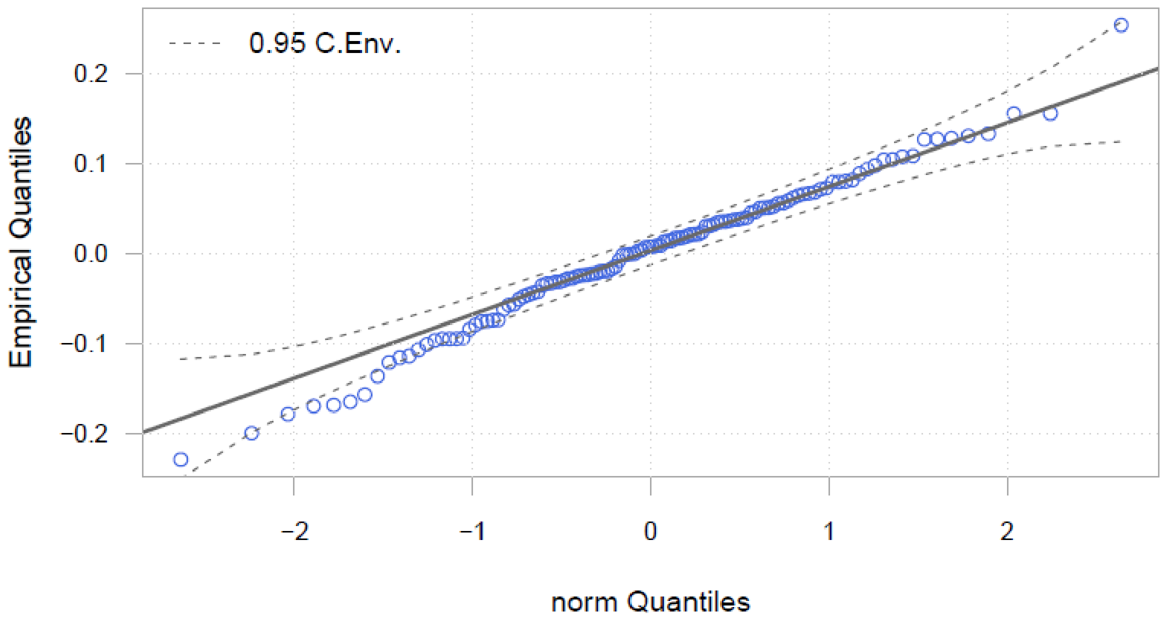

Figure 4.22: QQ-plot of residuals: DATGEN

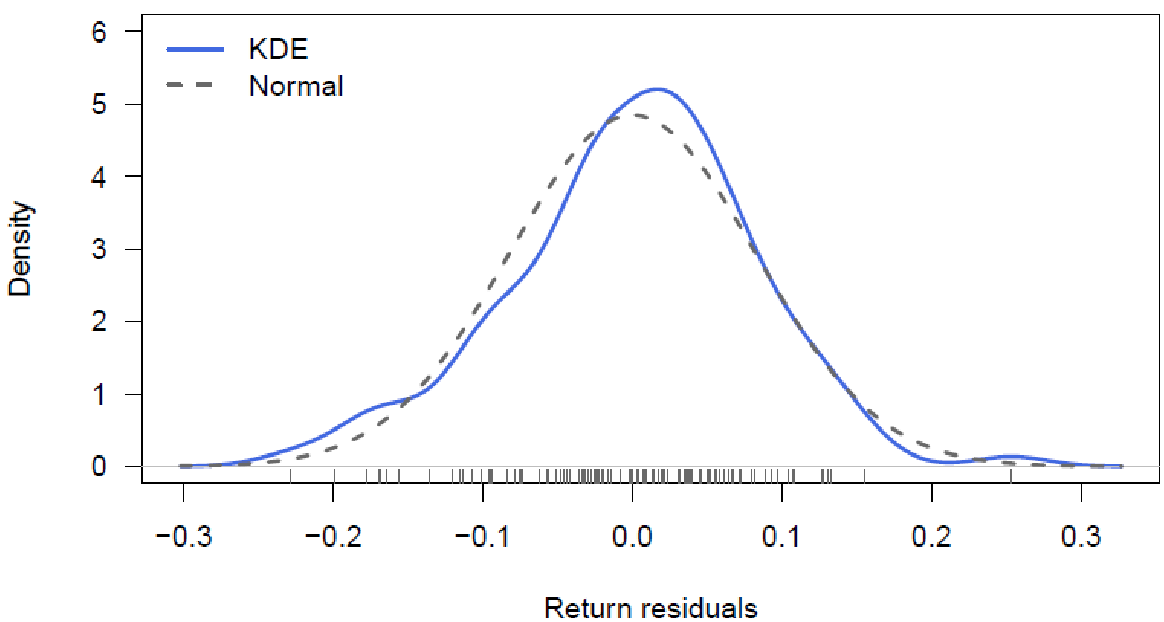

Figure 4.23: Non-parametric density of residuals with normal overlaid: DATGEN

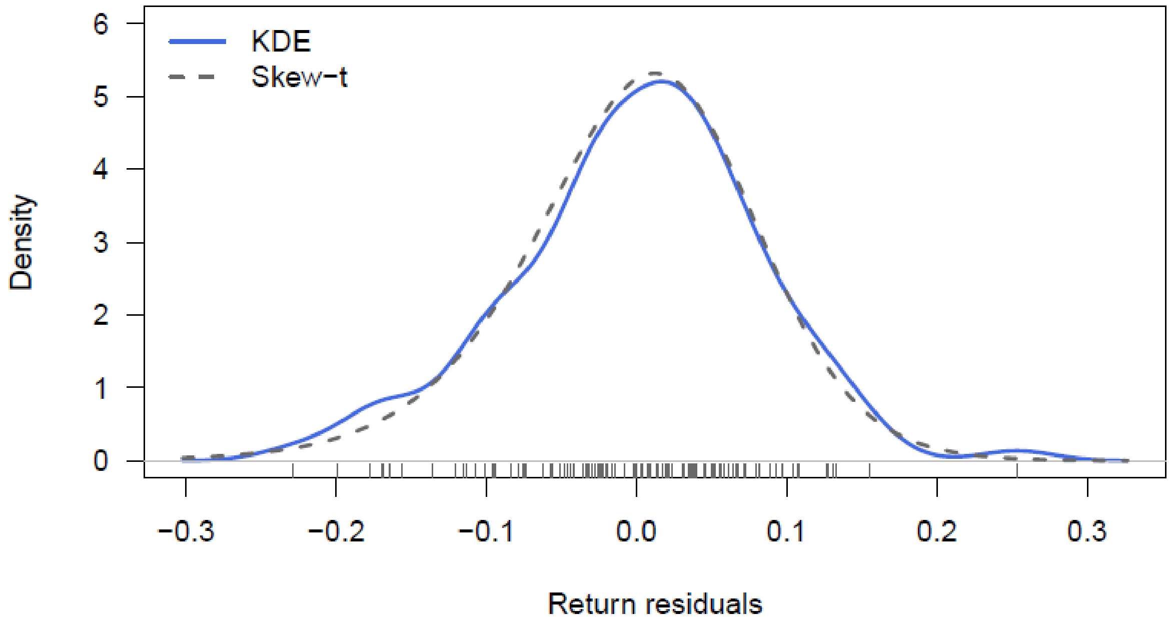

Figure 4.24: Non-parametric density of residuals with skew-t overlaid: DATGEN

Chapter 5. FITTING FUNDAMENTAL FACTOR MODELS

The purpose of this vignette is to demonstrate the use of the $fitFfm$ function and related control, analysis and plot functions in the $factorAnalytics$ package.

5.1 Overview

5.1.1 *Load Package*

The latest version of the $factorAnalytics$ package used in this vignette is hosted in the publicly available GitHub repository https://github.com/sangeeuw/factorAnalytics. There are plans for further updates to the package before its moved back to R-Forge and released on CRAN later this year. The package can be installed from GitHub as shown below.

```
library(devtools)
install_github("sangeeuw/factorAnalytics")
```

```
# load the package and its dependencies
library(factorAnalytics)
options(digits=3)
```

The focus of this vignette is on the $fitFfm$ function and related methods. The original function was designed by Doug Martin and initially implemented in S-PLUS by several University of Washington Ph.D. students: Christopher Green, Eric Aldrich, and Yindeng Jiang. Guy Yollin ported the function to R and Yi-An Chen modified that code as part of Google Summer of Code (GSOC) 2013. Sangeetha Srinivasan tested and expanded the functionalities and S3 methods as part of GSOC in 2014 and 2015. Doug Martin, Avinash Acharya, Lingjie Yi and Chindhanai Uthaisaad added options to fit EWMA or GARCH model for errors, enabled a market plus industry and/or sector and/or country model specification, etc. as part of GSOC 2016 and 2017. Refer to the other fundamental factor model vignette by Avinash Acharya for more examples elaborating on these recent functionalities and reporting functions.

5.1.2 *Summary of Related Functions*

Here's a summary of the fit function and related S3 methods (generic accessor functions) demonstrated in this vignette:

- $fitFfm\,(data,\ asset.var,\ ret.var,\ date.var, exposures.var,\ weight.var,$ $fit.method,\ rob.stats,\ full.resid.cov,\ z.score,\ add.intercept,\ lag.exposures,$ $resid.scale.type,\ lambda,\ GARCH.params,\ GARCH.MLE,\ std.return,$ $analysis,\ target.vol,\ \ldots)$: Fits a fundamental factor model for one or more asset returns or excess returns using T cross-sectional regressions a.k.a the "BARRA" approach (detailed in Grinold & Kahn, 2000), where T is the number of periods. Available fit methods include Least squares (LS), weighted least squares (WLS), robust regression (Rob) and weighted-robust regression (W-Rob). Options for computing residual variances include sample variance, EWMA, Robust EWMA and GARCH (1,1). An object of class ffm containing the fitted objects, factor exposures, estimated factor returns, R^2, residual volatility, etc. is returned.
- $coef\ (object,\ \ldots)$: Returns a $data.frame$ containing the coefficients (intercept and factor betas) for the last time period for all assets.
- $fitted\,(object,\ \ldots)$: Returns an xts data object of fitted asset returns from the factor model for all assets.
- $residuals\,(object,\ \ldots)$: Returns an xts data object of residuals from the fitted factor model for all assets.

- $fmCov\ (object,\ \ldots)$: Returns the $N \times N$ symmetric covariance matrix for asset returns based on the fitted factor model, using exposures from the last period.
- $fmSdDecomp\ (object,\ use,\ \ldots)$: Returns a list containing the standard deviation of asset returns based on the fitted factor model and the marginal, component and percentage component factor contributions estimated from the given sample. use specifies how missing values are to be handled.
- $fmVaRDecomp\ (object,\ factor.cov,\ p,\ type,\ use,\ \ldots)$: Returns a list containing the value-at-risk (VaR) for asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. $type$ specifies if VaR computation should be non-parametric (sample quantile) or based on a Normal distribution. And, p specifies the confidence level.
- $fmEsDecomp\ (object,\ factor.cov,\ p,\ type,\ use,\ \ldots)$: Returns a list containing the expected shortfall (ES) for asset returns based on the fitted factor model and the estimated marginal, component and percentage component factor contributions. $factor.cov$ allows for user-specified factor covariance matrix; defaults to the sample covariance of historical factor returns. $type$ specifies if ES computation should be non-parametric (sample quantile) or based on a Normal distribution. And, p specifies the confidence level.
- $plot\ (x,\ \ldots)$: The plot method for class ffm can be used for plotting factor model characteristics of a group of assets (default) or an individual asset. The user can select the type of plot either from the menu prompt or directly via argument $which$. In case multiple plots are needed, the menu is repeated after each plot (enter 0 to exit). User can also input a numeric vector of plot options via $which$.

- $predict\ (object,\ newdata,\ pred.date,\ \ldots)$: The $predict$ method for class ffm returns a vector or matrix of predicted values for new or simulated values of the fundamental characteristics. $pred.date$ allows user to choose the relevant date for the estimated factor exposures to be used in the prediction.
- $print\ (object,\ digits,\ \ldots)$: The $print$ method for class ffm prints the call, factor model dimension and summary statistics for the estimated factor returns, cross-sectional R^2 values and residual variances from the fitted object.
- $summary\ (object,\ \ldots)$: The summary method for class ffm returns an object of class $summary.ffm$ containing the summaries of the fitted objects. Printing the factor model summary object displays the call, estimated factor returns, R^2 and residual volatility for each period.

A complete list of related methods is shown below.

```
methods(class="ffm")

##  [1] coef          fitted        fmCov         fmEsDecomp    fmRsq
##  [6] fmSdDecomp    fmTstats      fmVaRDecomp   plot          portEsDecomp
## [11] portSdDecomp  portVaRDecomp portVolDecomp predict       print
## [16] repRisk       residuals     riskDecomp    summary
## see '?methods' for accessing help and source code
```

5.1.3 *Data*

The following examples primarily use the $Stock.df$ dataset. It contains fundamental and monthly return data for 447 stocks listed on the NYSE over an 8-year period. The dataset is balanced, i.e., every asset has a complete set of observations for all variables in each period. The following queries help understand key aspects of the dataset:

```
# load the dataset into the environment
data(Stock.df)
# get a list of the variable names
colnames(stock)

##  [1] "DATE"              "RETURN"            "TICKER"
##  [4] "PRICE"             "VOLUME"            "SHARES.OUT"
##  [7] "MARKET.EQUITY"     "LTDEBT"            "NET.SALES"
## [10] "COMMON.EQUITY"     "NET.INCOME"        "STOCKHOLDERS.EQUITY"
## [13] "LOG.MARKETCAP"     "LOG.PRICE"         "BOOK2MARKET"
## [16] "GICS"              "GICS.INDUSTRY"     "GICS.SECTOR"

# time period covered in the data
range(stock[,"DATE"])

## [1] "1996-02-29" "2003-12-31"

# number of stocks
length(unique(stock[,"TICKER"]))

## [1] 447

# count stocks by GICS sector as of the last time period
stocklist<-subset(stock,DATE=="2003-12-31")
table(stocklist$GICS.SECTOR)

##
##     Consumer Discretionary           Consumer Staples
##                         86                         30
##                     Energy                 Financials
##                         17                         55
##                Health Care                Industrials
##                         35                         89
##     Information Technology                  Materials
##                         57                         32
## Telecommunication Services                  Utilities
##                          6                         40
```

5.2 FITTING A FUNDAMENTAL FACTOR MODEL

A fundamental factor model uses observed cross-sectional asset characteristics such as dividend yield, earnings yield, book-to-market ratio, market capitalization, sector or industry classification, price volatility, price momentum, leverage, etc. to determine common risk factors that contribute to asset returns. Chapter 15 from Zivot & Jia-hui (2006) serves as a good reference for a description of the different multi-factor models, estimation methods and examples in S-PLUS.

There are 2 main approaches to estimating the fundamental factor model - the "BARRA" approach (explained in Grinold and Kahn, 2000) and the "Fama-French" approach (introduced by Fama and French, 1992). In the "BARRA" approach, the observed fundamental attributes are the factor betas and the unknown factor returns are estimated via cross-sectional regressions for each period. Due to cross-sectional heteroskedasticity of asset returns, ordinary least squares (OLS) estimation of the factor returns is inefficient. So, weighted least squares regression is performed as a second step to get efficient estimates, with the inverse of the estimated residual variances or market cap used as weights. $fitFfm$ described in this vignette uses the "BARRA" approach.

In the "Fama-French" approach, the factor returns are the observed returns of a hypothetical hedge portfolio that's long/short the top/bottom quintile of stocks for a given attribute (ex: market cap for the size factor). After the factor returns are computed for each characteristic, each asset's factor exposures are estimated via a time series regression.

Let's look at the arguments for $fitFfm$.

```
args(fitFfm)

## function (data, asset.var, ret.var, date.var, exposure.vars,
##     weight.var = NULL, fit.method = c("LS", "WLS", "Rob", "W-Rob"),
##     rob.stats = FALSE, full.resid.cov = FALSE, z.score = c("none",
##         "crossSection", "timeSeries"), add.intercept = FALSE,
##     lag.exposures = TRUE, resid.scale.type = c("stdDev", "EWMA",
##         "robEWMA", "GARCH"), GARCH.params = list(omega = 0.09,
##         alpha = 0.1, beta = 0.81), lambda = 0.9, GARCH.MLE = FALSE,
##     std.return = FALSE, analysis = c("none", "ISM", "NEW"), target.vol = 0.06,
##     ...)
## NULL
```

The default model fitting method is ordinary least squares (LS) regression, with the option to choose robust regression (Rob), weighted least squares (WLS) or weighted robust regression (W-Rob). The different model fitting options are demonstrated in the following sections. If weighted regression (WLS or W-Rob) is chosen, inverse of the residual variances are used as weights. $resid.scale.type$ allows the user to choose the method for computing residual variances - sample variance, EWMA, Robust EWMA and GARCH (1,1).

$z.score$ provides the option to standardize factor exposures across assets or across periods. $weight.var$ allows the user to specify higher weights to some assets when estimating factor exposures; for example, using the market cap of stocks as their weights. $add.intercept$ gives the option to add an intercept term for fitting a $Market + Sector$ or a $Market + Country + Sector$ model. These models can simultaneously include other style factors. $lag.exposures$ gives the option to use the factor exposures from the previous period to estimate factor returns for the current period. $full.resid.cov$ provides the option to choose between a diagonal vs. full residual covariance matrix. And, $rob.stats$ allows for robust estimates of covariance, correlation, location and univariate scale.

These and other control parameters are demonstrated in the following sections.

5.2.1 *Single Factor Model*

Here's an example of a single factor model using the book-to-market ratio, a proxy for the value factor, as the explanatory variable for the returns of 447 stocks in the dataset.

```
# Single Factor Model
fit.single <- fitFfm(data=stock, asset.var="TICKER", ret.var="RETURN",
                     date.var="DATE", exposure.vars="BOOK2MARKET")
```

The resulting object, $fit.single$, has the following attributes.

```
class(fit.single)

## [1] "ffm"

names(fit.single)

##  [1] "factor.fit"      "beta"            "factor.returns"
##  [4] "residuals"       "r2"              "factor.cov"
##  [7] "g.cov"           "resid.cov"       "return.cov"
## [10] "restriction.mat" "resid.var"       "call"
## [13] "data"            "date.var"        "ret.var"
## [16] "asset.var"       "exposure.vars"   "weight.var"
## [19] "fit.method"      "asset.names"     "factor.names"
## [22] "time.periods"    "activeWeights"   "activeReturns"
## [25] "IR"
```

The component $factor.fit$ contains a list of lm or $lmRob$ objects, one for each period. The fitted objects are of class lm if $fit.method = "LS"$ or $"WLS"$, or class $lmRob$, if $fit.method = "Rob"$ or $"W-Rob"$. The component $factor.returns$ contains the estimated factor returns and $beta$ contains the factor exposures from the last period. While, $r2$ and $resid.var$ denote the regression R^2 and estimated residual variance respectively. The estimated covariance matrix of factor returns, residuals and asset returns are given by $factor.cov$, $resid.cov$ and $return.cov$ respectively. The remaining components contain the input choices and the data.

