Monopoly Money

Market Concentration and Equity Returns in the United States

Christopher Babcock, Eric Mosher, Santiago J. Olalquiaga C., Arne Severijns

April 29, 2022

Abstract

This paper explores the relationship between market concentration and equity returns. Most current literature predicts that firms in less concentrated markets will earn higher returns. We test this hypothesis using stock returns data from 13,001 firms between 2001 and 2020. We use the Fama-French three-factor model and the Carhart four-factor model to test our findings and several robustness checks. Contrary to several previous studies, including the widely cited Hou and Robinson (2006), we find a statistically significant correlation between higher market concentration and excess equity returns (also known as "alpha"). However, when we expand our analysis to examine the period from 1987 to 2020, we find results that confirm Hou and Robinson's findings – but only for the period until the early 2000s. These findings suggest that the dynamic between market concentration and portfolio returns has shifted significantly since Hou and Robinson conducted their study, coinciding with a period of weakening competition in the US economy.

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

1 Introduction

Over the two decades from 2000 to 2020, a recurring theme in the business media and economic research has been the growing concentration of many industries in the United States. The National Bureau of Economic Research (NBER) finds that business concentration, profit margins, and market power increased in most U.S. industries from 1999 to 2019 (Philippon, 2019). Abdela and Steinbaum (2018) note that the number of mergers per year in the United States increased from 2,308 in 1985 to 15,361 in 2017, suggesting increased market concentration and monopoly power. Grullon, Larkin, and Michaely (2017) report that 75% of industries in the United States grew more concentrated between 1997 and 2017. Stiglitz (2019) remarks that in most sectors of the U.S. economy, just a few firms now dominate 75-90% of the market.

What are the effects of market concentration on investment returns?

In the 1970s, academic research in industrial organization (IO) has explored the link between market organization and equity pricing. Most IO theory suggests that industry concentration and equity returns should be negatively correlated. As a result, firms in less concentrated industries should offer a "competition premium" in the form of higher stock returns to induce people to invest.

The seminal work exploring this relationship came in 2006 by Hou & Robinson. Their research found a statistically significant positive relationship between lower market concentration – measured by Hirschman-Herfindahl Index (HHI) – and equity returns. This paper adds to existing research on the relationship between market concentration and equity pricing. We first evaluate the returns of companies in the most concentrated and least concentrated industries from 2000 to 2020. We find a positive relationship between higher market concentration and excess stock returns, or alpha, which runs counter to the findings of Hou and Robinson.

However, when we begin our analysis in 1987, covering much of the time studied by Hou and Robinson, we find that lower market concentration is associated with higher stock returns – but only until the early 2000s. We conclude that this change in the relationship between market concentration and returns may be due to broader shifts in the makeup of the United States economy.

The following section includes a brief literature review outlining the existing research. In Section 3, we describe our data cleaning process and analysis methodology. Section 4 details our model and estimation approach, using the Fama-French three-factor model and the Carhart four-factor model.

We layout our findings and recommendations for future research in Section 5. Finally, Section 6 offers our concluding thoughts and comments.

2 Literature Review

The majority of prior research establishes that highly concentrated industries exhibit a reduced incentive to innovate. They are already enjoying the excess profits accrued by their relatively large market power and face relatively little threat from new entrants into the market. These two traits make the stocks of firms in highly concentrated industries assets with relatively low risk, so they command a relatively low risk premium. On the flip side, firms operating in industries with little concentration are forced to invest in innovation to differentiate themselves from the competition in the hopes of attaining excess profits, such as engaging in research and development (R&D) and, simultaneously, are relatively more threatened by new entrants into their market. Taken together, these traits mean that stocks of firms from industries with little concentration are relatively riskier and, thus, should command higher premiums – i.e., they should yield relatively higher returns.