The print method displays a summary of the T cross-sectional regressions, where T is the number of periods, as shown below.

```
# print the fitted "ffm" object
fit.single

##
## Call:
## fitFfm(data = stock, asset.var = "TICKER", ret.var = "RETURN",
##     date.var = "DATE", exposure.vars = "BOOK2MARKET")
##
## Model dimensions:
## Factors  Assets Periods
##       1     447      94
##
## Factor returns across periods:
##   BOOK2MARKET
##  Min.   :-0.0332
##  1st Qu.:-0.0053
##  Median : 0.0045
##  Mean   : 0.0048
##  3rd Qu.: 0.0139
##  Max.   : 0.0446
##
## R-squared values across periods:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0010  0.0043  0.0076  0.0122  0.0475
##
## Residual Variances across assets:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0036  0.0141  0.0209  0.0294  0.0347  0.1590
```

Figure 5.1 shows a scatter plot of residuals for the 1st 6 stocks in the last period, including histograms, density overlays, correlations and significance stars. (A detailed list of plot options is provided later in section 5.4.)

```
# plot residual correlations for the single factor model
# default is to plot the 1st 6 assets
plot(fit.single, which=6, f.sub=1)
```

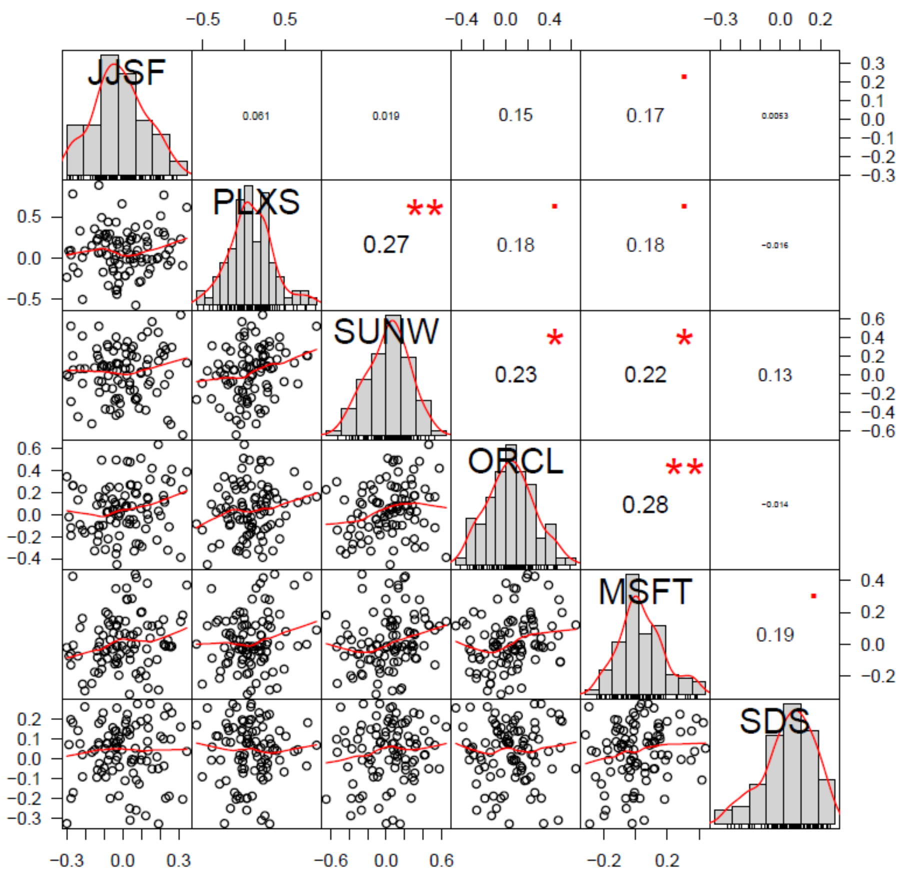

Figure 5.1: Single Factor Model: Residual Correlations

Note the high residual correlation between MSFT and ORCL; this might be due to their exposure to other omitted factors such as a sector/industry risk factor for "Software & Services". The next section demonstrates fitting an industry/sector factor model for these stocks.

```
# GICS industry/sector classification (1st 6 stocks; penultimate time period)
subset(stock,DATE=="2003-11-28")[1:6,c("TICKER","GICS.INDUSTRY","GICS.SECTOR")]
```

```
##     TICKER                    GICS.INDUSTRY            GICS.SECTOR
## 94    JJSF         Food, Beverage & Tobacco       Consumer Staples
## 189   PLXS Technology Hardware & Equipment Information Technology
## 284   SUNW Technology Hardware & Equipment Information Technology
## 379   ORCL             Software & Services Information Technology
## 474   MSFT             Software & Services Information Technology
## 569    SDS             Software & Services Information Technology
```

5.2.2 *BARRA-type Industry Factor Model*

A BARRA-type industry (sector) factor model is a fundamental factor model with multiple factors. Here is a demonstration using the 447 NYSE stocks in our dataset; where the 10 mutually exclusive GICS sector classifications are the 10 factors. The factor exposures will be dummy variables that indicate if a given stock belongs to a sector or not. Mutually exclusive sectors mean that each stock belongs to a unique sector in any given period. Notice that the average R^2 from the sector model is significantly higher (and average residual correlations are lower) than the single factor model.

```
# Sector Factor Model
fit.sector <- fitFfm(data=stock, asset.var="TICKER", ret.var="RETURN",
                     date.var="DATE", exposure.vars="GICS.SECTOR")
# compare r2: single factor vs. sector model
summary(fit.single$r2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0010  0.0043  0.0076  0.0122  0.0475
```

```
summary(fit.sector$r2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.023   0.060   0.113   0.137   0.195   0.519
```

```
# compare avg. non-diagonal correlations: single factor vs. sector model
mean(cor(residuals(fit.single))[cor(residuals(fit.single))!=1])

## [1] 0.0923

mean(cor(residuals(fit.sector))[cor(residuals(fit.sector))!=1])

## [1] -0.00121
```

Let's look at the fitted factor model from the last period in the data. We observe that "Energy", "Materials" and "Telecomm" sectors had particularly strong returns, with estimated factor returns over 10% for that month[22].

```
# print the summary from the last period's fit
num.periods <- length(fit.sector$time.periods)
summary(fit.sector$factor.fit[[num.periods]])

##
## Call:
## FUN(formula = ..1, data = data[x, , drop = FALSE], na.action = ..3,
##     contrasts = ..2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3984 -0.0806 -0.0067  0.0780  0.5362
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## Consumer Discretionary      0.0124     0.0154    0.80   0.4236
## Consumer Staples            0.0480     0.0261    1.84   0.0666 .
## Energy                      0.1131     0.0347    3.26   0.0012 **
## Financials                  0.0466     0.0193    2.41   0.0162 *
## Health Care                 0.0358     0.0242    1.48   0.1398
```

[22] "Energy" stocks rebounded in 2003 from the beating they took in 2002 following the Enron scandal. "Telecomm" stocks benefited from the increased spending by companies investing in internet-based phone systems during this period.

```
## Industrials                  0.0415     0.0152    2.74    0.0064 **
## Information Technology       0.0339     0.0190    1.79    0.0744 .
## Materials                    0.1146     0.0253    4.53   7.6e-06 ***
## Telecommunication Services   0.1025     0.0584    1.76    0.0799 .
## Utilities                    0.0684     0.0226    3.02    0.0027 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.143 on 437 degrees of freedom
## Multiple R-squared:  0.131,Adjusted R-squared:  0.112
## F-statistic: 6.61 on 10 and 437 DF,  p-value: 1.44e-09
```

Figure 5.2 shows the distribution of estimated monthly sector returns (from 1996 – 2003) in descending order of their mean. We find that the "Information Technology" sector had the highest average return (perhaps not surprising, given that the dataset covers the dot-com bubble).

```
# plot distribution of factor returns by sector sorted by means
plot(fit.sector, which=1, colorset="black", f.sub=1:10, lwd=1, sort.by="mean")
```

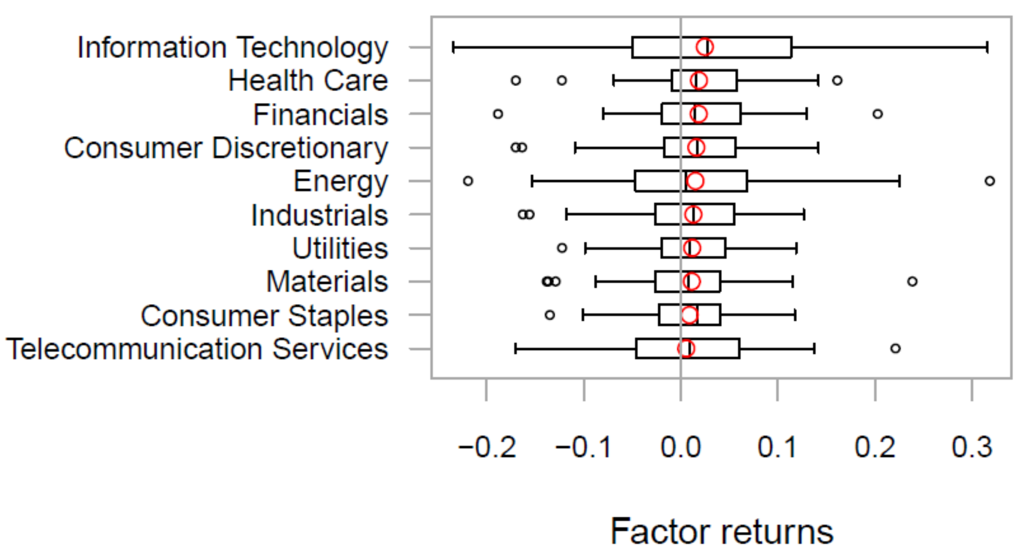

Figure 5.2: Sector Model: Distribution of factor returns sorted by mean

An extension of the above sector model is to isolate the market effect using an intercept term and reparametrizing the sector exposures so that they are measured relative to the common market factor. Here, the intercept is interpreted as the return to the market factor (sum of all sectors), while the other factors are excess returns for the sector over the market. The methodology behind this model was introduced in the context of a common country effect in Heston & Rouwenhorst (1995) and explained in Menchero (2010) as well. In $fitFfm$ the $market + sector$ model can be opted via the parameter $add.intercept$ as shown below.

```
# Market + Sector Factor Model
fit.mkt.sector <- fitFfm(data=stock, asset.var="TICKER", ret.var="RETURN",
                         date.var="DATE", exposure.vars="GICS.SECTOR",
                         add.intercept=TRUE)


# coefficients (factor exposures) for first 10 assets
t(coef(fit.mkt.sector)[1:10,])
```

```
##                            JJSF PLXS SUNW ORCL MSFT SDS TROW HON EMC XRIT
## Market                        1    1    1    1    1   1    1   1   1    1
## Consumer Discretionary        0    0    0    0    0   0    0   0   0    0
## Consumer Staples              1    0    0    0    0   0    0   0   0    0
## Energy                        0    0    0    0    0   0    0   0   0    0
## Financials                    0    0    0    0    0   0    1   0   0    0
## Health Care                   0    0    0    0    0   0    0   0   0    0
## Industrials                   0    0    0    0    0   0    0   1   0    0
## Information Technology        0    1    1    1    1   1    0   0   1    1
## Materials                     0    0    0    0    0   0    0   0   0    0
## Telecommunication Services    0    0    0    0    0   0    0   0   0    0
## Utilities                     0    0    0    0    0   0    0   0   0    0
```

```
# plot distribution of factor returns by sector sorted by means
plot(fit.mkt.sector, which=1, colorset="black", f.sub=1:10, lwd=1, sort.by="mean")
```

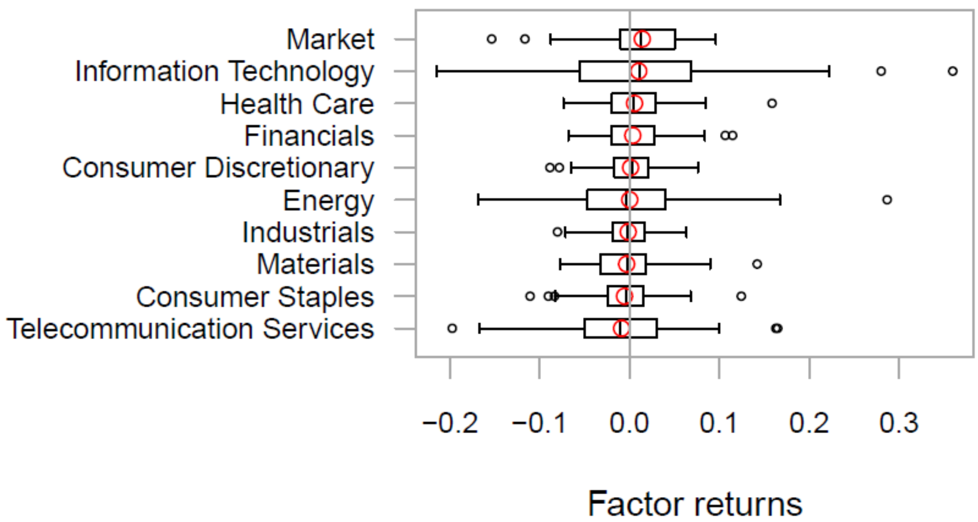

Figure 5.3: Market + Sector Model: Distribution of factor returns sorted by mean

The reparameterization of the market factor hasn't changed the order of sectors by mean factor return. The reader can verify that R^2 and other fit statistics haven't changed either.

5.2.3 *Multi-factor Model with Sector and Style Characteristics*

A fundamental factor model can simultaneously include both quantitative style factors, such as size (market cap), value (book-to-price ratio), price momentum etc., as well as sector/industry classifications. The next example demonstrates fitting a multi-factor model including 2 style factors, size and value, in addition to the sector model. Note that the adjusted-R^2 has improved.

```
# Market + Sector Factor Model
fit.style.sector <- fitFfm(data=stock, asset.var="TICKER", ret.var="RETURN",
  date.var="DATE", exposure.vars=c("GICS.SECTOR","LOG.MARKETCAP","BOOK2MARKET"))

# check if average adjusted R-squared improved vs. pure sector model
# adjusted r2 = 1 - ((n-1)*(1-r2)/(n-p-1))
print(adj.r2_style.sector <- 1-((447-1)*(1-mean(fit.style.sector$r2))/(447-12-1)))

## [1] 0.126
```

```
print(adj.r2_sector <- 1-((447-1)*(1-mean(fit.sector$r2))/(447-10-1)))

## [1] 0.117
```

Figure 4.4, Figure 4.5 and Figure 4.6, given below, show some properties (such as last period's factor exposures, time series of R2 values and factor returns) of the fitted factor model.

```
plot(fit.style.sector, which=2, f.sub=1:12, a.sub=1:10)
```

```
plot(fit.style.sector, which=4, las=2)
```

```
plot(fit.style.sector, which=12, f.sub=1:3, las=2, legend.loc="bottom", cex.legend=0.75)
```

Figure 4.7 and Figure 4.8 compares the kernel density of residuals for "MSFT" vs. normal and skew-t fits.

```
plot(fit.style.sector, plot.single=TRUE, which=10, asset.name="MSFT")
```

```
plot(fit.style.sector, plot.single=TRUE, which=11, asset.name="MSFT")
```

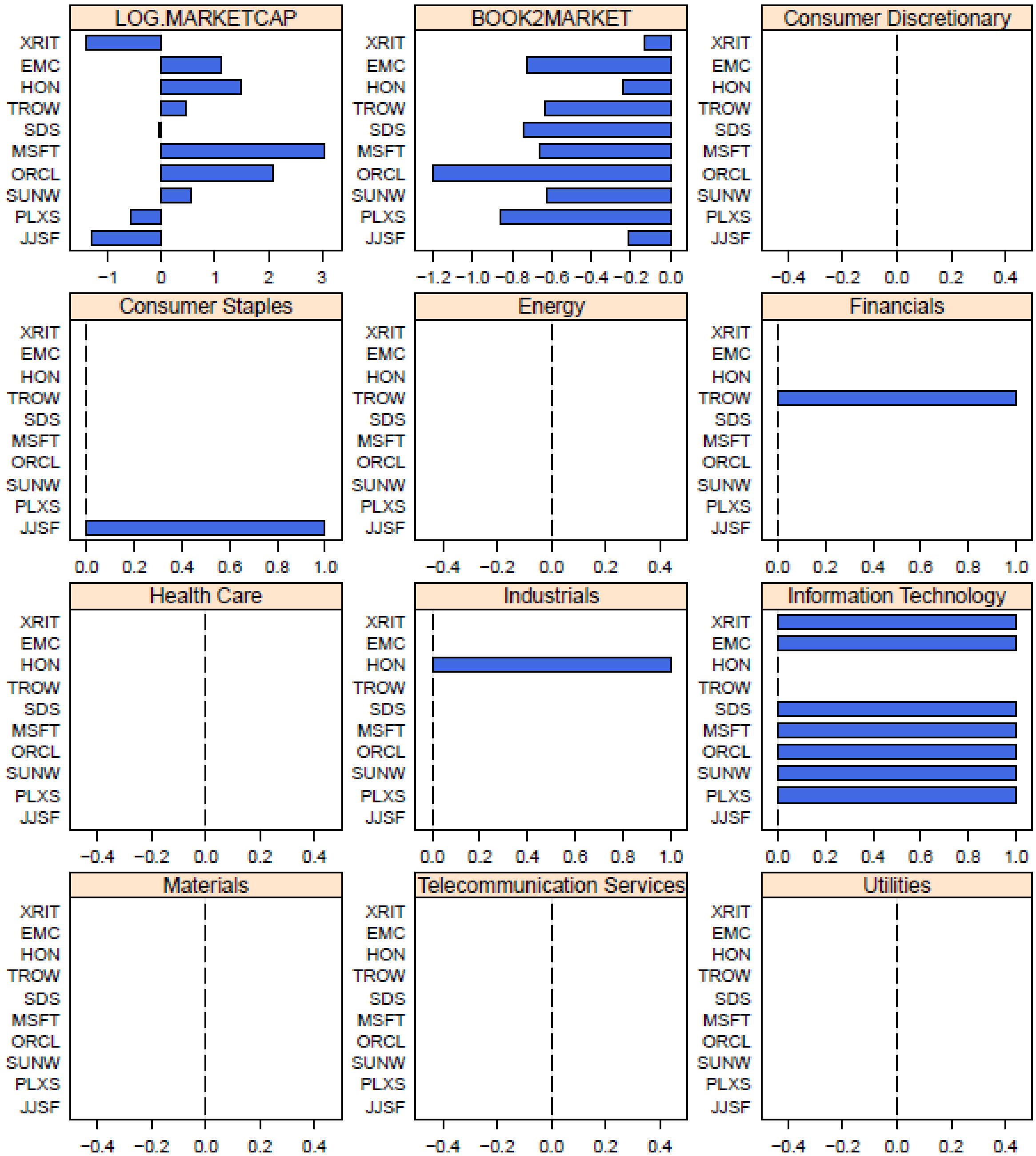

Figure 5.4: Factor exposures from the last period (1st 10 assets)

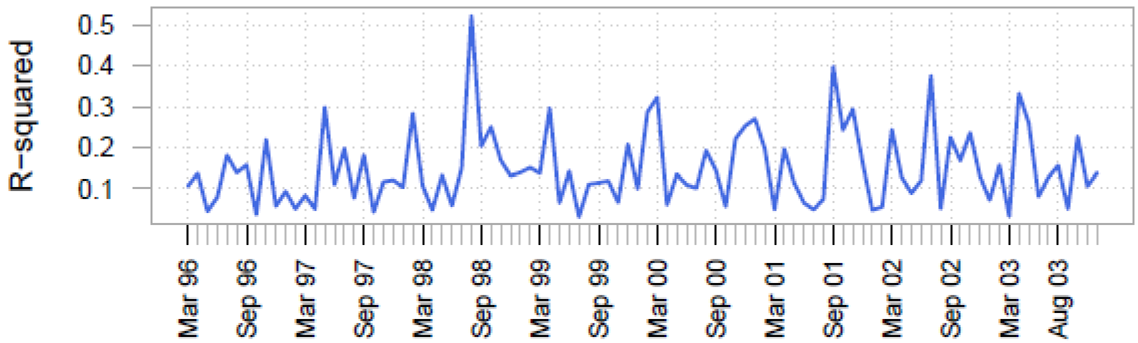

Figure 5.5: Time series of R^2 values

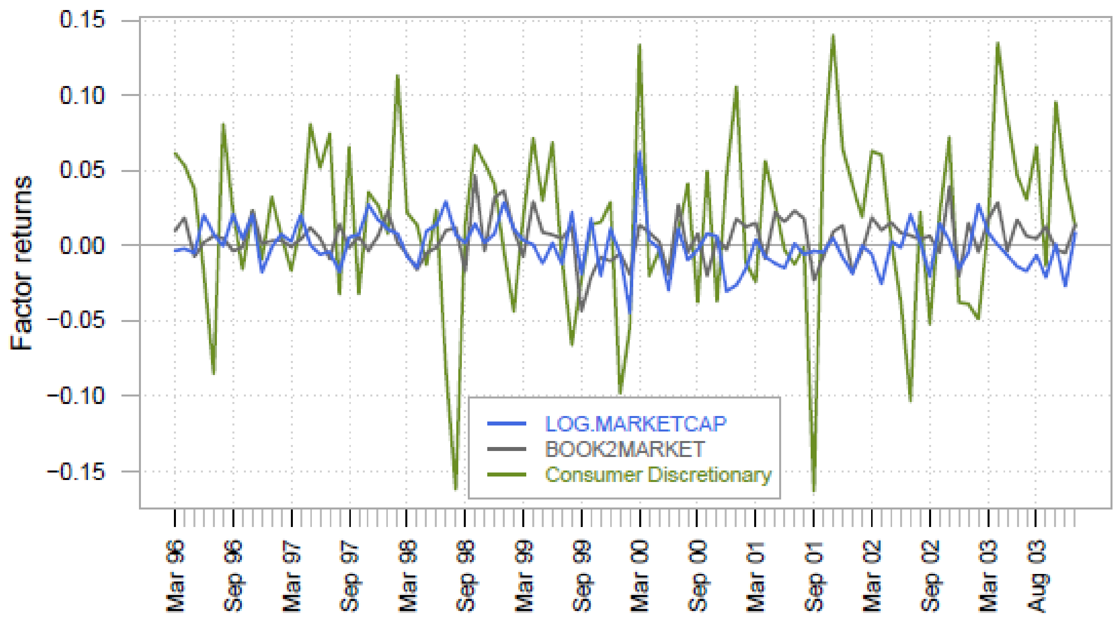

Figure 5.6: Time series of factor returns (displaying 1 sector and 2 style factors)

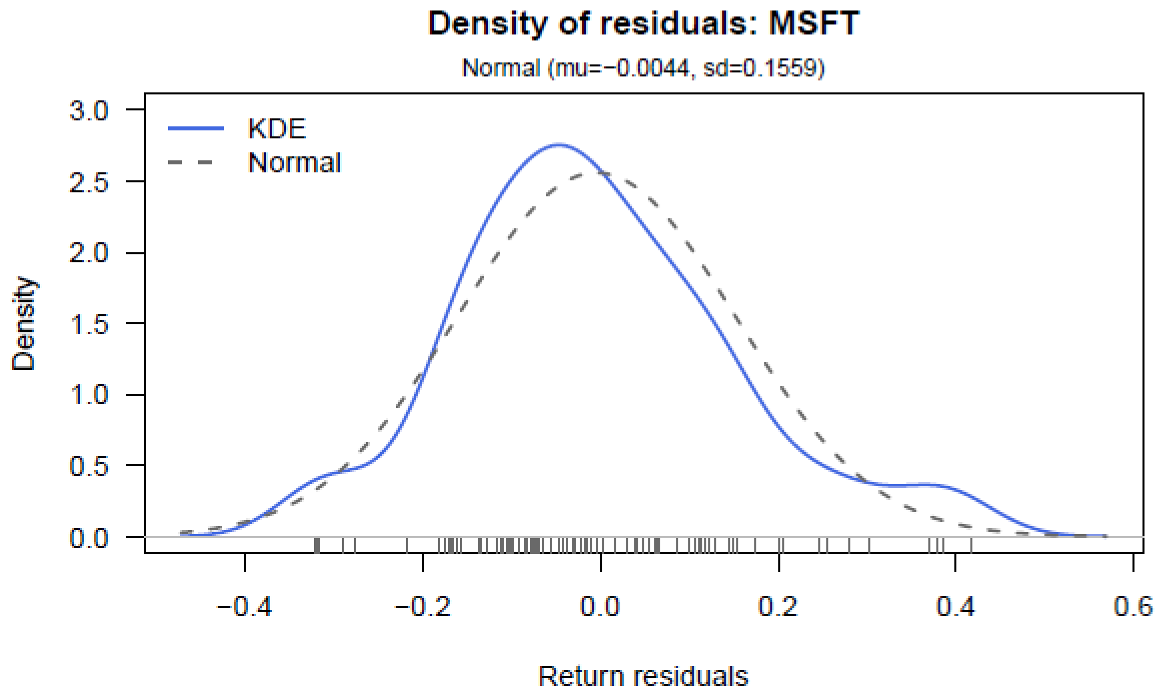

Figure 5.7: Non-parametric density of residuals with normal overlaid: MSFT

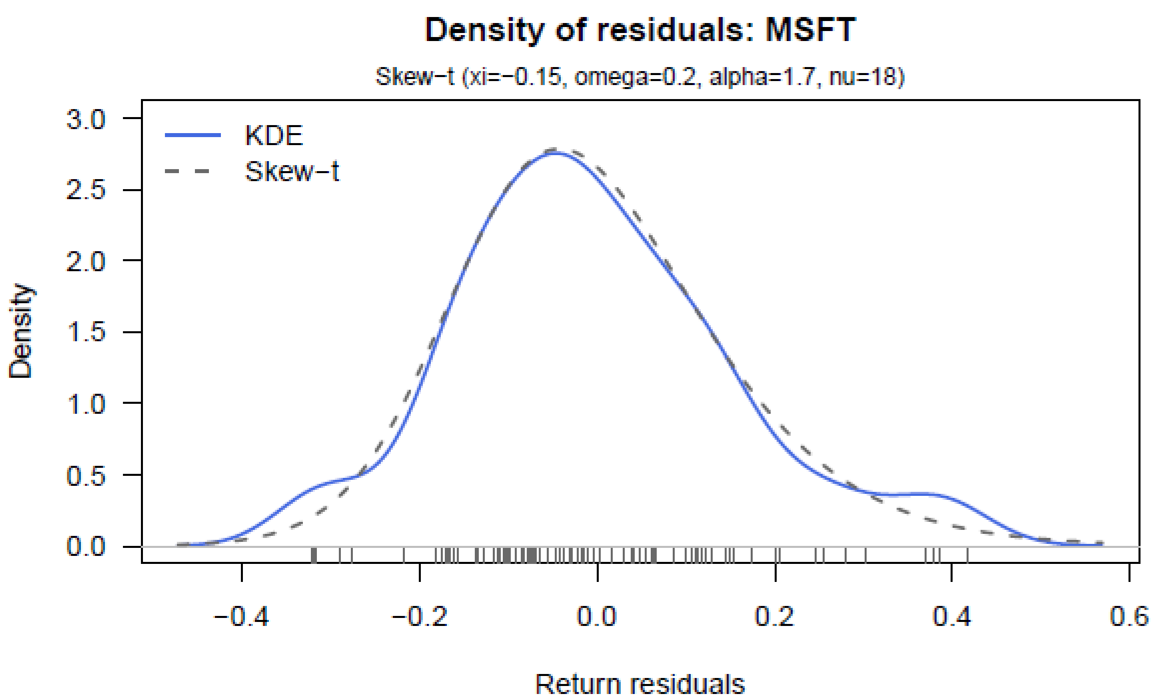

Figure 5.8: Non-parametric density of residuals with skew-t overlaid: MSFT

5.3 FACTOR MODEL COVARIANCE AND RISK DECOMPOSITION

5.3.1 *Factor Model Covariance*

Following Zivot & Jia-hui (2006), $R_{i,t}$, the return on asset i $(i = 1, 2, \ldots, N)$ at time t $(t = 1, 2, \ldots, T)$, is fitted with a factor model of the form,

$$R_{i,t} = \alpha_i + \boldsymbol{\beta}_i{}'\boldsymbol{f}_t + \varepsilon_{i,t} \tag{5.1}$$

where, α_i is the intercept, $\boldsymbol{f}_t$ is a $K \times 1$ vector of factor returns at time t, $\boldsymbol{\beta}_i$ is a $K \times 1$ vector of factor exposures for asset i and the error terms $\varepsilon_{i,t}$ are serially uncorrelated across time and contemporaneously uncorrelated across assets so that $\varepsilon_{i,t} \sim iid\ (0, \sigma_i^2)$. Thus, the variance of asset i's return is given by,

$$var(R_{i,t}) = \boldsymbol{\beta}_i{}'var\ (\boldsymbol{f}_t)\boldsymbol{\beta}_i + \sigma_i^2 \tag{5.2}$$

And the $N \times N$ covariance matrix of asset returns is,

$$var(\boldsymbol{R}) = \boldsymbol{\Omega} = \boldsymbol{B}\ var(\boldsymbol{F})\ \boldsymbol{B} + \boldsymbol{D} \tag{5.3}$$

where, $\boldsymbol{R}$ is the $N \times T$ matrix of asset returns, $\boldsymbol{B}$ is the $N \times K$ matrix of factor betas, $\boldsymbol{F}$ is the $K \times T$ matrix of factor returns and $\boldsymbol{D}$ is a diagonal matrix with σ_i^2 along the diagonal.

$fmCov$ computes the factor model covariance from a fitted factor model. The covariance of factor returns is the estimated covariance matrix. The factor exposures are observed values from the last period.

```
fmCov(fit.style.sector)[1:6,1:6]

##           JJSF      PLXS      SUNW     ORCL     MSFT       SDS
## JJSF  0.031186  0.003423 -0.004958 -0.01453 -0.02224 -0.000505
## PLXS  0.003423  0.068127  0.000755 -0.00183 -0.00491  0.002341
## SUNW -0.004958  0.000755  0.057259  0.01107  0.01489  0.002837
```

```
## ORCL -0.014532 -0.001825  0.011074  0.07685  0.03911  0.004194
## MSFT -0.022237 -0.004907  0.014887  0.03911  0.08100  0.004351
## SDS  -0.000505  0.002341  0.002837  0.00419  0.00435  0.030825

# factor model return correlation plot (for 1st 6 assets by default)
plot(fit.style.sector, which=8)
```

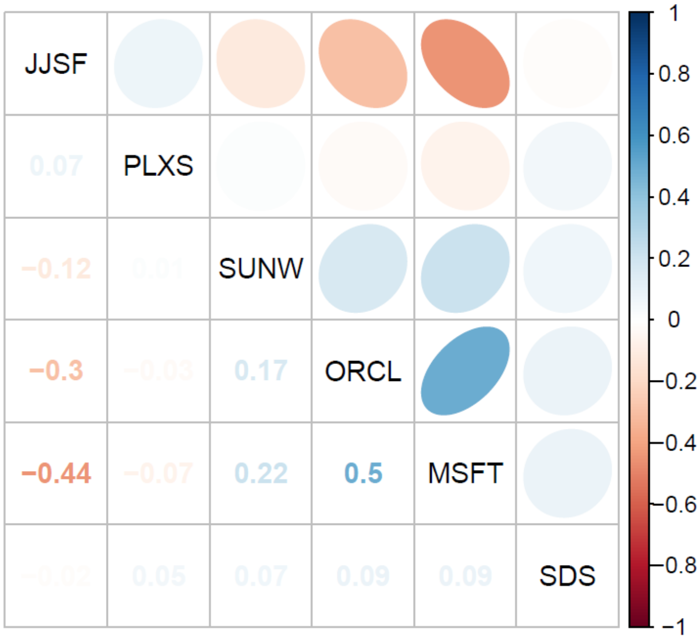

Figure 5.9: Factor model return correlation

5.3.2 *Standard Deviation Decomposition*

Following Meucci (2007), the standard deviation of asset i's return can be decomposed into the factor risk contributions using the factor model in equation 5.1 as shown below.

$$R_{i,t} = \boldsymbol{\beta}_i^{*\prime} \boldsymbol{f}_t^* \tag{5.4}$$

where, $\boldsymbol{\beta}_i^* = (\boldsymbol{\beta}_i' \ \sigma_i)$ and $\boldsymbol{f}_t^* = (\boldsymbol{f}_t' \ z_t)$, with $z_t \sim iid\ (0, 1)$ and σ_i is asset i's residual standard deviation. In other words, the residual is considered the $K + 1^{\text{th}}$ risk factor, where the exposure to the residual is the residual standard deviation, and the residual factor returns are assumed to be $iid \sim (0,1)$. By Euler's theorem, the standard deviation of asset i, σ_i, can be decomposed as:

$$\sigma_i = \sum_{k=1}^{K+1} cSd_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^* * mSd_{i,k}\right) \tag{5.5}$$

where, $cSd_{i,k}$ and $mSd_{i,k}$ are the component and marginal contributions to risk from the k^{th} factor. While the component contribution is the total contribution to risk from factor k, the marginal contribution to risk is the effect on the asset's standard deviation due to an incremental change in its exposure to the k^{th} factor, holding all else constant. Computing the component and marginal risk contributions is straight forward. Formulas are given below and details are in Meucci (2007).

$$\sigma_i = \sqrt{\boldsymbol{\beta}_i^{*\prime} cov(\boldsymbol{F}^*)\, \boldsymbol{\beta}_i^*} \tag{5.6}$$

$$\boldsymbol{mSd}_i = \frac{cov(\boldsymbol{F}^*)\, \boldsymbol{\beta}_i^*}{\sigma_i} \tag{5.7}$$

$$\boldsymbol{cSd}_i = \boldsymbol{\beta}_i^* \odot \boldsymbol{mSd}_i \tag{5.8}$$

The covariance term is approximated by the sample covariance and $\odot$ represents element-wise multiplication. $fmSdDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total standard deviation and component, marginal and percentage component contributions for each asset are returned.

```
decomp <- fmSdDecomp(fit.style.sector)
names(decomp)

## [1] "Sd.fm" "mSd"   "cSd"   "pcSd"

# get the factor model standard deviation for 1st 6 assets
decomp$Sd.fm[1:6]

##  JJSF  PLXS  SUNW  ORCL  MSFT   SDS
## 0.155 0.279 0.255 0.249 0.197 0.203
```

```r
# get the component contributions to Sd for (1st 6 assets, relevant factors)
decomp$cSd[1:6, c(1,2,4,9)]
```

```
##      LOG.MARKETCAP BOOK2MARKET Consumer Staples Information Technology
## JJSF      2.36e-03   -7.16e-05           0.0128                 0.0000
## PLXS      6.75e-04    1.88e-04           0.0000                 0.0469
## SUNW     -5.62e-05   -1.38e-04           0.0000                 0.0509
## ORCL      2.62e-03   -8.79e-05           0.0000                 0.0509
## MSFT      9.18e-03   -6.46e-04           0.0000                 0.0643
## SDS       3.99e-05    1.46e-05           0.0000                 0.0641
```

```r
# get the marginal factor contributions to Sd (1st 6 assets, relevant factors)
decomp$mSd[1:6, c(1,2,4,9)]
```

```
##      LOG.MARKETCAP BOOK2MARKET Consumer Staples Information Technology
## JJSF     -0.001820    3.32e-04          0.01282                0.00796
## PLXS     -0.001165   -2.19e-04          0.00328                0.04695
## SUNW     -0.000099    2.21e-04          0.00399                0.05090
## ORCL      0.001265    7.30e-05          0.00401                0.05093
## MSFT      0.003006    9.74e-04          0.00583                0.06433
## SDS      -0.000907   -1.97e-05          0.00475                0.06414
```

```r
# get the % component contributions to Sd (1st 6 assets, relevant factors)
decomp$pcSd[1:6, c(1,2,4,9)]
```

```
##      LOG.MARKETCAP BOOK2MARKET Consumer Staples Information Technology
## JJSF        1.5283    -0.04632              8.3                    0.0
## PLXS        0.2423     0.06760              0.0                   16.8
## SUNW       -0.0221    -0.05418              0.0                   20.0
## ORCL        1.0508    -0.03531              0.0                   20.5
## MSFT        4.6697    -0.32874              0.0                   32.7
## SDS         0.0196     0.00718              0.0                   31.6
```

```r
# plot the % component contributions to Sd (1st 6 assets, relevant factors)
plot(fit.style.sector, which=9, f.sub=c(1,2,4,9))
```

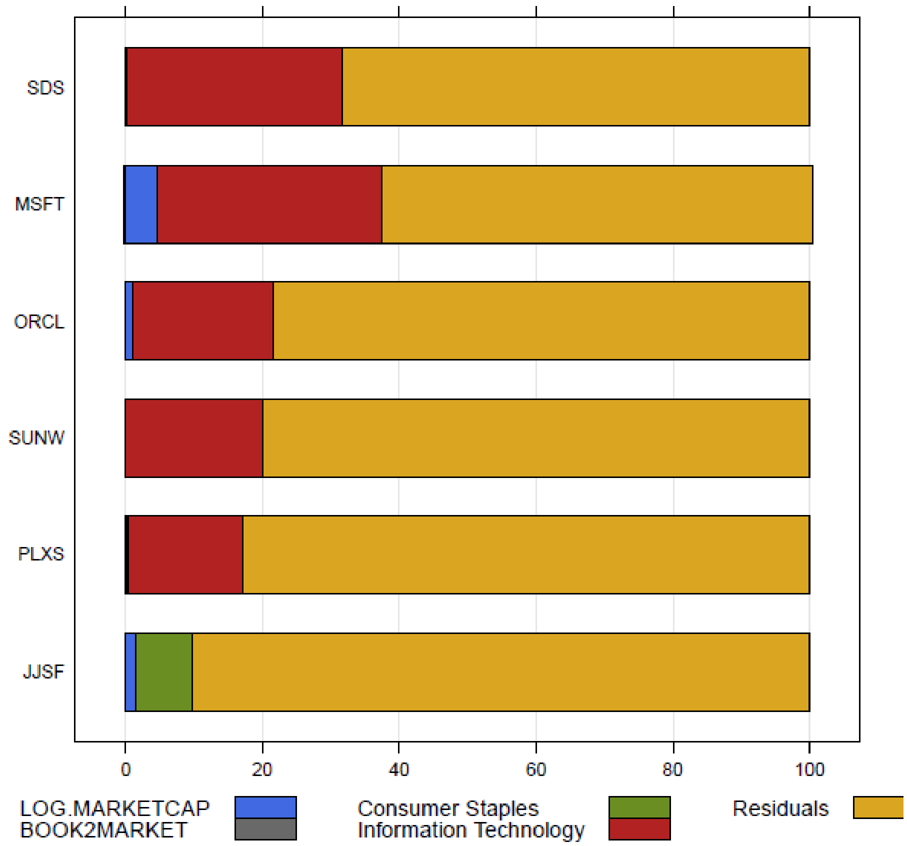

Figure 5.10: Percentage factor contribution to SD

5.3.3 *Value-at-Risk Decomposition*

Euler decomposition of return standard deviation shown above can also be applied to other risk measures such as value-at-risk (VaR) and expected shortfall (ES). The VaR version of equation 5.5 is given below. The value-at-risk of asset i can be decomposed as:

$$VaR_i = \sum_{k=1}^{K+1} cVaR_{i,k} = \sum_{k=1}^{K+1} \left(\beta_{i,k}^* * mVaR_{i,k}\right) \tag{5.9}$$

The marginal contribution to VaR_i is defined as the expectation of $\boldsymbol{F}^*$, conditional on the loss being equal to VaR_i. This is approximated as described in Epperlein and Smillie (2006) using a triangular smoothing kernel. $type$ gives the option to estimate VaR_i non-parametrically using the sample quantile (default) or assuming a normal distribution.

$fmVaRDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total VaR and component, marginal and percentage component contributions for each asset are returned.

```
decomp1 <- fmVaRDecomp(fit.style.sector, type="normal", p=0.10)
names(decomp1)

## [1] "VaR.fm"     "n.exceed"   "idx.exceed" "mVaR"       "cVaR"
## [6] "pcVaR"

# get the factor model value-at-risk for 1st 6 assets
decomp1$VaR.fm[1:6]

##   JJSF   PLXS   SUNW   ORCL   MSFT    SDS
## -0.190 -0.335 -0.304 -0.299 -0.229 -0.238

# print the number of VaR exceedences for 1st 6 assets
decomp1$n.exceed[1:6]

## JJSF PLXS SUNW ORCL MSFT  SDS
##   11    6   12    9    7    6

# plot the % component contributions to VaR (1st 6 assets, relevant factors)
plot(fit.style.sector, which=11, f.sub=c(1,2,4,9))
```

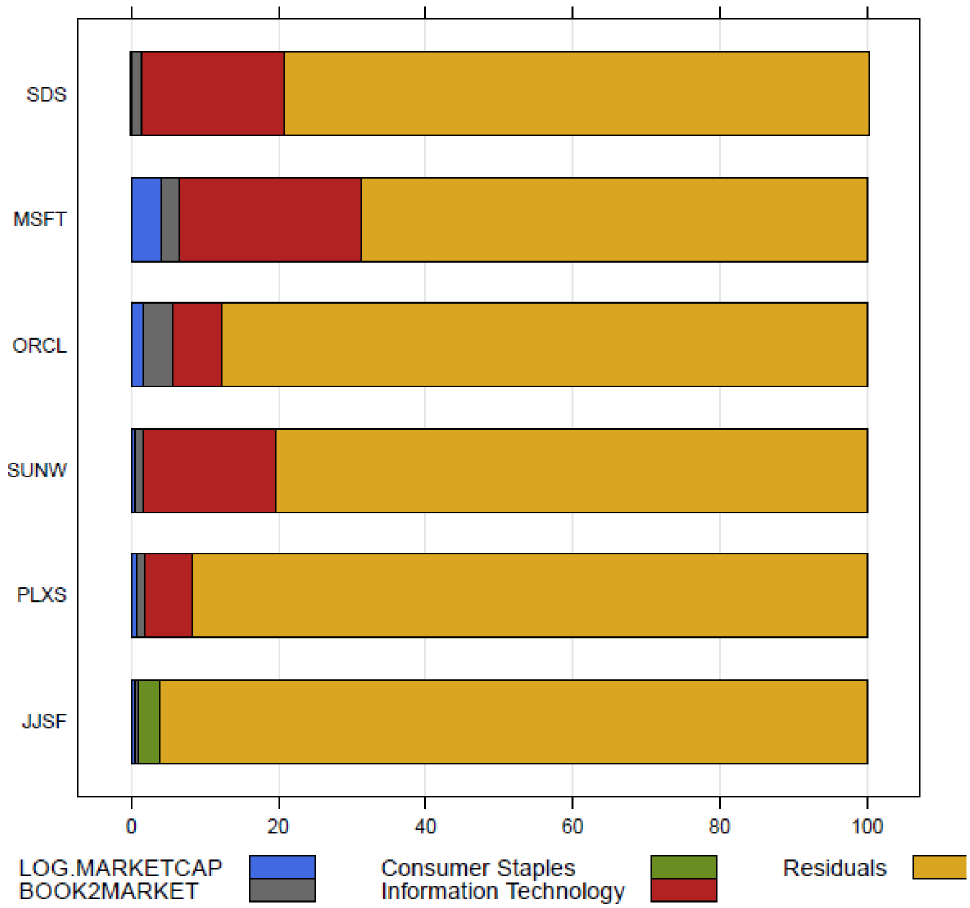

Figure 5.11: Percentage factor contribution to VaR

5.3.4 *Expected Shortfall Decomposition*

The expected shortfall (ES) version of equation 5.5 is given below. The expected shortfall of asset i can be decomposed as:

$$ES_i = \sum_{k=1}^{K+1} cES_{i,k} = \sum_{k=1}^{K+1} \left(\beta^*_{i,k} * mES_{i,k}\right) \tag{5.10}$$

The marginal contribution to ES_i is defined as the expectation of $\boldsymbol{F}^*$, conditional on the loss being less than or equal to ES_i. This is estimated as a sample average of the observations in that data window. Once again, $type$ gives the option to estimate ES_i non-parametrically using the sample quantile (default) or assuming a normal distribution.