The most common measure of market concentration is the Herfindahl-Hirschman Index (HHI). A higher HHI indicates higher market concentration, whereas the opposite suggests a more competitive environment. In the extreme case, an HHI of 1 is characteristic of an industry-wide monopoly, and an HHI of 0 reflects perfect competition.

A comprehensive literature exists exploring the relationship between equity returns and market structure, with notable contributions by Aivazian and Callen (1979), Booth (1981), Conine Jr. (1983), and Lee, Liaw, and Rahman (1990). These researchers largely posit a negative link between industry concentration and equity returns. Additional work from the industrialorganizational (IO) field also supports the idea of a competition premium, which established investors' demand for higher returns from companies facing greater competition (less concentrated markets) than from those facing reduced competition (more concentrated markets).

The most pivotal work investigating this relationship is from Hou and Robinson (2006). The authors hypothesize that firms in more concentrated industries, or ones with a high HHI, earn lower returns than those in less concentrated industries where competition is more intense. Using stock returns data from 1963 through 2001, their data and analysis show that firms in the highest quintile

of competitive industries earn an annual return 4 % higher than firms in the lowest quintile (most concentrated industries). The risk premium between these two quintiles grows when the economy is in recession. As a result, they present convincing evidence that firms in more concentrated industries earn lower returns, even after controlling for size, book-to-market, and momentum.

The work by Hou and Robinson (2006) inspired several researchers to investigate further the relationship between market concentration and alpha generation. Much of the subsequent research has supported Hou and Robinson's results.

Datta and Chakrabotty (2018) directed the Hou and Robinson model to the Indian stock market. They found a similar negative relationship between market concentration and stock returns at both the industry and firm level while also reporting that the Fama-French three-factor model was not sufficient to explain these abnormal returns.

However, not all research has supported these general findings. An initial objection was written by Sharma (2010), who explored other elements of product market structure beyond the HHI. He argues that market competition cannot be captured by industry concentration alone, and there is, in fact, a multidimensional relationship between product competition and equity returns.

Perhaps most notably, Bustamente and Donangelo (2014) argue that the correlation between market competition and equity returns is negative. Their argument rests on the procyclical nature of the entry threat on firm values. In other words, expected returns are reduced during periods of economic expansion when competitors enter the market more readily, thus negatively affecting the value of incumbent firms. Hence, their research suggests that higher market concentration may lead to higher returns, different from the findings of many other studies on this topic.

More recently, Applin (2019) also revisited the idea of a competition premium pioneered by Hou and Robinson. Interestingly, this paper finds that the premium for investing in less concentrated markets becomes statistically insignificant when taking the 2002-2018 period into account, thus putting the robustness of the competition premium into question. Instead, Applin presents evidence that these excess returns may behave parabolically relative to the degree of market concentration.

These later contributions by Bustamente and Donangelo, and Applin mark a shift in the literature and suggest that the question of how market concentration affects stock returns still lacks a definitive answer.

With this in mind, we craft our approach to exploring the impact of market concentration on

equity returns. We build on Hou and Robinson's work by extending the period examined (2000 through 2020), similar to Applin's work. Subsequently, we extend our analysis backward to explore the period from 1986 through 2020 to see if a longer time frame affects our results. We also repeat Hou and Robinson's use of the HHI index based on market share defined by sales. Our work shows a swift reversal of the dynamic between market concentration and equity returns around 2000, which marks the onset of a period with increasingly negative competition premia.

3 Methodology

3.a Data source

We obtain data from the COMPUSTAT and CRSP databases in WRDS, over years 2000 to 2020. We choose a 20-year period that includes more than one recession and several periods of prolonged economic growth.

3.b Cleaning and merging data

To organize our data, we first set up a table to link Compustat Fundamental (COMPUSTAT / COMPFUNDA) and Center for Research in Security Prices (CRSP) data via company and security identifiers. The global company key (gvkey) and permanent security identification number (permno) variables, are contained in both databases. To merge the two databases, we employ the CRSP CCMXPF_LNKHIST table, which contains historical linking information that has been used to merge COMPUSTAT and CRSP data in the past, following a methodology presented by Mingze Gao.