$fmESDecomp$ performs this decomposition for all assets in the given factor model fit object as shown below. The total ES and component, marginal and percentage component contributions for each asset are returned.

```
decomp2 <- fmEsDecomp(fit.style.sector, type="normal")
names(decomp2)

## [1] "ES.fm" "mES"   "cES"   "pcES"

# get the factor model expected shortfall for 1st 6 assets
decomp2$ES.fm[1:6]

##   JJSF   PLXS   SUNW   ORCL   MSFT    SDS
## -0.327 -0.597 -0.549 -0.534 -0.428 -0.442

# get the component contributions to ES for (1st 6 assets, relevant factors)
decomp2$cES[1:6, c(1,2,4,9)]

##       LOG.MARKETCAP BOOK2MARKET Consumer Staples Information Technology
## JJSF      -4.87e-03     0.00115          -0.0357                  0.000
## PLXS      -1.39e-03     0.00364           0.0000                 -0.122
## SUNW       1.15e-04     0.00351           0.0000                 -0.133
## ORCL      -5.41e-03     0.00568           0.0000                 -0.130
## MSFT      -1.91e-02     0.00409           0.0000                 -0.156
## SDS       -8.35e-05     0.00255           0.0000                 -0.153

# plot the % component contributions to ES (1st 6 assets, relevant factors)
plot(fit.style.sector, which=10, f.sub=c(1,2,4,9))
```

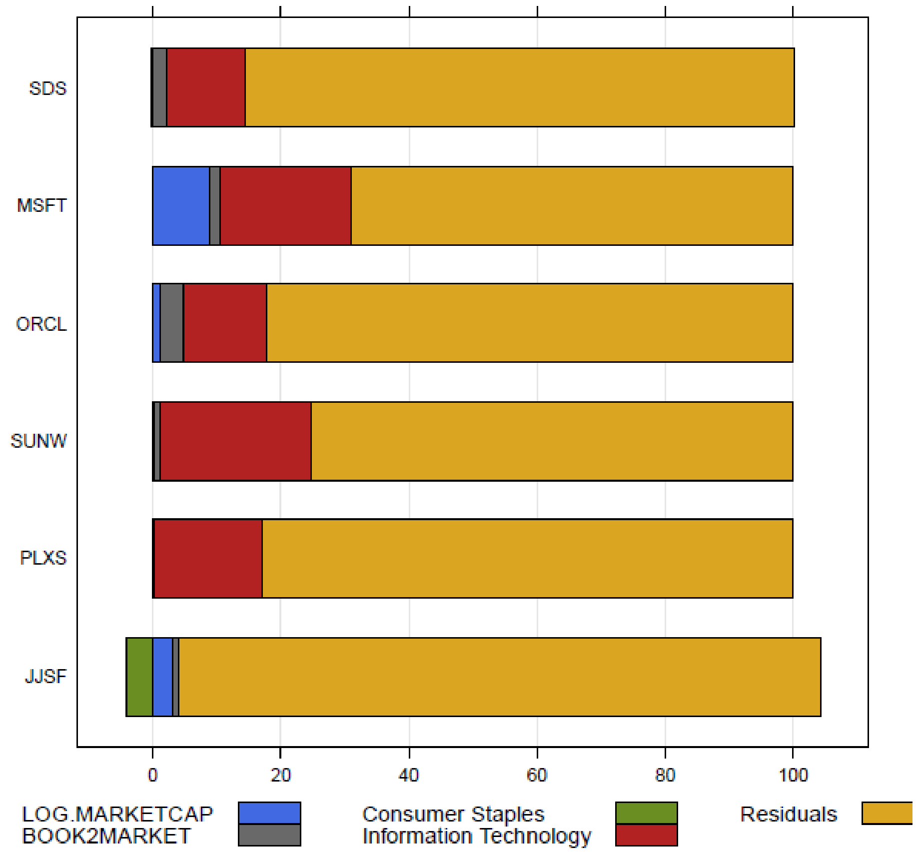

Figure 5.12: Percentage factor contribution to ES

5.4 Plot

Many types of plots for ffm objects have already been demonstrated. Let's look at all available arguments for plotting a ffm object.

```
## S3 method for class "ffm"
plot (x, which=NULL, f.sub=1:2, a.sub=1:6, plot.single=FALSE, asset.name,
      colorset=c("royalblue","dimgray","olivedrab","firebrick", "goldenrod",
                 "mediumorchid","deepskyblue","chocolate","darkslategray"),
      legend.loc="topleft", las=1, lwd=2, maxlag=15, ...)
```

5.4.1 *Group Plots*

This is the default option for plotting. Simply running $plot\ (fit)$, where fit is any ffm object, will bring up the following menu for group plots.

```
plot(fit.sector)


# Make a plot selection (or 0 to exit):
#
#  1: Distribution of factor returns
#  2: Factor exposures from the last period
#  3: Actual and Fitted asset returns
#  4: Time-series of R-squared values
#  5: Residual variance across assets
#  6: Scatterplot matrix of residuals, with histograms, density overlays,
#     correlations and significance stars
#  7: Factor Model Residual Correlation
#  8: Factor Model Return Correlation
#  9: Factor Contribution to SD
# 10: Factor Contribution to ES
# 11: Factor Contribution to VaR
# 12: Time series of factor returns
#
# Selection:
```

Note: Only a subset of assets and factors selected by $a.sub$ and $f.sub$ are plotted. The 1st 2 factors (or just the solitary factor for a single factor model) and the 1st 6 assets are shown by default.

```
# Examples of group plots: looping disabled & no. of assets displayed = 4.
plot(fit.style.sector, which=3, a.sub=1:3, legend.loc=NULL, lwd=1)
```

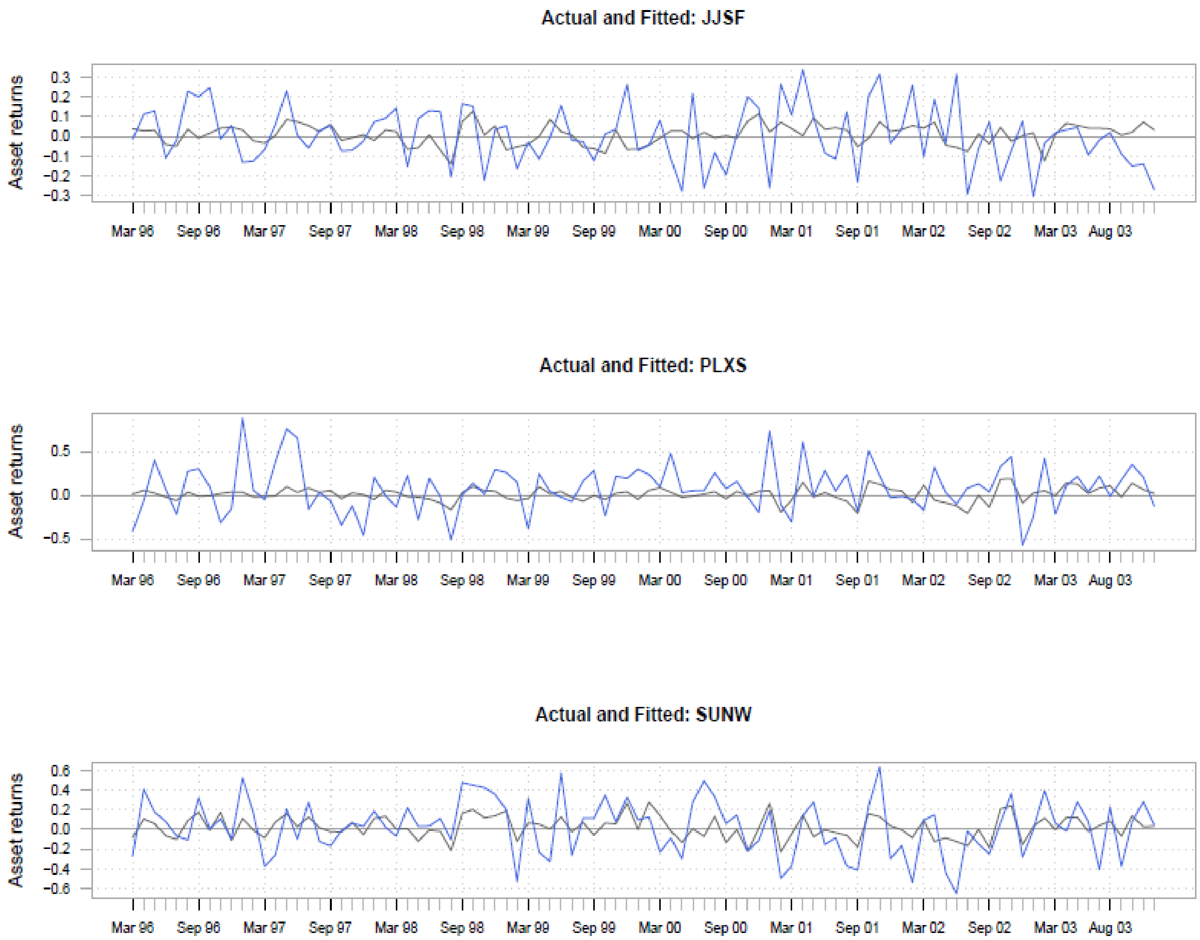

Figure 5.13: Actual (blue) and fitted (grey) factor model returns for the 1st 3 assets

5.4.2 *Menu and Looping*

If the plot type argument $which$ is not specified, a menu prompts for user input. In case multiple plots are needed, the menu is repeated after each plot (enter 0 to exit). User can also input a numeric vector of plot options via argument $which$.

5.4.3 *Individual Plots*

Setting $plot.single = TRUE$ enables individual asset plots. If there is more than one asset fit by the fitted object x, $asset.name$ is also necessary. In case the ffm object x contains only a single asset's fit, $plot.ffm$ can infer $asset.name$ without user input. Here's the individual plot menu.

```
plot(fit.style.sector, plot.single=TRUE, asset.name="MSFT")


# Make a plot selection (or 0 to exit):
#
#  1: Actual and fitted asset returns
#  2: Actual vs. fitted asset returns
#  3: Residuals vs. fitted asset returns
#  4: Residuals with standard error bands
#  5: Time series of squared residuals
#  6: Time series of absolute residuals
#  7: SACF and PACF of residuals
#  8: SACF and PACF of squared residuals
#  9: SACF and PACF of absolute residuals
# 10: Non-parametric density of residuals with normal overlaid
# 11: Non-parametric density of residuals with skew-t overlaid
# 12: Histogram of residuals with non-parametric density and normal overlaid
# 13: QQ-plot of residuals
#
# Selection:
```

Here are some examples which don't need interactive user input.

```
plot(fit.style.sector, plot.single=TRUE, asset.name="MSFT", which=4)
```

```
plot(fit.style.sector, plot.single=TRUE, asset.name="MSFT", which=9)
```

```
plot(fit.style.sector, plot.single=TRUE, asset.name="MSFT", which=13)
grid()
```

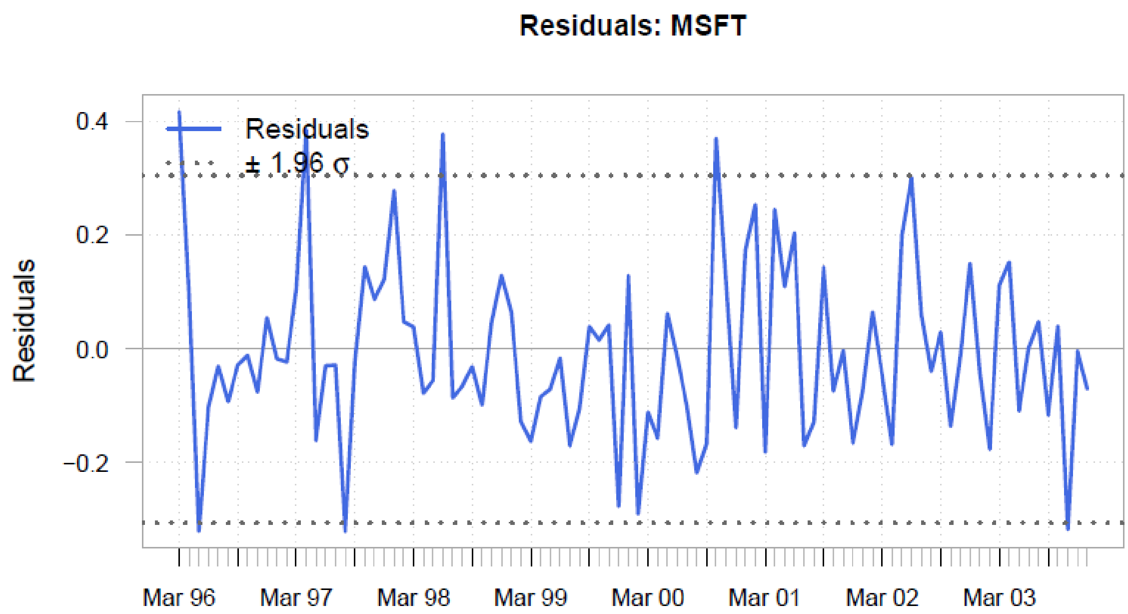

Figure 5.14: Time series plot of residuals with standard error bands: MSFT

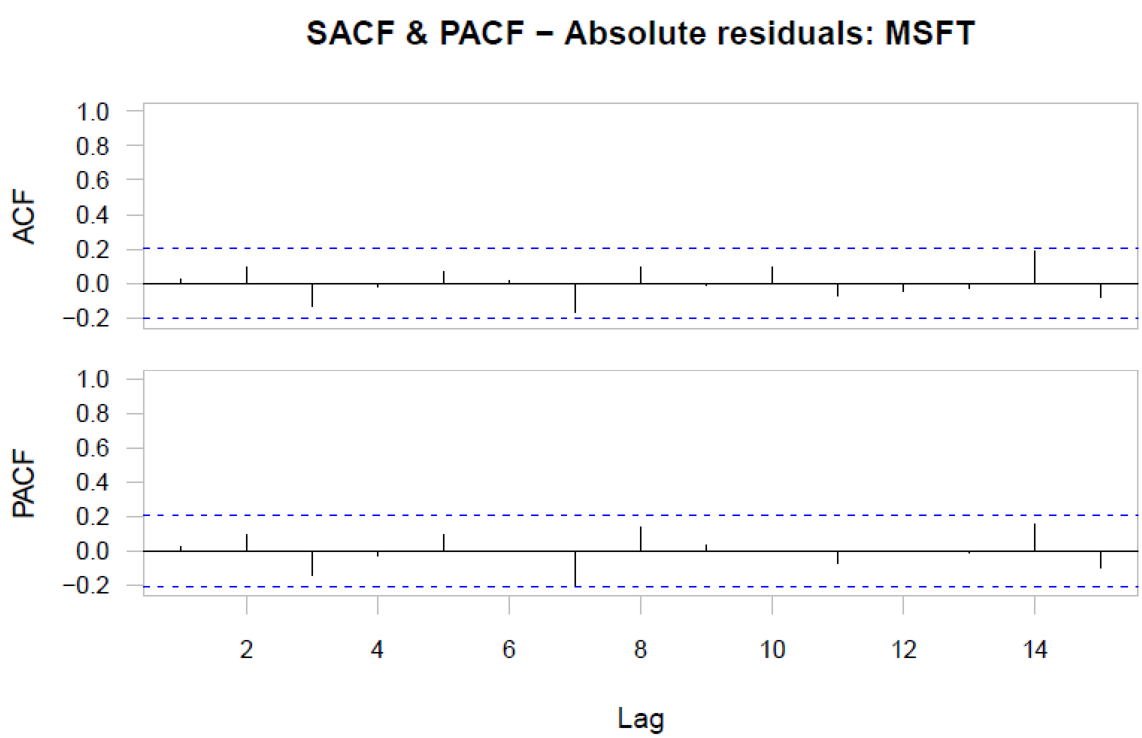

Figure 5.15: SACF and PACF of absolute residuals: MSFT

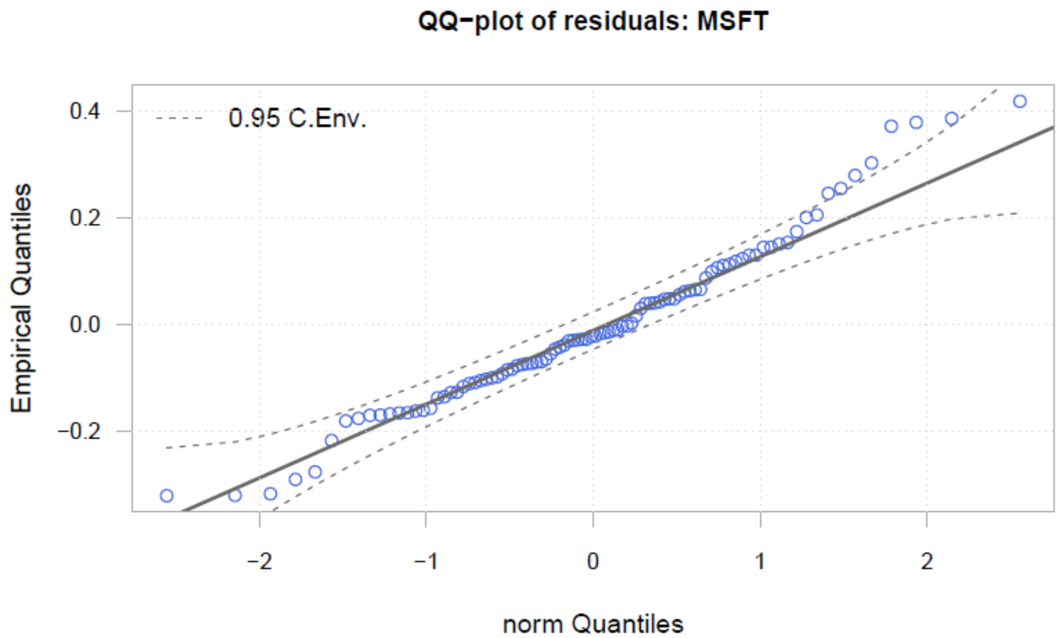

Figure 5.16: QQ-plot of residuals: MSFT

BIBLIOGRAPHY

[1] Bai, J., & Ng, S. (2002). Determining the number of factors in approximate factor models. Econometrica, 70(1), 191-221.

[2] Bansal, R., Hsieh, D. A., & Viswanathan, S. (1993). A new approach to international arbitrage pricing. The Journal of Finance, 48(5), 1719-1747.

[3] Bansal, R., & Viswanathan, S. (1993). No arbitrage and arbitrage pricing: A new approach. The Journal of Finance, 48(4), 1231-1262.

[4] Bender, J., Briand, R., Nielsen, F., & Stefek, D. (2010). Portfolio of risk premia: A new approach to diversification. The Journal of Portfolio Management, 36(2), 17-25.

[5] Berndt, E. R. (1991). The practice of econometrics: classic and contemporary. Addison Wesley Publishing Company.

[6] Boudt, K. M. R., Darras, J., & Peeters, B. (2013). Dynamic risk based asset allocation. Wilmott Magazine, 2013(67), 62-65.

[7] Brown, M., & Canova, C. (2011). Using multiple risk models for superior portfolio management… A practice not just for quants. Axioma Research Paper, No. 032.

[8] Brown, M. (2014). Axioma's Macroeconomic Model: Insights into equity portfolios from a new perspective. Axioma Research Paper, No. 048.

[9] Bruder, B., & Roncalli, T. (2012). Managing risk exposures using the risk budgeting approach. SSRN, working paper.

[10] Carhart, M. M. (1997). On persistence in mutual fund performance. The Journal of finance, 52(1), 57-82.

[11] Chen, N. F., Roll, R., & Ross, S. A. (1986). Economic forces and the stock market. Journal of business, 383-403.

[12] Connor, G. (1995). The three types of factor models: A comparison of their explanatory power. Financial Analysts Journal, 51(3), 42-46.

[13] Conner, G., & Korajczyk, R. (1986). Performance measurement with the arbitrage pricing theory. Journal of Financial Economics, 15, 373-394.

[14] Connor, G., & Korajczyk, R. A. (1993). A test for the number of factors in an approximate factor model. the Journal of Finance, 48(4), 1263-1291.

[15] Deguest, R., Martellini, L., & Meucci, A. (2013). Risk parity and beyond-from asset allocation to risk allocation decisions. SSRN, working paper.

[16] Dittmar, R. F. (2002). Nonlinear pricing kernels, kurtosis preference, and evidence from the cross section of equity returns. The Journal of Finance, 57(1), 369-403.

[17] Epperlein, E., & Smillie, A. (2006). Portfolio risk analysis Cracking VAR with kernels. Rish-London-Risk Magazine Limited-, 19(8), 70.

[18] Faff, R. W., Hillier, D., & Hillier, J. (2000). Time varying beta risk: An analysis of alternative modelling techniques. Journal of Business Finance & Accounting, 27(5-6), 523-554.

[19] Fama, E. F., & French, K. R. (1992). The cross-section of expected stock returns. the Journal of Finance, 47(2), 427-465.

[20] Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. Journal of financial economics, 33(1), 3-56.

[21] Ghysels, E. (1998). On Stable Factor Structures in the Pricing of Risk: Do Time-Varying Betas Help or Hurt? The Journal of Finance, 53(2), 549-573.

[22] Grinold, R. C., & Kahn, R. N. (2000). Active Portfolio Management: A Quantitative Approach for Producing Superior Returns and Controlling Risk. McGraw-Hill, New York.

[23] Henriksson, R. D., & Merton, R. C. (1981). On market timing and investment performance. II. Statistical procedures for evaluating forecasting skills. Journal of business, 513-533.

[24] Herskovic, B., Kelly, B., Lustig, H., & Van Nieuwerburgh, S. (2016). The common factor in idiosyncratic volatility: Quantitative asset pricing implications. Journal of Financial Economics, 119(2), 249-283.

[25] Heston, S. L., & Rouwenhorst, K. G. (1995). Industry and country effects in international stock returns. Journal of Portfolio Management, 21, 53-53.

[26] Inker, B. (2010). The hidden risks of risk parity portfolios. GMO white paper.

[27] Jones, C. I. (2016). The facts of economic growth. In Handbook of macroeconomics (Vol. 2, pp. 3-69). Elsevier.

[28] Lintner, J. (1965). Security prices, risk, and maximal gains from diversification. The journal of finance, 20(4), 587-615.

[29] Litterman, R., & Scheinkman, J. (1991). Common factors affecting bond returns. Journal of fixed income, 1(1), 54-61.

[30] Lohre, H., Opfer, H., & Ország, G. (2014). Diversifying risk parity. The Journal of Risk, 16(5), 53.

[31] Maillard, S., Roncalli, T., & Teïletche, J. (2010). The Properties of Equally Weighted Risk Contribution Portfolios. Journal of Portfolio Management, 36(4), 60.

[32] Maio, P., & Philip, D. (2015). Macro variables and the components of stock returns. Journal of Empirical Finance, 33, 287-308.

[33] Markowitz, H. (1952). Portfolio selection. The journal of finance, 7(1), 77-91.

[34] Menchero, J., & Mitra, I. (2008). The structure of hybrid factor models. Journal of Investment Management, 6(3), 35-47.

[35] Menchero, J., & Davis, B. (2010). The characteristics of factor portfolios. Journal of Performance Measurement, 15(1), 52-62.

[36] Menchero, J., Morozov, A., & Pasqua, A. (2013). Predicting Risk at Short Horizons. MSCI Analytics Research.

[37] Meucci, A. (2007). Risk contributions from generic user-defined factors. The Risk Magazine, 84-88.

[38] Meucci, A. (2009). Risk and asset allocation. Springer Science & Business Media.

[39] Miller, G. (2006). Needles, haystacks, and hidden factors. Journal of Portfolio Management, 32(2), 25.

[40] Page, S., & Taborsky, M. A. (2011). The myth of diversification: risk factors versus asset classes. Journal of Portfolio Management, 37(4), 1.

[41] Qian, E. (2005). Risk parity portfolios: Efficient portfolios through true diversification. Panagora Asset Management.

[42] Qian, E. (2011). Risk parity and diversification. Journal of Investing, 20(1), 119.

[43] Raffinot, T. (2014). A simple and efficient macroeconomic framework for asset allocation. SSRN Electronic Journal, Working paper.

[44] Roncalli, T., & Weisang, G. (2012). Risk parity portfolios with risk factors. SSRN, working paper.

[45] Ruppert, D. (2011). Statistics and data analysis for financial engineering (Vol. 13). New York: Springer.

[46] Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. The journal of finance, 19(3), 425-442.

[47] Sharpe, W. F. (1970). Portfolio theory and capital markets. McGraw-Hill College.

[48] Shepard, P. (2009). Second order risk. Working paper, MSCI Barra.

[49] Shepard, P. (2011). The Barra Integrated Model (BIM301). MSCI Model Insights.

[50] Swinkels, L., & Van Der Sluis, P. J. (2006). Return-based style analysis with time-varying exposures. The European Journal of Finance, 12(6-7), 529-552.

[51] Zivot, E., & Wang, J. (2007). Modeling financial time series with S-Plus® (Vol. 191). Springer Science & Business Media.

APPENDIX A: R CODE FOR CHAPTER 1

```
library(factorAnalytics)
library(corrplot)


###########################
load("~/Documents/Disssertation/fmComparison.RData")
T <- nrow(exretM.xts)
N <- ncol(exretM.xts)
col1=1:10
col2=c("royalblue","dimgray","olivedrab","firebrick","goldenrod",
       "mediumorchid","deepskyblue","chocolate","darkslategray")

models <- c("Sample covariance","FFC4","FFC4.rob","FFC4.lasso","SFM4",
            "SFM5","FFM","FFM4+sect","FFC4+stat","FFM+sect+stat",
            "FFC4+ffm")

############################

# function to estimate weights, exp. return and risk for globalMinVar
strategy
MinVar <- function(omega, mu) {
  y <- solve(omega, rep(1,nrow(omega)))
  w <- y/sum(y)
  mu <- w %*% mu
  sig <- sqrt(1/sum(y))
  list(sig=sig, mu=mu, w=w)
}

############################

# Model 1: Sample Covariance

# full sample statistics
Omega <- cov(exretM.xts) # return covariance
Mu <- apply(exretM.xts, 2, mean) # expected return
sig2 <- diag(Omega) # return variances

##########################

# Model 2/3/4: Fama-French-Carhart 4 factor model

# fit time series factor model with FFC 4 factors
FFC4 <- fitTsfm(asset.names=colnames(exretM.xts),
factor.names=colnames(ff.xts)[-1],
                data=merge(exretM.xts, ff.xts))

# use robust least squares fit method
FFC4.rob <- fitTsfm(asset.names=colnames(exretM.xts),
factor.names=colnames(ff.xts)[-1],
```

```
                    data=merge(exretM.xts, ff.xts),
fit.method="Robust")

# use lasso subset selection with Cp criterion
FFC4.lasso <- fitTsfm(asset.names=colnames(exretM.xts),
factor.names=colnames(ff.xts)[-1],
                      data=merge(exretM.xts, ff.xts),
variable.selection="lars")

##########################

# Model 5/6: Statistical factor model

SFM4 <- fitSfm(exretM.xts, k=4)
SFM5 <- fitSfm(exretM.xts, k=5)

##########################

# Model 7, 8: FFM and FFM.sect

FFM4 <- fitFfm(data=capIQ200.cs, asset.var="TICKER", ret.var="ExRET",
               date.var="Date", fit.method="WLS", lag.exposures =
FALSE,
               exposure.vars=c("BP","AnnVol1M","PM12M1M","LogMktCap"))

FFM4.sect <- fitFfm(data=capIQ200.cs, asset.var="TICKER",
ret.var="ExRET",
                    date.var="Date", fit.method="WLS", lag.exposures =
FALSE,

exposure.vars=c("BP","AnnVol1M","PM12M1M","LogMktCap","SECTOR"))

##########################

# Model 9: Stat factor extracted from FFC4 residuals

FFC4.stat <- fitSfm(residuals(FFC4), k=5)

##########################

# Model 10: Stat factor extracted from FFM4 residuals

FFM4.sect.stat <- fitSfm(FFM4.sect$residuals, k=5)

##########################

# Model 11: FFM to residuals from FFC4

newdata <- capIQ200.cs
FFC4.resid <- residuals(FFC4)
count.row <- 1
```

```
for (t in unique(capIQ200.cs[,"Date"])) {
  for (i in unique(capIQ200.cs[,"TICKER"])) {
    newdata[count.row,"ExRET"] <- FFC4.resid[t,i]
    count.row <- count.row + 1
  }
}

FFC4.ffm <- fitFfm(newdata, asset.var="TICKER", ret.var="ExRET",
                    date.var="Date", fit.method="WLS", lag.exposures =
FALSE,

exposure.vars=c("BP","AnnVol1M","PM12M1M","LogMktCap","SECTOR"))


############################

# In-sample performance

############################

# Table1: Explanatory power as in Connor (1995)

Table1.df <- round(as.data.frame(
100*c(1 - mean((FFC4$resid.sd)^2)/mean(sig2),

1 - mean((FFC4.rob$resid.sd)^2)/mean(sig2),

1 - mean((FFC4.lasso$resid.sd)^2)/mean(sig2),

1 - mean((SFM4$resid.sd)^2)/mean(sig2),

1 - mean((SFM5$resid.sd)^2)/mean(sig2),

1 - mean(diag(FFM4$resid.cov))/mean(sig2),

1 - mean(diag(FFM4.sect$resid.cov))/mean(sig2),

1 - mean((FFC4.stat$resid.sd)^2)/mean(sig2),

1 - mean((FFM4.sect.stat$resid.sd)^2)/mean(sig2),

1 - mean(diag(FFC4.ffm$resid.cov))/mean(sig2)
                                 )),2)

Table1.df <- cbind(models[-1], Table1.df)
colnames(Table1.df) <- c("Model","Explanatory power")
write.table(Table1.df, "fmComparison/Output/Table1", quote=FALSE,
row.names=FALSE)

###########################

# Graph2: distribution of R-squared across assets
```

```
pdf(file = "fmComparison/Output/Graph2a.pdf")
d1 <- density(FFC4$r2)
d2 <- density(FFC4.rob$r2)
d3 <- density(FFC4.lasso$r2)
d4 <- density(SFM4$r2)
d5 <- density(SFM5$r2)
ymax <- ceiling(max(0,d1$y,d2$y,d3$y,d4$y,d5$y))
plot(d1, main="Distribution of R-squared across assets", col=col2[1],
lwd=2,
     ylim=c(0,ymax), xlim=c(-0.1,1), xlab=paste("R-squared
(N=",N,")",sep=""))
lines(d2, col=col2[2], lwd=2)
lines(d3, col=col2[3], lwd=2)
lines(d4, col=col2[4], lwd=2)
lines(d5, col=col2[5], lwd=2)
legend("topright", legend=models[2:6], col=col2[1:5], lwd=2)
dev.off()

pdf(file = "fmComparison/Output/Graph2b.pdf")
d6 <- density(FFM4$r2)
d7 <- density(FFM4.sect$r2)
ymax <- ceiling(max(0,d6$y,d7$y))
plot(d6, main="Distribution of R-squared across time", col=col2[1],
lwd=2,
     ylim=c(0,ymax), xlim=c(-0.2,1), xlab=paste("R-squared
(T=",T,")",sep=""))
lines(d7, col=col2[2], lwd=2)
legend("topright", legend=models[7:8], col=col2[1:2], lwd=2)
dev.off()

pdf(file = "fmComparison/Output/Graph2c.pdf")
d8 <- density(FFC4.stat$r2)
d9 <- density(FFM4.sect.stat$r2)
d10 <- density(FFC4.ffm$r2)
ymax <- ceiling(max(0,d8$y,d9$y,d10$y))
plot(d8, main="Distribution of R-squared for secondary models",
col=col2[1], lwd=2,
     ylim=c(0,ymax), xlim=c(-0.1,1), xlab=paste("R-squared (N=",N,",
T=",T,")",sep=""))
lines(d9, col=col2[3], lwd=2)
lines(d10, col=col2[4], lwd=2)
legend("topright", legend=models[9:11], col=col2[c(1,3,4)], lwd=2)
dev.off()

#########################

# Graph3: Residual correlation structure for FFC4
pdf(file = "fmComparison/Output/Graph3-FFC4.pdf", width=9, height=9)
corrplot(cor(residuals(FFC4)), method="color", order="hclust",
addrect=10,
         hclust.method="ward", tl.cex=0.4, cl.cex=0.6)
dev.off()
```

```
pdf(file="fmComparison/Output/Graph3-FFC4.rob.pdf", width=9, height=9)
corrplot(cor(residuals(FFC4.rob)), method="color", order="hclust",
addrect=10,
         hclust.method="ward", tl.cex=0.3)
dev.off()

# print ordered stocklist based on GSECTOR
write.table(stocklist[with(stocklist, order(GSECTOR)),c(2,4,5,7,8)],
            "fmComparison/Output/Table-stocklist", row.names=FALSE)

#########################

# Graph4: Example of rolling regression coefficients
pdf(file = "fmComparison/Output/Graph4-PG.pdf")
plot(FFC4, plot.single=TRUE, asset.name="PG", which=18, xlab="")
dev.off()
pdf(file = "fmComparison/Output/Graph4-OXY.pdf")
plot(FFC4, plot.single=TRUE, asset.name="OXY", which=18, xlab="")
dev.off()
write.table(round(summary(FFC4)$sum.list[["PG"]]$coefficients,3),
            "fmComparison/Output/Table4-PG")
write.table(round(summary(FFC4)$sum.list[["OXY"]]$coefficients,3),
            "fmComparison/Output/Table4-OXY")

########################

# Table5: Consistency of market beta over 5-yr sub-samples: time
series regressions

# sub-sample regressions (1995-2000, 2001-2007, 2008-2012)
FFC4.1995 <- fitTsfm(asset.names=c("OXY","PG"),
factor.names=colnames(ff.xts)[-1],
                     data=merge(exretM.xts["/2000"], ff.xts["/2000"]))
FFC4.2001 <- fitTsfm(asset.names=c("OXY","PG"),
factor.names=colnames(ff.xts)[-1],
                     data=merge(exretM.xts["2001/2007"],
ff.xts["2001/2007"]))
FFC4.2007 <- fitTsfm(asset.names=c("OXY","PG"),
factor.names=colnames(ff.xts)[-1],
                     data=merge(exretM.xts["2008/2012"],
ff.xts["2008/2012"]))
beta <- rbind(FFC4$beta[c("OXY","PG"),], FFC4.1995$beta,
FFC4.2001$beta, FFC4.2007$beta)
alpha <- rbind(FFC4$alpha[c("OXY"),], FFC4$alpha[c("PG"),],
FFC4.1995$alpha,
               FFC4.2001$alpha, FFC4.2007$alpha)
coef <- cbind(alpha, beta)
rownames(coef) <-
c("OXY.full","PG.full","OXY1","PG1","OXY2","PG2","OXY3","PG3")
colnames(coef) <- c("(Intercept)","mkrtf","smb","hml","umd")
write.table(round(as.data.frame(coef[c(1,3,5,7,2,4,6,8),]),2),
            "fmComparison/Output/Table5")
```

```
############################

# In-sample GMV portfolio

############################

# Sample Covariance
GMV <- MinVar(Omega,Mu)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- as.data.frame(100*c(GMV$sig, GMV$mu, round(weights.df$x,
2)))
colnames(Table6.df) <- "Sample Covariance"
rownames(Table6.df) <- c("Minimum Risk", "Expected Return",
weights.df[,1])

# Time series factor model

# factor model covariance matrices
Omega.FFC4 <- fmCov(FFC4)
Omega.FFC4.rob <- fmCov(FFC4.rob)
Omega.FFC4.lasso <- fmCov(FFC4.lasso)

Mu.FFC4 <- apply(fitted(FFC4), 2, mean)
Mu.FFC4.rob <- apply(fitted(FFC4.rob), 2, mean)
Mu.FFC4.lasso <- apply(fitted(FFC4.lasso), 2, mean)

# In-sample GMV portfolio
GMV <- MinVar(Omega.FFC4, Mu.FFC4)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
GMV <- MinVar(Omega.FFC4.rob, Mu.FFC4.rob)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
GMV <- MinVar(Omega.FFC4.lasso, Mu.FFC4.lasso)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
colnames(Table6.df) <- models[1:4]

# Statistical factor model

# factor model covariance matrices
Omega.SFM4 <- fmCov(SFM4)
Omega.SFM5 <- fmCov(SFM5)

Mu.SFM4 <- apply(fitted(SFM4), 2, mean)
Mu.SFM5 <- apply(fitted(SFM5), 2, mean)
```

```
# In-sample GMV portfolio
GMV <- MinVar(Omega.SFM4, Mu.SFM4)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
GMV <- MinVar(Omega.SFM5, Mu.SFM5)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
colnames(Table6.df) <- models[1:6]

# Fundamental factor model

# factor model covariance matrices
Omega.FFM4 <- FFM4$return.cov
Omega.FFM4.sect <- FFM4.sect$return.cov

Mu.FFM4 <- apply(fitted(FFM4), 2, mean)
Mu.FFM4.sect <- apply(fitted(FFM4.sect), 2, mean)

# In-sample GMV portfolio
GMV <- MinVar(Omega.FFM4, Mu.FFM4)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
GMV <- MinVar(Omega.FFM4.sect, Mu.FFM4.sect)
weights.df <- aggregate(GMV$w,list(stocklist$SECTOR),sum)
Table6.df <- cbind(Table6.df, 100*c(GMV$sig, GMV$mu,
round(weights.df$x, 2)))
colnames(Table6.df) <- models[1:8]

Table6.df[1,] <- sqrt(12)*Table6.df[1,]
Table6.df[2,] <- 12*Table6.df[2,]

View(round(Table6.df,2))

############################

# Out-of-sample performance comparison

############################

# Model 2/3/4: Fama-French-Carhart 4 factor model


# rolling regression
w <- 60 # window size
b.FFC4 <- rep(0,T-w) # standardized return forecast
b.FFC4.rob <- rep(0,T-w) # standardized return forecast
mu.FFC4.eq <- rep(0,T-w) # forecasted return of eq.wt portfolio
mu.FFC4.rob.eq <- rep(0,T-w) # forecasted return of eq.wt portfolio
r.eq <- rep(0,T-w) # realized return of eq.wt portfolio
```

```
for (t in (w+1):T) {
  data <- merge(exretM.xts[(t-w):(t-1),], ff.xts[(t-w):(t-1),])
  FFC4.w <- fitTsfm(asset.names=colnames(exretM.xts),
factor.names=colnames(ff.xts)[-1],
                    data=data)
  FFC4.rob.w <- fitTsfm(asset.names=colnames(exretM.xts),
factor.names=colnames(ff.xts)[-1],
                        data=data, fit.method="Robust")

  Omega.FFC4.w <- fmCov(FFC4.w)
  Omega.FFC4.rob.w <- fmCov(FFC4.rob.w)
  Mu.FFC4.w <- as.matrix(FFC4.w$alpha) + as.matrix(FFC4.w$beta) %*%
t(coredata(ff.xts[t,-1]))
  Mu.FFC4.rob.w <- as.matrix(FFC4.rob.w$alpha) +
as.matrix(FFC4.rob.w$beta) %*% t(coredata(ff.xts[t,-1]))

  # equally weighted portfolio
  w.eq <- rep(1/N, N)
  mu.FFC4.eq[t-w] <- w.eq %*% Mu.FFC4.w
  sig.FFC4.eq <- sqrt(t(w.eq) %*% Omega.FFC4.w %*% w.eq)
  mu.FFC4.rob.eq[t-w] <- w.eq %*% Mu.FFC4.rob.w
  sig.FFC4.rob.eq <- sqrt(t(w.eq) %*% Omega.FFC4.rob.w %*% w.eq)

  # bias statistic
  b.FFC4[t-w] <- mu.FFC4.eq[t-w]/sig.FFC4.eq
  b.FFC4.rob[t-w] <- mu.FFC4.rob.eq[t-w]/sig.FFC4.rob.eq

  # realized return of equally weighted portfolio
  r.eq[t-w] <- w.eq %*% as.vector(exretM.xts[t,])
}

# bias statistic
bias.FFC4 <- sqrt(var(b.FFC4))
bias.FFC4.rob <- sqrt(var(b.FFC4.rob))

# IC
cor(r.eq, mu.FFC4.eq)
cor(r.eq, mu.FFC4.rob.eq)


####################################################
# Model 5/6: Statistical factor model

# rolling regression
w <- 60 # window size
b.SFM4 <- rep(0,T-w) # standardized return forecast
b.SFM5 <- rep(0,T-w) # standardized return forecast
mu.SFM4.eq <- rep(0,T-w) # forecasted return of eq.wt portfolio
mu.SFM5.eq <- rep(0,T-w) # forecasted return of eq.wt portfolio
r.eq <- rep(0,T-w) # realized return of eq.wt portfolio
```

```
for (t in (w+1):T) {
  ret.w <- exretM.xts[(t-w):(t-1),]
  SFM4.w <- fitSfm(ret.w, k=4)
  SFM5.w <- fitSfm(ret.w, k=5)

  Omega.SFM4.w <- fmCov(SFM4.w)
  Omega.SFM5.w <- fmCov(SFM5.w)
  # forcast for next period factor
  Mu.SFM4.w <- as.matrix(SFM4.w$alpha) + as.matrix(SFM4.w$loadings)
%*% coredata(SFM4$factors)[t,]
  Mu.SFM5.w <- as.matrix(SFM5.w$alpha) + as.matrix(SFM5.w$loadings)
%*% coredata(SFM5$factors)[t,]

  # equally weighted portfolio
  w.eq <- rep(1/N, N)
  mu.SFM4.eq[t-w] <- w.eq %*% Mu.SFM4.w
  sig.SFM4.eq <- sqrt(t(w.eq) %*% Omega.SFM4.w %*% w.eq)
  mu.SFM5.eq[t-w] <- w.eq %*% Mu.SFM5.w
  sig.SFM5.eq <- sqrt(t(w.eq) %*% Omega.SFM5.w %*% w.eq)

  # bias statistic
  b.SFM4[t-w] <- mu.SFM4.eq[t-w]/sig.SFM4.eq
  b.SFM5[t-w] <- mu.SFM5.eq[t-w]/sig.SFM5.eq

  # realized return of equally weighted portfolio
  r.eq[t-w] <- w.eq %*% as.vector(exretM.xts[t,])
}

# bias statistic
bias.SFM4 <- sqrt(var(b.SFM4))
bias.SFM5 <- sqrt(var(b.SFM5))

# IC
cor(r.eq, mu.SFM4.eq)
cor(r.eq, mu.SFM5.eq)

#####################################################

# Model 7: FFM

# rolling regression
w <- 60 # window size
b.FFM4 <- rep(0,T-w) # standardized return forecast
b.FFM4.sect <- rep(0,T-w) # standardized return forecast
mu.FFM4.eq <- rep(0,T-w) # forecasted return of eq.wt portfolio
mu.FFM4.sect.eq <- rep(0,T-w) # forecasted return of eq.wt portfolio
r.eq <- rep(0,T-w) # realized return of eq.wt portfolio

for (t in (w+1):T) {
  ret.w <- exretM.xts[(t-w):(t-1),]
  FFM4.w <- fitFfm(data=subset(capIQ200.cs,Date %in%
as.character(index(ret.w))),
```

```
                   asset.var="TICKER", ret.var="ExRET",
date.var="Date", fit.method="WLS",

exposure.vars=c("BP","AnnVol1M","PM12M1M","LogMktCap"), lag.exposures
= FALSE)
  FFM4.sect.w <- fitFfm(data=subset(capIQ200.cs,Date %in%
as.character(index(ret.w))),
                        asset.var="TICKER", ret.var="ExRET",
date.var="Date", fit.method="WLS",

exposure.vars=c("BP","AnnVol1M","PM12M1M","LogMktCap","SECTOR"),
lag.exposures = FALSE)

  Omega.FFM4.w <- FFM4.w$return.cov
  Omega.FFM4.sect.w <- FFM4.sect.w$return.cov

  Mu.FFM4.w <- as.matrix(FFM4.w$beta) %*%
t(coredata(FFM4$factor.returns[t,]))
  Mu.FFM4.sect.w <- as.matrix(FFM4.sect.w$beta) %*%
t(coredata(FFM4.sect$factor.returns[t,]))

  # equally weighted portfolio
  w.eq <- rep(1/N, N)
  mu.FFM4.eq[t-w] <- w.eq %*% Mu.FFM4.w
  sig.FFM4.eq <- sqrt(t(w.eq) %*% Omega.FFM4.w %*% w.eq)
  mu.FFM4.sect.eq[t-w] <- w.eq %*% Mu.FFM4.sect.w
  sig.FFM4.sect.eq <- sqrt(t(w.eq) %*% Omega.FFM4.sect.w %*% w.eq)

  # bias statistic
  b.FFM4[t-w] <- mu.FFM4.eq[t-w]/sig.FFM4.eq
  b.FFM4.sect[t-w] <- mu.FFM4.sect.eq[t-w]/sig.FFM4.sect.eq

  # realized return of equally weighted portfolio
  r.eq[t-w] <- w.eq %*% as.vector(exretM.xts[t,])
}

# bias statistic
bias.FFM4 <- sqrt(var(b.FFM4))
bias.FFM4.sect <- sqrt(var(b.FFM4.sect))

# IC
cor(r.eq, mu.FFM4.eq)
cor(r.eq, mu.FFM4.sect.eq)
```