From the COMPUSTAT database in Wharton Research Data Services (WRDS) we obtain annual company fundamentals data, including global company key, company name, fiscal year end, sales data, CUSIP number, and North American Identity Classification System (NAICS) classification numbers. The NAICS and sales data are the inputs we use to construct a measure of industry concentration, upon which to build our long/short portfolios. We exclude data where observations for the NAICS or sales variables are missing.

For companies with NAICS classification codes greater than 3 digits in length, we truncate the code after the third digit. Doing this creates 95 unique 3-digit NAICS codes, which we later

use as the defining variable for industry groupings, to calculate market concentration and sort our portfolios. We choose to use NAICS codes rather than Standard Industrial Classification (SIC) codes because NAICS offer a greater level of detail than SIC, and because the US government stopped assigning SIC codes after 2004. By using NAICS we expect to find a more robust data set that better reflects the real changes in the economy during the period from 2000 to 2020.

From the CRSP database, we extract data from the CRSP Monthly Stock File (MSF). We merge CRSP data with the COMPUSTAT data outlined above using SQL join commands. Since CRSP data are monthly and COMPUSTAT data are annual, we replicate each COMPUSTAT data row 12 times to link with the monthly data from CRSP.

In total, we created a merged table with 1,307,593 rows with data from 13,001 companies and 95 distinct industries defined by NAICS codes.

3.c Calculating the Hirschman-Herfindahl Index (HHI)

As a measure of market concentration, we calculate the Hirschman-Herfindahl Index (HHI). The HHI is an established measure of market concentration that has been extensively used in the US by the Department of Justice in issues related to the competitive effects of mergers and acquisitions (Rhoades, 1993).

The HHI is defined as the sum of squared market shares of the firms within a given market:

$$\text{Herfindahl}_j = \sum_{i=1}^{I} s_{ij}^2$$

where s_{ij} is the market share of firm i in industry j.

We use the NAICS classification codes at the three-digit level as our definitions for markets.

We use SQL commands to calculate the HHI for each industry, based on sales. First, we create a new variable, Ind_sales, that sums sales data across all firms in an industry in any given fiscal year, where the industry is defined by a 3-digit NAICS code. The resulting variable has the same figure for each industry in each year. Using Ind_sale, we calculate the market share for each company by dividing their yearly individual company sales by the Ind_sale value associated with its 3-digit NAICS code, as calculated above. We then follow the HHI formula: we square each company's individual market share, and then add up the squared company market shares for each 3-digit

NAICS industry code, for every fiscal year. The resulting HHI values represent the annual market concentration of each industry, based on company sales as a percentage of total industry sales. During our robustness checks, we also modeled HHI through equity and assets instead of sales. As Hou and Robinson found, these variables are correlated with sales. Equity performed worse than sales over our baseline test, but assets actually improved our alpha.

3.d Final cleaning of data

Because of duplication in our dataset, we use the PROC NODUPKEY command to remove duplicate values. We also check CRSP.MSF to find the full possible set of returns data from 2000 to 2020, to make sure we do not have more data than possible. CRSP.MSF contains 1.7 million entries from 2000 to 2020. In our merged dataset we have 1.3 million, which is below the upper limit in CRSP.

3.e Sorting the data to construct long/short portfolios

We construct our portfolios by sorting the most and least concentrated industries. We use PROC SQL commands to rank industries by the degree to which they are concentrated, as approximated by each industry's HHI value for each calendar year. Our resulting ranking is in decimal values from 0 to 1, for ease of use in further classifications of the data, where low positions in the ranking (i.e., ranking values close to zero) correspond to low HHI values (signaling little market concentration), and high positions in the ranking (i.e., ranking values close to one) correspond to high HHI values (indicating high degrees of market concentration).