APPENDIX B: ESTIMATED TIME-SERIES MODEL IN CHAPTER 1

	(Intercept)	mktrf	smb	hml	umd
AAPL	0.023	1.06	0.36	-0.704	-0.125
ABM	0.001	0.838	0.707	0.682	0.063
ABT	0.006	0.459	-0.512	0.017	0.071
AGN	0.01	0.603	0	0.118	-0.016
AIR	0.002	0.996	1.47	0.724	-0.246
ALB	0.006	1.222	0.06	0.606	-0.178
ALK	0.007	0.819	0.08	0.327	-0.195
AMGN	0.009	0.64	-0.112	-0.444	0.128
AOS	0.007	0.641	0.574	0.731	-0.133
APOG	0.006	1.026	0.719	0.823	-0.29
ARCB	-0.002	0.948	0.567	0.985	-0.262
ARW	-0.001	1.521	0.52	0.157	-0.202
ASTE	0.008	1.111	0.58	0.46	-0.081
AVT	0	1.215	0.708	0.143	-0.447
AXP	0.005	1.39	-0.22	0.626	-0.4
BA	0	0.972	0.051	0.796	-0.174
BHE	0.005	0.999	0.909	-0.182	-0.19
BID	0.002	1.782	0.277	0.494	-0.02
BKH	0.001	0.8	0.085	0.915	0.178
BMY	0.004	0.727	-0.656	0.12	0.135
BOBE	0	0.52	0.819	0.797	-0.032
CASY	0.005	0.666	-0.034	0.65	0.06
CAT	0.004	1.329	0.079	0.935	-0.166
CATO	0.011	0.45	0.861	0.547	-0.457
CBT	0.001	1.198	0.485	1.198	-0.298
CDI	-0.002	1.023	0.774	1.072	-0.286
CDNS	0.001	1.541	0.275	-0.26	0.027
CGNX	0.004	1.52	0.187	-0.358	-0.058
CHD	0.012	0.375	-0.156	0.197	-0.012
CL	0.007	0.729	-0.615	-0.104	0.154
CSCO	0.01	1.382	0.04	-0.973	-0.203
CSH	0.007	0.89	0.739	0.659	-0.291
CYN	0.001	0.971	-0.007	0.976	0.038
DCI	0.007	0.701	0.434	0.451	-0.155
DD	-0.001	1.095	-0.091	0.578	-0.18
DGII	-0.002	1.164	1.111	-0.015	-0.242
DIS	0.001	1.03	-0.082	0.281	-0.151

DOW	-0.001	1.362	0.392	1.24	-0.475
EAT	0.006	0.909	0.255	0.606	-0.25
EMR	0.003	1.002	-0.246	0.353	-0.136
ETH	0.002	1.336	0.703	1.152	-0.162
GE	0	1.339	-0.41	0.304	0.016
GFF	-0.005	1.082	0.834	0.865	0.171
GK	-0.001	0.848	0.23	0.924	-0.083
GMT	-0.002	1.168	0.371	1.04	-0.042
GVA	0.005	0.634	0.596	0.657	0.043
HAL	0.005	1.338	0.069	0.245	-0.136
HD	0.005	1.062	0.069	0.125	0.086
HE	0.002	0.355	-0.058	0.583	0.036
HPQ	-0.002	1.189	0.607	-0.441	-0.06
HSC	-0.002	1.198	0.336	0.892	-0.152
HTLD	0.004	0.546	0.37	0.583	-0.069
HUB.B	0.003	0.9	0.044	0.426	-0.136
IBM	0.011	0.839	-0.295	-0.614	-0.377
INTC	0.008	1.167	-0.106	-0.961	-0.285
IVC	-0.007	0.651	0.572	0.596	0.289
JBHT	0.011	1.022	0.267	0.414	-0.253
JJSF	0.006	0.76	0.63	0.478	0.168
JNJ	0.006	0.575	-0.597	0.027	0.083
KAMN	0.001	1.017	0.797	0.852	0.102
KMT	-0.001	1.253	0.418	0.871	-0.135
KO	0.003	0.591	-0.378	0.104	-0.016
KWR	0.002	1.382	0.683	1.638	-0.163
LANC	0.004	0.428	0.156	0.455	0
LDL	-0.003	0.727	1.067	1.158	-0.58
LLY	0.005	0.652	-0.472	0.125	0.072
LMT	0.003	0.477	0.106	0.801	-0.075
LNN	0.007	0.877	0.626	0.661	-0.053
LOW	0.005	1.073	0.112	0.455	0.021
LZB	0.002	1.383	1.239	1.708	-0.225
MAN	-0.002	1.295	0.254	0.592	-0.19
MCD	0.005	0.778	-0.564	0.196	0.093
MCS	-0.001	0.954	0.453	0.757	-0.204
MDT	0.003	0.776	-0.109	0.163	0.074
MENT	0.003	1.382	0.902	-0.036	-0.312
MLHR	0.003	1.064	0.367	0.783	-0.16
MLI	0.004	1.067	0.221	0.506	0.105
MYE	0.002	1.084	1.161	1.212	-0.35
NDSN	0.003	1.058	0.361	0.497	-0.079
NFG	0.003	0.743	-0.024	0.639	0.147

NJR	0.005	0.341	-0.006	0.31	0.209
NKE	0.01	0.85	-0.519	0.371	-0.138
NWN	0.002	0.284	0.061	0.489	0.161
OLN	-0.004	1.175	0.419	1.13	0.38
ORCL	0.013	1.255	0.34	-1.283	0.259
OXM	0.008	1.131	1.182	1.026	-0.543
OXY	0.006	0.96	-0.13	0.784	0.072
PEP	0.003	0.732	-0.41	0.091	0.214
PG	0.006	0.442	-0.288	0.197	0.068
PNC	0.002	1	-0.171	0.982	-0.266
PNY	0.004	0.393	-0.118	0.613	0.158
PRGS	0.002	1.159	0.652	-0.083	0.265
RGR	0.011	0.207	0.734	0.157	-0.176
RPM	0.002	0.905	0.343	0.527	-0.174
RYN	0.006	0.861	0.188	0.536	-0.174
SCHL	-0.002	0.833	0.26	0.28	-0.118
SIGI	-0.001	0.742	0.561	1.114	0.043
SKYW	0.002	1.172	0.35	0.782	0.041
SMP	0.001	1.203	1.061	1.423	-0.378
SMRT	0.006	1.373	0.253	1.205	-0.617
SO	0.005	0.22	-0.16	0.561	0.17
SON	-0.001	0.873	-0.009	0.461	-0.151
SONC	0.002	1.002	0.441	0.566	0.123
SPF	0.004	1.529	1.207	1.806	-0.471
SWX	0.002	0.642	0.158	0.531	0.109
TDW	0	1.197	0.195	0.485	-0.044
TFX	0.003	0.774	0.191	0.653	-0.124
TG	0.001	0.839	0.769	0.525	0.197
TRN	-0.002	1.568	0.766	1.358	-0.264
TTC	0.006	0.91	0.544	0.709	0.002
TTI	0.003	1.578	0.818	1.403	-0.418
TXI	-0.001	1.453	0.632	1.474	-0.031
TXN	0.006	1.398	0.292	-0.736	0.046
UFPI	0.003	1.061	0.795	0.889	-0.14
UNP	0.005	0.799	-0.038	0.61	-0.101
UTX	0.006	1.088	-0.269	0.271	-0.051
VICR	-0.002	1.501	0.434	-0.131	-0.204
VSH	0.002	1.541	0.779	0.157	-0.778
WDFC	0.003	0.595	0.145	0.107	-0.014
WGO	0.003	1.468	1.632	1.441	-0.373
WMT	0.007	0.619	-0.527	-0.13	0.065
WWW	0.005	0.838	0.676	0.54	-0.184

APPENDIX C: ESTIMATED STATISTICAL MODEL IN CHAPTER 1

	(Intercept)	F.1	F.2	F.3	F.4	F.5
AAPL	0.018	0.082	-0.227	-0.067	0.157	0.215
ABM	0	0.078	-0.003	-0.052	-0.064	0.075
ABT	0.006	0.013	0.001	0.057	-0.045	-0.043
AGN	0.009	0.042	0.001	0.031	-0.039	-0.033
AIR	-0.004	0.113	-0.099	0.009	-0.248	0.168
ALB	0.003	0.105	0.024	0.058	0.081	0.001
ALK	0.001	0.073	-0.029	0.047	-0.151	0.103
AMGN	0.008	0.029	-0.109	-0.044	-0.117	-0.01
AOS	0.004	0.075	0.056	0.034	-0.011	0.051
APOG	0.002	0.099	0.082	0.046	-0.066	0.082
ARCB	-0.006	0.104	0.072	0.089	0.015	0.04
ARW	-0.007	0.135	-0.163	0.079	0.054	0.044
ASTE	0.004	0.1	-0.045	0.096	0.039	0.106
AVT	-0.007	0.124	-0.156	0.043	0.056	0.019
AXP	0.001	0.112	0.044	-0.015	-0.008	-0.162
BA	-0.002	0.076	0.035	0.128	0.012	-0.016
BHE	-0.001	0.104	-0.196	-0.108	0.068	0.162
BID	0	0.137	-0.023	0.05	-0.013	-0.153
BKH	0.002	0.052	0.043	0.063	-0.081	-0.058
BMY	0.003	0.025	0	0.096	-0.083	-0.041
BOBE	-0.001	0.064	0.062	-0.019	-0.041	0.082
CASY	0.003	0.053	0.043	0.038	-0.076	0.019
CAT	0	0.114	0.045	0.108	0.022	-0.039
CATO	0.005	0.076	0.038	0.027	0.025	0.204
CBT	-0.002	0.111	0.093	0.019	0.002	0.01
CDI	-0.005	0.109	0.102	0.057	0.074	0.026
CDNS	-0.002	0.113	-0.178	0.003	-0.003	-0.131
CGNX	-0.001	0.126	-0.189	0.016	0.127	-0.035
CHD	0.011	0.025	0.027	0.023	-0.004	-0.024
CL	0.007	0.029	0.001	0.079	-0.017	-0.057
CSCO	0.003	0.1	-0.224	-0.051	-0.01	-0.001
CSH	0.003	0.09	0.043	-0.033	-0.038	0.036
CYN	0	0.071	0.049	0.052	-0.057	0.04
DCI	0.004	0.073	0.02	0.038	-0.019	0.002
DD	-0.004	0.086	0.057	0.05	0.007	-0.041
DGII	-0.009	0.106	-0.184	0.058	-0.161	0.157
DIS	-0.002	0.077	-0.037	0.041	0.048	-0.027

DOW	-0.005	0.134	0.115	-0.022	0.02	-0.099
EAT	0.001	0.092	0.064	0.035	0.108	0.172
EMR	0	0.075	0.001	0.082	0.023	-0.053
ETH	0	0.121	0.042	0.02	-0.085	-0.019
GE	-0.002	0.081	-0.011	0.005	-0.036	-0.049
GFF	-0.005	0.092	-0.002	0.015	-0.075	0.086
GK	-0.003	0.079	0.075	0.003	-0.077	-0.037
GMT	-0.003	0.101	0.072	0.023	-0.034	-0.029
GVA	0.004	0.056	0	0.044	-0.055	0.082
HAL	0.003	0.102	-0.082	0.12	0.118	-0.17
HD	0.003	0.073	-0.044	-0.036	-0.099	0.005
HE	0.003	0.023	0.052	0.005	-0.024	-0.048
HPQ	-0.005	0.095	-0.199	-0.066	0.05	0.013
HSC	-0.004	0.104	0.008	0.132	0.055	-0.004
HTLD	0.002	0.057	0.018	0.062	-0.053	0.04
HUB.B	0	0.074	0.016	0.071	0.024	-0.002
IBM	0.005	0.064	-0.122	0.002	0.078	0.014
INTC	0.001	0.088	-0.252	-0.066	0.086	0.076
IVC	-0.005	0.046	0.008	-0.011	-0.029	0.011
JBHT	0.006	0.098	-0.014	0.066	-0.044	0.04
JJSF	0.005	0.066	0.024	0.042	-0.093	0.084
JNJ	0.005	0.022	0.009	0.067	-0.045	-0.074
KAMN	0.001	0.096	0.039	-0.023	-0.084	-0.012
KMT	-0.004	0.113	0.03	0.113	0.04	0.013
KO	0.002	0.03	0.03	0.099	-0.027	-0.055
KWR	0	0.13	0.142	0.008	0.011	0.017
LANC	0.002	0.037	0.043	0.018	-0.084	0.04
LDL	-0.006	0.101	0.079	-0.016	0.093	0.01
LLY	0.004	0.031	0.006	0.076	-0.039	-0.071
LMT	0.003	0.045	0.1	0.071	0.035	0.022
LNN	0.007	0.085	0.059	0.016	-0.005	-0.156
LOW	0.003	0.081	0.001	-0.043	-0.071	0.055
LZB	0.004	0.157	0.14	-0.513	-0.044	-0.31
MAN	-0.006	0.113	-0.028	0.039	0.018	-0.057
MCD	0.004	0.041	-0.016	0.089	-0.021	0.001
MCS	-0.005	0.092	0.079	-0.034	-0.063	0.025
MDT	0.003	0.047	-0.032	0.016	-0.04	-0.029
MENT	0	0.128	-0.176	-0.086	-0.021	-0.374
MLHR	0	0.098	-0.007	-0.026	0.002	-0.022
MLI	0.002	0.079	0.003	0.122	-0.014	0.053
MYE	-0.004	0.13	0.097	-0.075	-0.065	0.163
NDSN	0.002	0.091	0.006	-0.031	0.063	-0.037
NFG	0.004	0.049	0.039	0.067	0.025	-0.083

NJR	0.006	0.019	0.029	0.046	-0.033	0
NKE	0.007	0.064	0.058	0.066	0.005	-0.055
NWN	0.004	0.016	0.048	0.033	-0.039	-0.001
OLN	-0.002	0.083	0.064	0.143	0.031	0.077
ORCL	0.01	0.082	-0.271	-0.11	-0.001	-0.003
OXM	0.003	0.133	0.108	-0.137	0.043	0.122
OXY	0.006	0.065	0.065	0.152	0.064	-0.06
PEP	0.003	0.034	0	0.056	-0.052	-0.073
PG	0.005	0.021	0.012	0.046	-0.119	-0.067
PNC	0	0.074	0.071	0.044	-0.044	-0.052
PNY	0.005	0.021	0.057	0.037	-0.087	-0.03
PRGS	0.001	0.083	-0.124	-0.074	-0.122	-0.01
RGR	0.008	0.045	0.016	-0.009	0.089	0.119
RPM	-0.001	0.081	0.023	0.031	0.022	0.084
RYN	0.003	0.078	0.017	0.007	0.016	0.021
SCHL	-0.003	0.068	0.022	-0.051	-0.036	-0.119
SIGI	-0.001	0.069	0.058	0.01	-0.078	-0.007
SKYW	-0.002	0.093	-0.049	0.048	-0.225	0.109
SMP	-0.002	0.132	0.116	-0.196	0.09	0.112
SMRT	0.003	0.145	0.218	-0.165	0.496	0.107
SO	0.007	0.006	0.062	0.044	-0.054	-0.013
SON	-0.004	0.072	0.017	0.065	-0.002	-0.021
SONC	0	0.082	0.057	-0.017	-0.067	0.116
SPF	0.001	0.162	0.134	-0.355	-0.204	-0.053
SWX	0.002	0.046	0.032	0.037	-0.055	-0.003
TDW	-0.001	0.09	-0.026	0.136	0.02	-0.146
TFX	0	0.069	0.016	0.104	-0.017	0.044
TG	0.001	0.073	-0.039	0.027	-0.123	0.023
TRN	-0.006	0.15	0.071	0.087	0.009	-0.066
TTC	0.005	0.08	0.029	0.011	-0.051	-0.021
TTI	0	0.155	0.09	0.116	0.267	-0.057
TXI	-0.003	0.126	0.073	0.098	-0.011	0.069
TXN	0.001	0.105	-0.267	-0.062	0.082	0.034
UFPI	0	0.105	0.046	-0.069	-0.043	0.096
UNP	0.003	0.066	0.051	0.054	0.006	-0.054
UTX	0.003	0.075	-0.026	0.089	-0.022	-0.022
VICR	-0.007	0.125	-0.129	0.118	-0.077	-0.237
VSH	-0.007	0.169	-0.113	-0.09	0.241	-0.068
WDFC	0.002	0.046	0.017	0.001	-0.046	-0.015
WGO	-0.003	0.167	0.027	-0.211	-0.176	0.151
WMT	0.005	0.033	-0.008	0.042	-0.025	-0.034
WWW	0.002	0.09	0.011	0	0.011	-0.008

APPENDIX D: ESTIMATED FUNDAMENTAL MODEL IN CHAPTER 1

Month	BP	AnnVol1M	PM12M1M	LogMktCap	Consumer Discretionary	Consumer Staples	Energy	Financials	Health Care	Industrials	Information Technology	Materials	Utilities
1	0.012	0.108	0.003	-0.005	0.061	0.076	0.106	0.098	0.109	0.073	0.067	0.096	0.093
2	-0.047	-0.06	0.077	0.001	0.012	0.009	0.022	0.044	0.04	0	-0.042	0.041	0.04
3	-0.102	-0.149	0.034	0	0.05	0.045	0.071	0.051	0.071	0.047	0.113	0.017	0.115
4	-0.129	0.045	-0.076	0.002	0.081	0.025	0.109	0.106	0.037	0.064	-0.003	0.076	0.091
5	-0.072	-0.064	0.04	-0.002	0.096	0.04	0.141	0.073	0.088	0.096	0.033	0.08	0.103
6	-0.099	-0.116	-0.068	0.004	-0.011	0.025	0.037	0.049	0.033	0.037	0.013	0.056	0.006
7	-0.101	-0.036	-0.051	-0.005	0.209	0.193	0.199	0.203	0.188	0.21	0.257	0.224	0.174
8	0.026	0.066	-0.005	0	0.044	-0.009	0.095	-0.035	-0.013	0.014	-0.082	-0.005	-0.039
9	-0.1	0.042	-0.014	-0.007	0.174	0.164	0.228	0.193	0.136	0.199	0.251	0.2	0.186
10	-0.113	0.183	-0.036	-0.003	0.101	0.092	0.036	0.113	0.089	0.11	0.056	0.081	0.105
11	-0.115	-0.021	-0.021	0.004	-0.054	-0.056	-0.066	0.016	-0.075	-0.033	-0.111	-0.078	0.021
12	-0.118	-0.163	-0.048	-0.006	0.193	0.147	0.161	0.249	0.207	0.18	0.193	0.217	0.197
13	-0.13	0.038	-0.133	-0.004	0.188	0.16	0.192	0.204	0.162	0.181	0.16	0.174	0.214
14	-0.053	-0.087	0.016	0.008	-0.064	-0.126	-0.133	-0.081	-0.084	-0.085	-0.049	-0.107	-0.124
15	0.002	-0.06	0.065	0.001	-0.059	0.007	0.087	-0.008	-0.04	-0.014	-0.026	-0.027	0.006
16	-0.105	0.174	-0.056	0.003	-0.038	-0.047	-0.013	0.032	-0.042	0.007	0.048	-0.012	0.004
17	-0.083	-0.024	-0.004	-0.016	0.389	0.399	0.377	0.434	0.366	0.384	0.345	0.346	0.372
18	0.025	-0.083	0.055	0.013	-0.272	-0.226	-0.161	-0.226	-0.225	-0.249	-0.245	-0.237	-0.258
19	0.009	-0.206	-0.011	0.004	0.009	0.005	-0.102	0.02	-0.028	-0.027	-0.06	-0.034	-0.067
20	-0.007	-0.062	-0.004	-0.006	0.137	0.15	0.172	0.112	0.104	0.137	0.138	0.139	0.117
21	-0.076	-0.114	-0.022	0.012	-0.18	-0.187	-0.233	-0.173	-0.188	-0.17	-0.159	-0.196	-0.192
22	-0.078	0.194	-0.078	-0.006	0.185	0.176	0.164	0.207	0.174	0.198	0.193	0.17	0.174
23	0.005	0.285	-0.051	0.004	-0.102	-0.111	-0.144	-0.097	-0.118	-0.115	-0.18	-0.1	-0.088
24	0.027	0.256	-0.015	0.008	-0.208	-0.193	-0.158	-0.139	-0.278	-0.183	-0.119	-0.171	-0.245
25	-0.041	0.148	-0.003	-0.019	0.362	0.32	0.327	0.345	0.348	0.374	0.413	0.381	0.395
26	-0.036	0.151	-0.05	-0.005	0.158	0.162	0.2	0.176	0.167	0.154	0.128	0.153	0.146
27	-0.022	0.02	0.004	-0.005	0.08	0.084	0.199	0.098	0.122	0.087	0.042	0.093	0.113
28	0.035	-0.027	-0.027	0.01	-0.179	-0.196	-0.271	-0.172	-0.198	-0.209	-0.211	-0.188	-0.158
29	-0.095	-0.259	0.023	-0.008	0.309	0.3	0.266	0.333	0.282	0.292	0.279	0.255	0.353
30	-0.148	0.111	0.05	-0.002	0.057	0.016	-0.116	-0.017	0.012	0.055	0.084	0.018	-0.003
31	-0.058	0.229	-0.012	-0.003	0.128	0.103	0.058	0.083	0.078	0.087	0.101	0.135	0.087
32	-0.121	0.233	-0.046	-0.007	0.177	0.222	0.181	0.234	0.185	0.174	0.131	0.202	0.218
33	0.057	0.131	-0.002	0.007	-0.184	-0.235	-0.216	-0.174	-0.183	-0.215	-0.189	-0.167	-0.225
34	-0.003	-0.083	0.005	0.003	-0.055	-0.057	-0.136	-0.078	-0.07	-0.093	-0.096	-0.073	-0.078
35	-0.155	-0.074	0.098	-0.002	0.086	0.121	0.022	0.06	0.132	0.087	0.088	0.031	0.169

36	-0.231	-0.14	-0.023	-0.003	0.111	0.117	0.054	0.172	0.169	0.14	0.187	0.089	0.163
37	-0.024	-0.361	0.054	-0.007	0.184	0.16	0.144	0.124	0.171	0.171	0.19	0.166	0.224
38	-0.232	-0.116	-0.005	-0.019	0.627	0.561	0.655	0.608	0.674	0.59	0.731	0.573	0.67
39	-0.111	0.051	-0.074	0.008	-0.125	-0.038	-0.031	-0.077	-0.122	-0.092	-0.031	-0.071	-0.099
40	-0.141	0.019	-0.056	-0.001	0.158	0.092	-0.005	0.197	0.103	0.141	0.145	0.113	0.103
41	-0.151	-0.033	0.034	-0.013	0.418	0.363	0.337	0.394	0.412	0.36	0.494	0.365	0.381
42	-0.132	0.171	-0.004	0.004	-0.119	-0.145	-0.181	-0.145	-0.101	-0.087	-0.087	-0.156	-0.121
43	-0.101	0.005	-0.038	0.011	-0.211	-0.247	-0.284	-0.234	-0.251	-0.247	-0.299	-0.241	-0.181
44	-0.177	-0.013	-0.036	-0.007	0.246	0.252	0.508	0.266	0.258	0.244	0.26	0.238	0.226
45	0.097	-0.122	-0.107	0.015	-0.245	-0.281	-0.214	-0.233	-0.263	-0.188	-0.217	-0.156	-0.292
46	-0.056	0.089	-0.072	-0.008	0.164	0.161	0.111	0.154	0.195	0.14	0.191	0.118	0.218
47	-0.038	0.16	0.073	-0.004	0.09	0.087	0.146	0.076	0.066	0.106	0.135	0.067	0.051
48	-0.118	-0.116	-0.05	-0.006	0.171	0.182	0.208	0.147	0.166	0.182	0.269	0.192	0.246
49	-0.024	0.113	-0.049	0.013	-0.34	-0.312	-0.217	-0.34	-0.277	-0.333	-0.246	-0.341	-0.272
50	-0.074	0.064	0.016	-0.008	0.186	0.119	0.085	0.254	0.109	0.158	0.143	0.176	0.169
51	-0.11	-0.193	-0.023	0.011	-0.116	-0.037	-0.097	-0.027	-0.111	-0.1	-0.031	-0.146	-0.131
52	-0.063	0.273	0.011	-0.009	0.13	0.134	0.058	0.096	0.137	0.095	0.23	0.132	0.129
53	-0.141	-0.026	0.032	0.003	0.049	-0.009	0.071	0.029	-0.057	0.045	0.134	0.066	0.002
54	-0.085	0.038	-0.038	-0.004	0.028	0.013	0.054	0.097	0.12	0.034	0.11	0.047	0.098
55	-0.052	0.138	-0.063	-0.008	0.076	0.01	0.135	0.061	0.105	0.09	0.343	0.095	0.086
56	-0.114	0.102	-0.034	-0.001	0.142	0.036	0.267	0.189	0.033	0.124	0.103	0.165	0.119
57	-0.119	0.019	0.001	-0.006	0.154	0.14	0.201	0.14	0.191	0.224	0.141	0.145	0.263
58	-0.05	0.085	-0.054	0.004	-0.116	-0.088	0.041	-0.042	-0.06	-0.095	-0.157	-0.106	-0.056
59	-0.116	0.076	-0.052	-0.008	0.187	0.172	0.16	0.098	0.259	0.16	0.287	0.125	0.151
60	-0.012	0.292	-0.028	0.005	-0.177	-0.236	-0.284	-0.166	-0.278	-0.161	-0.296	-0.178	-0.15
61	-0.016	-0.034	0.015	-0.002	0.105	0.012	0.164	0.113	0.083	0.089	0.162	0.077	0.116
62	-0.158	-0.22	-0.029	-0.019	0.581	0.596	0.625	0.564	0.614	0.578	0.533	0.519	0.621
63	-0.086	-0.15	-0.042	-0.007	0.337	0.288	0.198	0.201	0.266	0.308	0.29	0.33	0.232
64	-0.028	-0.283	-0.003	-0.003	0.174	0.209	0.144	0.192	0.239	0.18	0.099	0.154	0.185
65	-0.119	-0.028	-0.124	-0.022	0.643	0.594	0.687	0.658	0.656	0.639	0.581	0.666	0.677
66	-0.031	0.032	-0.114	-0.016	0.436	0.327	0.474	0.398	0.338	0.392	0.497	0.336	0.278
67	0.026	-0.279	-0.026	-0.01	0.3	0.301	0.347	0.285	0.352	0.296	0.19	0.304	0.264
68	-0.05	-0.079	-0.071	-0.011	0.267	0.256	0.284	0.307	0.259	0.269	0.239	0.281	0.363
69	-0.05	0.097	-0.058	-0.009	0.248	0.218	0.376	0.255	0.228	0.277	0.278	0.216	0.23
70	-0.004	-0.014	0.014	-0.006	0.138	0.162	0.153	0.139	0.178	0.174	0.128	0.176	0.165
71	0.011	0.221	-0.013	-0.007	0.106	0.131	-0.104	0.113	0.077	0.071	0.047	0.081	0.073
72	-0.025	-0.07	0.007	0.002	0.001	0.014	0	0.029	0.022	0.046	0.009	0.003	-0.02
73	-0.001	-0.27	-0.026	-0.012	0.334	0.319	0.26	0.285	0.343	0.303	0.31	0.347	0.264
74	-0.097	-0.213	0.013	-0.001	0.087	0.099	0.044	0.097	0.089	0.08	0.007	0.094	0.115
75	-0.098	0.017	-0.049	-0.01	0.309	0.294	0.312	0.234	0.221	0.298	0.362	0.292	0.243
76	-0.006	0.283	0.043	0.008	-0.165	-0.21	-0.317	-0.161	-0.18	-0.167	-0.132	-0.157	-0.196
77	-0.055	-0.112	-0.034	-0.017	0.502	0.477	0.466	0.468	0.444	0.507	0.484	0.46	0.49

78	-0.015	0.137	0.024	-0.002	0.045	0.051	-0.004	0	-0.017	0.032	0.051	-0.024	0.003
79	-0.065	-0.041	0.003	-0.004	0.167	0.143	0.288	0.15	0.141	0.144	0.059	0.184	0.144
80	-0.01	0.036	-0.136	-0.018	0.432	0.476	0.459	0.489	0.428	0.48	0.441	0.45	0.431
81	0.033	-0.07	0.083	-0.015	0.355	0.309	0.34	0.364	0.314	0.319	0.287	0.332	0.335
82	-0.063	-0.185	-0.024	0.004	-0.019	-0.025	-0.022	-0.063	-0.057	-0.033	-0.041	-0.026	-0.038
83	-0.121	-0.014	0.011	-0.018	0.414	0.428	0.401	0.427	0.369	0.424	0.359	0.465	0.443
84	-0.023	-0.203	-0.009	0.018	-0.383	-0.371	-0.406	-0.347	-0.32	-0.369	-0.381	-0.38	-0.331
85	-0.09	-0.213	-0.096	-0.012	0.383	0.415	0.493	0.403	0.372	0.381	0.384	0.393	0.411
86	-0.073	-0.14	0.027	-0.013	0.3	0.338	0.324	0.292	0.348	0.319	0.223	0.272	0.365
87	-0.108	0.096	-0.084	-0.001	0.061	0.052	0.1	0.066	0.009	0.045	0.167	0.062	0.074
88	0.033	-0.008	-0.173	-0.003	0.081	0.021	0.148	0.098	0.079	0.11	0.238	0.086	0.005
89	0.02	-0.187	-0.004	-0.009	0.204	0.26	0.229	0.216	0.227	0.227	0.12	0.217	0.258
90	-0.037	-0.124	-0.034	-0.002	0.067	0.044	0.094	0.107	0.082	0.049	0.169	0.024	0.066
91	-0.045	-0.242	-0.008	-0.004	0.166	0.147	0.234	0.162	0.151	0.144	0.232	0.135	0.13
92	-0.089	-0.103	-0.114	-0.007	0.241	0.249	0.229	0.218	0.21	0.212	0.206	0.247	0.262
93	-0.065	0.356	-0.127	0	0.001	-0.014	-0.032	-0.024	-0.011	0.008	-0.026	0.015	0.019
94	0.013	0.178	-0.043	0.001	-0.038	-0.054	0.07	0.026	-0.047	-0.044	0.029	-0.057	0.005
95	-0.001	0.355	-0.032	0.003	-0.175	-0.134	-0.197	-0.154	-0.144	-0.135	-0.23	-0.13	-0.124
96	-0.101	0.364	-0.044	-0.006	0.108	0.111	0.072	0.175	0.12	0.154	0.144	0.141	0.109
97	-0.046	0.062	-0.005	-0.004	0.14	0.103	0.175	0.16	0.104	0.151	0.211	0.101	0.134
98	0.011	-0.056	-0.005	0.001	-0.02	-0.003	-0.038	-0.03	-0.051	-0.054	-0.052	-0.069	-0.023
99	-0.004	0.144	0.032	0	0.037	0.005	-0.043	0.048	-0.028	0.027	0.049	0.032	-0.004
100	-0.049	0.05	-0.059	-0.007	0.207	0.2	0.189	0.22	0.171	0.211	0.208	0.198	0.197
101	-0.024	0.171	-0.025	0.006	-0.113	-0.141	-0.041	-0.098	-0.098	-0.105	-0.126	-0.05	-0.114
102	-0.017	0.068	-0.023	-0.003	0.105	0.065	0.139	0.099	0.088	0.055	0.138	0.042	0.091
103	-0.07	0.118	-0.003	-0.004	0.134	0.137	0.127	0.14	0.101	0.09	0.047	0.091	0.126
104	-0.044	0.108	-0.006	-0.004	0.091	0.092	0.057	0.076	0.047	0.082	0.077	0.074	0.104
105	-0.09	0.131	-0.072	0.001	-0.05	-0.037	-0.022	-0.027	-0.018	-0.006	-0.089	-0.011	-0.017
106	-0.044	-0.048	0.024	-0.001	0.059	0.042	0.031	0.071	0.047	0.051	0.1	0.068	0.061
107	-0.064	0.405	0.011	-0.002	-0.009	-0.008	0.043	0.037	0	0.041	-0.035	0.033	0.06
108	-0.003	-0.237	0.044	0	0.034	-0.012	0.069	0.007	0.005	0.012	-0.006	0.052	0.018
109	-0.003	-0.161	-0.014	-0.003	0.103	0.14	0.097	0.119	0.131	0.106	0.061	0.131	0.119
110	-0.004	-0.005	0.008	-0.002	0.076	0.021	0.171	0.064	0.041	0.087	0.081	0.097	0.066
111	-0.035	-0.101	0.082	0.002	0.023	-0.017	-0.03	0.021	-0.02	0.009	0.092	-0.046	0.021
112	0.046	0.128	0.007	-0.006	0.123	0.128	0.144	0.128	0.127	0.169	0.145	0.177	0.111
113	-0.043	0.231	-0.053	0.003	-0.047	-0.081	-0.122	-0.039	-0.059	-0.065	-0.064	-0.059	-0.067
114	-0.064	-0.055	0.079	0	-0.005	0.033	0.045	-0.025	-0.004	-0.018	-0.011	-0.033	0.028
115	-0.048	-0.025	0.039	0	0.03	0.024	0.131	0.046	0.03	0.034	0.073	0.082	0.03
116	0.029	-0.081	-0.023	0.003	-0.085	-0.071	-0.077	-0.101	-0.086	-0.053	-0.074	-0.097	-0.069
117	-0.065	-0.24	-0.02	-0.001	0.055	0.088	0.096	0.108	0.119	0.083	0.062	0.058	0.116
118	0.028	0.325	-0.045	0.006	-0.122	-0.173	-0.185	-0.135	-0.152	-0.166	-0.111	-0.211	-0.146
119	0.001	0.24	-0.059	-0.005	0.083	0.069	0.18	0.12	0.094	0.07	0.056	0.078	0.116