We then use these rankings to create our market concentration-sorted portfolios and compare their returns. To build the portfolios, we take the firms from the 20%-most concentrated and 20%-least concentrated industries in our sample, as ranked above. Identifying the portfolios based on the highest and lowest quintiles follows the methodology of Hou and Robinson (2006). We construct the portfolios as follows:

For our low-market concentration portfolio, we group together companies from industries that exhibit little market concentration (i.e., highly competed industries that have low HHI values). We set the cut-off for low-concentration industries we include in our portfolio as companies with an HHI ranking between 0 and 0.2. In order to identify this group of firms, we define a new variable

("index") and assign them an index value of 1.

For our high-market concentration portfolio, we group together companies from industries that exhibit significant market concentration (i.e. industries with few participants that have high HHI values). We set the cut-off for high-concentration industries we include in our portfolio as companies with an HHI ranking between 0.8 and 1. In order to identify this group of firms, we assign them an index value of 2.

Companies in the low-market concentration portfolio have an average HHI value of 0.4%, while companies in the high-market concentration portfolio have an average HHI value of 1.7%. HHI values for the sample range from 0.1% to 2.5%.

During our robustness checks we altered the portfolio weights to top and bottom 10% instead of the top and bottom 20% quintiles. The 10% portfolio performed worse. We believe this is due to having less of both the "good" and "bad" firms in our overall portfolio. To employ our strategy in practice, optimizing the weighting would be a key step to building the best portfolio.

3.f Lagging HHI data

The HHI information is contemporaneous to stock prices, so in order to construct portfolios that are realistic, we need to do the sorting based on HHI value rankings from the preceding year. If an investor were creating a portfolio based on the most and least concentrated industries in year t=1, he would need to look at market data from the previous year, t=0. To mimic this, we lag the HHI and rank variables by 12 months.

3.g Calculating cumulative returns

To assess how well our long and short portfolios perform over the two decades we study, we calculate value-weighted monthly returns for the portfolios, where the weights are the market capitalizations of the firms in our portfolios.

We first calculate the mean monthly returns of each portfolio, where the mean is weighted by the market capitalization of every asset in the portfolio twelve months prior (when the portfolio would have been constructed).

We then compute the cumulative returns over time, using a "do loop." Additionally, we compute the difference in returns between the high-industry concentration and low-industry concentration

portfolios over the sample.

3.h Extending the sample period

Following the construction of the portfolios, calculation of returns, and estimation of the model, we extend the time coverage in our sample backward to encompass the years from 1986 to 2020. In doing so, we partially cover the sample period studied by Hou and Robinson. We conduct this extension as a robustness check of our results, to compare our results in-sample to Hou and Robinson's results, and to get better statistical estimates through the use of bigger samples.

4 Model and Estimation Approach

We analyze our final dataset using the Fama-French three-factor model, introduced by Eugene Fama and Kenneth French in 1992.

$$RetDiff_t = \alpha + \beta_1 mktrf_t + \beta_2 smb_t + \beta_3 hml_t$$

The model uses three factors to estimate stock returns: market returns (*mktrf*), the outperformance of small-capitalization companies versus large-capitalization companies (*smb*), and the outperformance of high book-to-market companies over low book-to-market companies (*htl*). The intercept α represents the excess returns over the returns predicted by the three factors used in the model: *mktrf*, *smb*, and *htl*.

As part of our robustness check of our results, we also employ the Carhart (1997) four-factor model. This model builds on the Fama-French three-factor model by adding a fourth factor, momentum (*umd*), into the regression equation. In this context, momentum refers to the rate of price change of a stock.

$$RetDiff_t = \alpha + \beta_1 mktrf_t + \beta_2 smb_t + \beta_3 hml_t + \beta_4 umd_t$$

As described above, our proposed investment strategy sorts companies into two portfolios: one with companies with the highest quintile of market concentration, and a second with the

lowest quintile of market concentration. Our proposed strategy is to buy stocks from the higher-concentration portfolio and sell stocks from the lower-concentration portfolio.

To test our hypothesis, we calculate the monthly difference in cumulative returns between the two portfolios. Ultimately we generate 240 observations, one for each month over the 20-year period. We regress this set of observations against the factors in the Fama-French three-factor model and Carhart four-factor model and test for the significance of the intercept, which will be our result for alpha. We run a standard OLS regression, compute a t-test, and calculate White heteroskedasticity-robust standard errors.

In the following section we describe our results and several robustness checks we perform on our data.

5 Results

5.a Cumulative returns

Overall, we find that the cumulative returns of the most concentrated market portfolio outperformed the cumulative returns of the least concentrated market portfolio by a statistically significant margin, over the 20-year period studied. A $1 investment made in 2000 in the most concentrated portfolio was worth over $5 by 2020, while the same investment in the least concentrated portfolio was worth under $2 after 20 years. (See Figure 1)

5.b Fama-French three-factor estimation results

To check our results against theoretical models of stock performance, we first regress the monthly difference in cumulative returns against the three factors in the Fama-French three-factor model – *mktrf*, *smb*, and *htl* – as described above in Section 4. We find an alpha value of about 0.005, which is statistically significant at the 5% level (p-value = 0.0494).

Of the three passive factors, we found the following coefficients: -0.0003 for *mktrf* (p-value = 0.5844), -0.0007 for *smb* (p-value = 0.4868), and 0.00446 for *hml* (p-value < .0001). Notably, only *hml* – high book-to-market ratio minus low – is found to be statistically significant in our model. (See Figure 2)

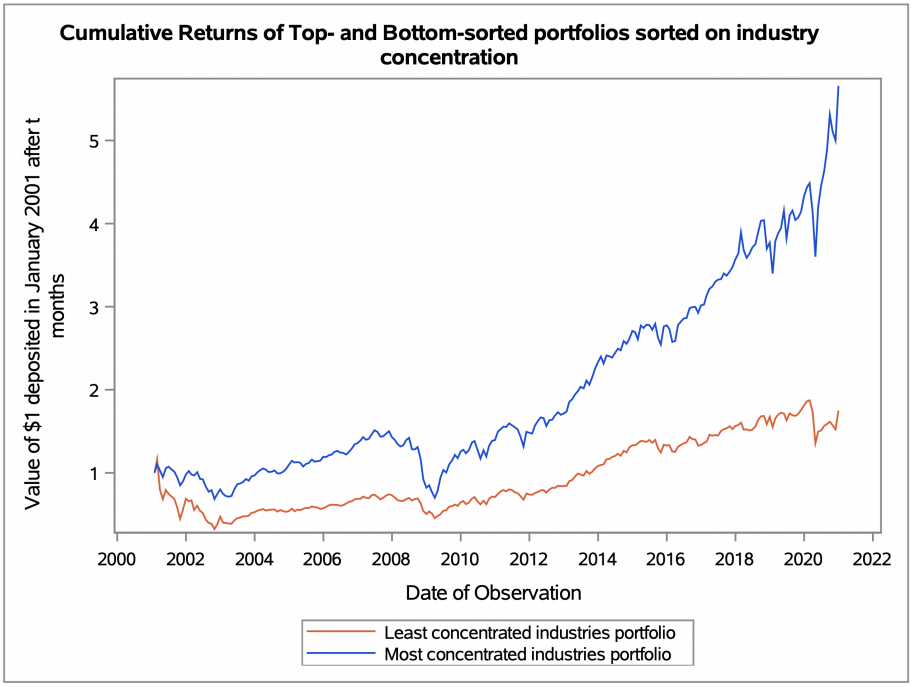

Figure 1: Cumulative Returns of concentration-ranked portfolios

Parameter Estimates					
Variable	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**
Intercept	1	0.00499	0.00253	1.98	0.0494
mktrf	1	-0.00032272	0.00058915	-0.55	0.5844
smb	1	-0.00072183	0.00104	-0.70	0.4868
hml	1	0.00446	0.00087151	5.12	<.0001