120	-0.002	0.358	0.01	0.005	-0.164	-0.143	-0.12	-0.16	-0.175	-0.137	-0.148	-0.143	-0.141
121	-0.073	0.193	0.05	0.001	-0.092	-0.05	-0.005	-0.031	-0.027	-0.025	-0.075	-0.055	-0.02
122	-0.062	-0.194	0.013	0	0.052	0.042	0.159	0.086	0.023	0.08	0.083	0.066	0.102
123	-0.017	-0.055	-0.015	0.003	-0.045	-0.058	-0.113	0.003	-0.092	-0.038	-0.071	-0.046	-0.087
124	-0.084	0.262	-0.101	0.003	-0.057	-0.079	-0.069	-0.05	-0.091	-0.042	-0.017	-0.054	-0.077
125	0.002	-0.094	0.024	-0.005	0.104	0.111	0.123	0.107	0.153	0.115	0.117	0.131	0.097
126	0.02	0.136	-0.039	-0.009	0.179	0.228	0.447	0.241	0.205	0.238	0.199	0.219	0.209
127	-0.069	0.039	0.001	0.003	-0.035	-0.059	-0.16	-0.036	-0.082	-0.028	-0.063	-0.021	-0.038
128	-0.142	0.013	-0.03	-0.013	0.385	0.341	0.444	0.376	0.36	0.386	0.382	0.353	0.356
129	-0.047	0.153	-0.007	0.005	-0.174	-0.14	-0.083	-0.129	-0.175	-0.121	-0.106	-0.123	-0.111
130	0.041	-0.116	-0.014	0.009	-0.228	-0.197	-0.178	-0.225	-0.239	-0.232	-0.268	-0.263	-0.195
131	-0.066	0.107	-0.024	-0.005	0.11	0.138	0.118	0.11	0.119	0.116	0.113	0.152	0.157
132	-0.072	-0.238	0.003	-0.008	0.253	0.229	0.276	0.284	0.299	0.229	0.273	0.272	0.292
133	-0.006	0.095	-0.084	-0.002	0.06	0.083	0.023	0.06	0.015	0.034	0.087	0.039	0.039
134	-0.06	0.181	-0.039	0.006	-0.089	-0.14	-0.28	-0.119	-0.137	-0.125	-0.132	-0.096	-0.134
135	-0.111	-0.014	-0.039	-0.01	0.331	0.279	0.367	0.304	0.303	0.319	0.345	0.349	0.328
136	0.013	0.083	-0.019	-0.001	0.013	0.009	0.045	0.022	-0.001	0.012	0.004	0.02	0.01
137	0.059	-0.055	0.045	0.003	-0.089	-0.067	-0.164	-0.063	-0.078	-0.095	-0.098	-0.072	-0.096
138	-0.042	0.129	-0.032	0.005	-0.091	-0.096	-0.175	-0.147	-0.109	-0.065	-0.11	-0.06	-0.104
139	-0.048	0.011	-0.023	-0.005	0.129	0.126	0.145	0.13	0.09	0.14	0.146	0.161	0.147
140	-0.099	-0.001	0.02	-0.008	0.227	0.213	0.299	0.211	0.203	0.202	0.23	0.218	0.241
141	-0.061	0.097	-0.02	0.006	-0.122	-0.109	-0.097	-0.086	-0.064	-0.087	-0.068	-0.099	-0.062
142	-0.087	-0.089	-0.037	-0.005	0.177	0.156	0.233	0.184	0.156	0.226	0.219	0.212	0.155
143	-0.082	0.144	-0.005	-0.003	0.044	0.046	0.067	0.049	0.057	0.052	0.059	0.043	0.018
144	-0.111	0.021	0.028	0.004	-0.115	-0.092	-0.073	-0.137	-0.094	-0.087	-0.08	-0.114	-0.088
145	-0.087	-0.144	-0.018	-0.007	0.262	0.258	0.177	0.278	0.267	0.254	0.292	0.251	0.305
146	-0.09	-0.08	0.046	0.004	-0.055	-0.038	0.02	-0.001	-0.025	-0.037	-0.028	-0.04	-0.021
147	0.009	-0.322	0.009	-0.009	0.295	0.316	0.308	0.356	0.302	0.327	0.313	0.276	0.293
148	-0.034	-0.136	-0.023	0.004	-0.08	-0.012	-0.128	-0.043	-0.062	-0.065	-0.089	-0.071	-0.031
149	-0.003	0.007	0.018	-0.002	0.033	0.04	0.114	-0.011	0.014	0.056	0.066	0.028	0.063
150	-0.013	0.012	-0.176	0.011	-0.241	-0.316	-0.288	-0.288	-0.299	-0.269	-0.365	-0.312	-0.278
151	-0.023	0.168	0.035	0.006	-0.207	-0.17	-0.071	-0.219	-0.193	-0.206	-0.209	-0.156	-0.186
152	-0.038	0.158	0.043	0	-0.04	0.018	-0.092	-0.01	-0.068	-0.032	-0.007	0.011	0.028
153	-0.062	-0.019	-0.1	0.009	-0.15	-0.185	-0.032	-0.146	-0.186	-0.099	-0.156	-0.109	-0.112
154	-0.067	-0.093	-0.028	-0.01	0.284	0.27	0.417	0.273	0.316	0.316	0.365	0.309	0.312
155	0.008	-0.09	0.021	-0.003	0.01	0.044	0.138	-0.026	0.083	-0.015	0.029	-0.025	0.054
156	0.011	-0.102	-0.044	-0.014	0.361	0.414	0.226	0.537	0.452	0.381	0.346	0.399	0.312
157	-0.036	-0.146	-0.193	-0.005	0.168	0.229	0.261	0.209	0.19	0.221	0.202	0.209	0.223
158	0.03	0.118	0.016	0.004	-0.152	-0.117	-0.41	-0.218	-0.192	-0.284	-0.308	-0.28	-0.141
159	-0.057	-0.066	0.03	0.003	-0.186	-0.126	-0.265	-0.07	-0.136	-0.13	-0.145	-0.141	-0.053
160	-0.055	-0.074	-0.079	0	-0.034	0.055	-0.012	-0.037	-0.009	0.041	0.002	-0.056	0.104

161	-0.091	0.16	-0.084	-0.01	0.239	0.19	0.185	0.097	0.255	0.201	0.167	0.182	0.165
162	-0.046	-0.118	-0.137	-0.01	0.189	0.227	0.297	0.126	0.301	0.16	0.267	0.173	0.244
163	-0.032	-0.131	0.008	-0.008	0.228	0.206	0.146	0.152	0.134	0.133	0.224	0.078	0.097
164	-0.053	0.291	-0.09	0.001	-0.024	-0.064	-0.226	-0.232	-0.101	-0.135	-0.084	-0.035	-0.091
165	-0.016	0.525	-0.392	0.01	-0.445	-0.393	-0.441	-0.593	-0.457	-0.495	-0.443	-0.486	-0.399
166	-0.066	0.226	-0.018	0.009	-0.287	-0.236	-0.095	-0.316	-0.205	-0.291	-0.26	-0.247	-0.192
167	-0.069	0.646	-0.034	0.007	-0.357	-0.269	-0.523	-0.416	-0.282	-0.369	-0.287	-0.517	-0.176
168	-0.019	0.458	-0.06	0.003	-0.105	-0.104	-0.208	-0.177	-0.113	-0.157	-0.09	-0.069	-0.117
169	0.037	0.37	-0.07	0.009	-0.337	-0.302	-0.351	-0.274	-0.3	-0.321	-0.308	-0.332	-0.312
170	-0.063	0.389	-0.016	0.006	-0.192	-0.148	-0.152	-0.211	-0.172	-0.18	-0.181	-0.164	-0.138
171	-0.038	-0.207	-0.038	0.003	-0.031	-0.009	0.023	0.025	-0.018	-0.023	-0.021	-0.03	-0.019
172	-0.005	0.318	-0.005	0.021	-0.502	-0.507	-0.542	-0.473	-0.463	-0.49	-0.514	-0.501	-0.479
173	-0.006	0.22	-0.017	-0.006	0.129	0.111	0.129	0.112	0.122	0.127	0.169	0.116	0.163
174	-0.019	0.073	-0.013	-0.002	0.012	0.048	0.007	0.058	0.033	0.002	-0.022	0.002	0.01
175	-0.021	0.306	0.002	0.004	-0.121	-0.114	-0.204	-0.137	-0.124	-0.122	-0.094	-0.12	-0.107
176	-0.078	0.362	-0.012	0.002	-0.008	-0.034	0.003	0.029	-0.026	0.017	-0.026	0.029	0.022
177	-0.044	0.328	-0.031	0.003	-0.058	-0.091	-0.11	-0.048	-0.135	-0.075	-0.071	-0.058	-0.044
178	-0.012	-0.181	0.01	-0.004	0.088	0.087	0.032	0.096	0.056	0.081	0.083	0.086	0.078
179	-0.076	-0.083	-0.014	0	-0.06	0.015	0.035	0.026	0.037	0.004	-0.012	0.007	0.072
180	-0.035	0.295	-0.002	0.006	-0.142	-0.148	-0.119	-0.141	-0.171	-0.137	-0.146	-0.116	-0.121
181	-0.017	-0.342	0.005	-0.003	0.12	0.08	0.108	0.07	0.111	0.089	0.096	0.098	0.127
182	-0.041	0.475	0.013	0	-0.006	-0.01	-0.001	-0.04	0.022	-0.003	0.034	-0.004	-0.022
183	-0.016	0.16	0.013	0.002	-0.074	-0.053	-0.098	-0.066	-0.06	-0.028	-0.019	-0.031	-0.032
184	-0.038	0.016	-0.037	-0.01	0.266	0.247	0.364	0.256	0.189	0.271	0.258	0.262	0.256
185	0.009	0.198	-0.072	0	0.037	0.034	0.061	0.038	0.026	0.054	0.044	0.058	-0.019
186	-0.066	0.105	-0.042	0.009	-0.208	-0.229	-0.156	-0.148	-0.211	-0.17	-0.137	-0.177	-0.147
187	-0.041	0.02	-0.01	-0.005	0.168	0.142	0.22	0.167	0.165	0.171	0.154	0.161	0.163
188	-0.091	0.216	0.019	-0.007	0.155	0.188	0.183	0.192	0.187	0.189	0.129	0.195	0.194
189	-0.051	0.261	-0.036	0.005	-0.117	-0.107	-0.145	-0.094	-0.064	-0.13	-0.105	-0.102	-0.094
190	-0.066	-0.022	-0.045	-0.006	0.173	0.201	0.166	0.216	0.218	0.164	0.179	0.16	0.182
191	-0.058	-0.126	0.024	-0.003	0.094	0.084	0.09	0.116	0.091	0.104	0.077	0.09	0.09
192	-0.061	0.132	-0.04	-0.003	0.051	0.086	0.095	0.071	0.057	0.028	0.051	0.054	0.077
193	-0.027	-0.243	0.029	-0.007	0.28	0.285	0.217	0.263	0.269	0.255	0.231	0.23	0.307
194	0.001	-0.216	-0.063	0.003	-0.034	-0.023	-0.172	-0.085	-0.04	-0.076	-0.042	-0.095	-0.044
195	-0.023	0.255	-0.079	0.005	-0.067	-0.116	-0.031	-0.082	-0.149	-0.087	-0.055	-0.063	-0.066
196	-0.041	0.017	0.018	0.006	-0.113	-0.128	-0.117	-0.15	-0.128	-0.118	-0.138	-0.093	-0.118
197	0.035	-0.133	0.025	-0.003	0.152	0.106	0.078	0.124	0.121	0.107	0.078	0.13	0.104
198	-0.045	0.148	-0.067	-0.002	0.099	0.035	0.074	0.057	0.047	0.099	0.129	0.122	0.02
199	-0.076	0.185	-0.032	0.005	-0.073	-0.126	-0.105	-0.087	-0.119	-0.096	-0.105	-0.123	-0.113
200	-0.013	-0.095	0.002	-0.002	0.109	0.106	0.02	0.137	0.101	0.089	0.09	0.093	0.054
201	-0.066	0.05	0.065	0	0.004	0	0.023	0.043	0.019	-0.004	-0.016	0.021	0.022

202	-0.069	-0.15	-0.04	-0.007	0.203	0.226	0.086	0.168	0.186	0.154	0.141	0.155	0.2
203	-0.035	0.008	-0.051	-0.006	0.167	0.196	0.181	0.161	0.198	0.17	0.164	0.196	0.181
204	0.002	-0.034	-0.022	0.002	-0.026	-0.014	0.014	-0.057	-0.027	-0.018	-0.032	-0.05	-0.015
205	-0.015	0.34	0.035	0.006	-0.142	-0.203	-0.224	-0.145	-0.171	-0.183	-0.162	-0.185	-0.174
206	-0.01	0.108	0.028	-0.002	0.037	0.05	0.033	0.04	0.074	0.057	0.005	0.044	0.068
207	-0.027	0.035	0.014	-0.003	0.069	0.054	0.031	0.073	0.079	0.089	0.042	0.073	0.069
208	-0.014	0.051	0	-0.003	0.083	0.089	0.141	0.053	0.083	0.086	0.079	0.092	0.024
209	0.041	0.03	-0.026	-0.006	0.125	0.108	0.121	0.129	0.126	0.141	0.139	0.156	0.093

APPENDIX E: ASSET CLASS INDEXES USED IN CHAPTER 2

Asset Class	**Historical Index**
U.S. Large & Mid Cap Equity	MSCI USA TR Index
U.S. Small Cap Equity	MSCI USA Small Cap TR Index
Developed ex-U.S. Equity UH	MSCI World ex USA Index (USD)
EM Equity UH	MSCI Emerging Markets Index Net (USD)
Global Equity UH	MSCI AC World TR Index (USD)
Global Commodities UH	Bloomberg/DJ Commodity TR Index (USD)
Global Infrastructure UH	S&P Global Infrastructure TR Index (USD)
Developed REITs UH	FTSE EPRA/NAREIT Developed Real Estate TR Index (USD)
U.S. Core Fixed Income	Barclays U.S. Aggregate Bond Index
Global Credit H	Barclays Global Aggregate Bond Index Hedged (USD)
U.S. Short Govt	BofAML 1-3 Year US Treasury Index TR
U.S. Short Credit	Barclays US Credit 1-3 Year Index TR
U.S. Long Govt/Credit	Barclays U.S. Long Government/Credit USD
U.S. TIPS	Barclays U.S. TIPS Index TR
Cash	BofAML U.S. 3-Month T-Bill Index TR
U.S. High Yield	BofAML US HY Master II TR
Global High Yield H	BofAML Global High Yield 2% Constrained Index TR (USDH)
EMD (hard)	JPM EMBI Global Diversified Composite Index TR (USD)
Bank Loans	S&P Leveraged Loan 100 Index TR
Global Convertibles H	TR CV GLOBAL HEDGED (USD) - TOT RETURN IND
Conscious Currency	Russell Conscious Currency Index TR (USD)

APPENDIX F: FIT STATISTICS FOR ESTIMATED PEER AVERGE FACTOR EXPOSURES IN CHAPTER 2

```
Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)             -1.511e-04  7.571e-05  -1.995   0.0479 *
XFGlobal.Equity.UH.1     3.878e-01  6.540e-03  59.295  < 2e-16 ***
XFUS.Size.1              6.357e-02  8.949e-03   7.104 5.04e-11 ***
XFExcess.Dev..ex.US.UH  -1.342e-01  1.239e-02 -10.831  < 2e-16 ***
XFExcess.EM.UH          -3.521e-02  6.322e-03  -5.570 1.20e-07 ***
XFCommodity              4.186e-02  4.947e-03   8.461 2.63e-14 ***
XFExcess.Global.Infra    3.781e-02  8.935e-03   4.231 4.11e-05 ***
XFDuration.1             5.152e-01  5.110e-02  10.081  < 2e-16 ***
XFTerm                  -3.774e-02  1.743e-02  -2.166   0.0320 *
XFExcess.Credit.H        4.160e-01  4.569e-02   9.104 6.31e-16 ***
XFConscious.Currency.1  -8.523e-03  1.449e-02  -0.588   0.5574
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0005761 on 145 degrees of freedom
Multiple R-squared:  0.9822,	Adjusted R-squared:  0.981
F-statistic: 800.2 on 10 and 145 DF,  p-value: < 2.2e-16
```

F 1: Summary statistics from the factor model fitted for 15-30% equity peer group

```
Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)             -1.511e-04  7.571e-05  -1.995   0.0479 *
XFGlobal.Equity.UH.1     3.878e-01  6.540e-03  59.295  < 2e-16 ***
XFUS.Size.1              6.357e-02  8.949e-03   7.104 5.04e-11 ***
XFExcess.Dev..ex.US.UH  -1.342e-01  1.239e-02 -10.831  < 2e-16 ***
XFExcess.EM.UH          -3.521e-02  6.322e-03  -5.570 1.20e-07 ***
XFCommodity              4.186e-02  4.947e-03   8.461 2.63e-14 ***
XFExcess.Global.Infra    3.781e-02  8.935e-03   4.231 4.11e-05 ***
XFDuration.1             5.152e-01  5.110e-02  10.081  < 2e-16 ***
XFTerm                  -3.774e-02  1.743e-02  -2.166   0.0320 *
XFExcess.Credit.H        4.160e-01  4.569e-02   9.104 6.31e-16 ***
XFConscious.Currency.1  -8.523e-03  1.449e-02  -0.588   0.5574
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0005761 on 145 degrees of freedom
Multiple R-squared:  0.9822,	Adjusted R-squared:  0.981
F-statistic: 800.2 on 10 and 145 DF,  p-value: < 2.2e-16
```

F 2: Summary statistics from the factor model fitted for 30-50% equity peer group

```
Coefficients:
                        Estimate Std. Error t value Pr(>|t|)    
(Intercept)           -1.786e-04  6.288e-05  -2.841 0.005147 ** 
XFGlobal.Equity.UH.1   6.177e-01  5.454e-03 113.249  < 2e-16 ***
XFUS.Size.1            8.013e-02  7.462e-03  10.739  < 2e-16 ***
XFExcess.Dev..ex.US.UH -2.797e-01  1.040e-02 -26.887  < 2e-16 ***
XFExcess.EM.UH        -7.759e-02  5.129e-03 -15.127  < 2e-16 ***
XFCommodity            2.878e-02  4.127e-03   6.972    1e-10 ***
XFExcess.Global.Infra  1.227e-02  7.452e-03   1.647 0.101798    
XFDuration.1           2.251e-01  1.733e-02  12.990  < 2e-16 ***
XFExcess.Credit.H      1.413e-01  3.737e-02   3.781 0.000227 ***
XFConscious.Currency.1 -5.714e-03  1.215e-02  -0.470 0.638840    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.000484 on 146 degrees of freedom
Multiple R-squared:  0.9941,	Adjusted R-squared:  0.9937 
F-statistic:  2733 on 9 and 146 DF,  p-value: < 2.2e-16
```

F 3: Summary statistics from the factor model fitted for 50-70% equity peer group

```
Coefficients:
                        Estimate Std. Error t value Pr(>|t|)    
(Intercept)           -2.736e-04  8.247e-05  -3.317  0.00115 ** 
XFGlobal.Equity.UH.1   7.966e-01  6.432e-03 123.855  < 2e-16 ***
XFUS.Size.1            1.139e-01  1.006e-02  11.323  < 2e-16 ***
XFExcess.Dev..ex.US.UH -3.084e-01  1.405e-02 -21.951  < 2e-16 ***
XFExcess.EM.UH        -8.318e-02  6.861e-03 -12.123  < 2e-16 ***
XFCommodity            3.023e-02  5.578e-03   5.420 2.39e-07 ***
XFExcess.Global.Infra  3.033e-02  1.006e-02   3.015  0.00303 ** 
XFDuration.1           1.019e-01  1.854e-02   5.494 1.69e-07 ***
XFConscious.Currency.1 -1.426e-02  1.634e-02  -0.873  0.38412    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0006541 on 147 degrees of freedom
Multiple R-squared:  0.9933,	Adjusted R-squared:  0.993 
F-statistic:  2744 on 8 and 147 DF,  p-value: < 2.2e-16
```

F 4: Summary statistics from the factor model fitted for 70-85% equity peer group

```
Coefficients:
                         Estimate Std. Error t value Pr(>|t|)    
(Intercept)            -0.0002001  0.0001010  -1.982   0.0494 *  
XFGlobal.Equity.UH.1    0.9357362  0.0074809 125.084   <2e-16 ***
XFUS.Size.1             0.1412192  0.0118894  11.878   <2e-16 ***
XFExcess.Dev..ex.US.UH -0.3369845  0.0164240 -20.518   <2e-16 ***
XFExcess.EM.UH         -0.1039115  0.0081652 -12.726   <2e-16 ***
XFCommodity             0.0132566  0.0065419   2.026   0.0445 *  
XFConscious.Currency.1 -0.0290025  0.0200684  -1.445   0.1505    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0008052 on 149 degrees of freedom
Multiple R-squared:  0.9928,	Adjusted R-squared:  0.9925 
F-statistic:  3411 on 6 and 149 DF,  p-value: < 2.2e-16
```

F 5: Summary statistics from the factor model fitted for 85-100% equity peer group

VITA

Sangeetha Srinivasan was born in a small town in south India and finished her high school education in Chennai, Tamil Nadu, India, before pursuing a B Tech in Electrical & Electronics Engineering at Pondicherry Engineering College, Pondicherry, India. In Augist 2005, she came to the U.S. with the goal of pursuing graduate studies in Mathematics at Auburn University, Auburn, specializing in Discrete Mathematics (Deign theory & Graph theory), graduating with an MS in Math in December 2007. A personal situation brought her to Seattle and the ongoing financial crisis inspired her to pursue a PhD in Economics at the University of Washington, Seattle. During the course of the PhD program, Sangeetha obtained an MA in Economics and a graduate certificate in Computational Finance & Risk Management. The latter provided the foundation and opportunity (particularly, Google Summer of Code 2014, 2015) to develop an R package for fitting factor models for asset returns in collaboration with other students and faculty members, and helped her identify and chose a research topic for this dissertation. In mid-2015, Sangeetha joined the retail multi-asset team at Russell Investments, Seattle and had the opportunity to develop a multi-asset factor model used for portfolio risk decompositions among other projects in equity return forecasting and portfolio design, regime-based asset allocation etc. After graduating from UW, Seattle in June 2018, she plans to continue working at Russell Investments in the role of an Asset Allocation Strategist.