Figure 2: Estimates from baseline regression

We note that there may be some overlap between *smb*, which measures returns of small-cap versus large-cap companies, and our sorting variable. We know that markets with very high concentration levels will tend to have larger companies, while highly competitive markets will tend to have smaller companies. Since we design our portfolio based on market concentration, our sorting variable may be aligned with company size and the *smb* variable. We suppose that the lack of statistical significance of the *smb* coefficient may come from this overlap. After correcting for potential heteroskedasticity in the data by calculating White robust standard errors, our alpha estimation is

slightly less significant (p-value = 0.0625), though still significant at the 10% level. (See Figure 3)

Nonlinear OLS Parameter Estimates				
Parameter	Estimate	Approx Std Err	t Value	Approx Pr > |t|
a1	0.004988	0.00266	1.87	0.0625
b1	-0.00032	0.000904	-0.36	0.7215
b2	-0.00072	0.00131	-0.55	0.5834
b3	0.004464	0.00184	2.43	0.0159

Figure 3: Estimates with White robust standard errors

5.c Carhart four-factor estimation results

We develop a secondary regression model using the Carhart four-factor model of stock returns to see whether adding a momentum factor would impact our findings. Encouragingly, our overall result remains the same. In the four-factor model, we identify a alpha value equal to 0.00416, slightly lower than in our three-factor model estimation. Our estimated alpha is statistically significant at the 10% level, with a p-value = 0.0952.

While the statistical significance of our alpha value drops slightly between the three-factor and four-factor model, we find higher significance and higher coefficients for most of the factors: 0.05805 for *mktrf* (p-value = 0.3633); -0.04713 for *smb* (p-value = 0.6437); 0.47380 for *hml* (p-value < 0.0001); and 0.18302 for *umd* (p-value = 0.0010). These results suggest that including momentum generally increases the significance of the other factors in the regression. (See Figure 4)

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	0.00416	0.00249	1.68	0.0952
mktrf	Excess Return on the Market	1	0.05805	0.06373	0.91	0.3633
smb	Small-Minus-Big Return	1	-0.04713	0.10176	-0.46	0.6437
hml	High-Minus-Low Return	1	0.47380	0.08573	5.53	<.0001
umd	Momentum Factor	1	0.18302	0.05489	3.33	0.0010

Figure 4: Estimates from Carhart four-factor estimation

Again, we estimate our model a second time using White robust standard errors, correcting for potential heteroskedasticity in our data. We find that our alpha estimate is slightly less statistically significant (p-value = 0.1243), as would be expected. (See Figure 5)

Nonlinear OLS Parameter Estimates				
Parameter	Estimate	Approx Std Err	t Value	Approx Pr > |t|
a1	0.004164	0.00270	1.54	0.1243
b1	0.058052	0.0808	0.72	0.4733
b2	-0.04713	0.1193	-0.40	0.6932
b3	0.473798	0.1779	2.66	0.0083
b4	0.183019	0.1318	1.39	0.1664

Figure 5: Estimates from Carhart four-factor estimation with White robust standard errors

5.d Extending the time frame to 1987

Realize that the years we examine in our initial research are different from the years studied in Hou and Robinson (2006). Their research examines stock returns from 1963 to 2001, ending precisely when our analysis begins. Obviously, that was a time of dramatic change in the U.S.

economy, from the dot-com bubble bursting in the early part of the decade to the financial crisis and subsequent recovery in the latter part of the 2000s. The two decades leading up to 2020 were also a time of increasing market concentration.

We find that, for the period from 1987 until 2000, the least concentrated industries perform better, and for the period from 2000 until 2020, the most concentrated industries perform better. (See Figure 6 and Figure 7.)

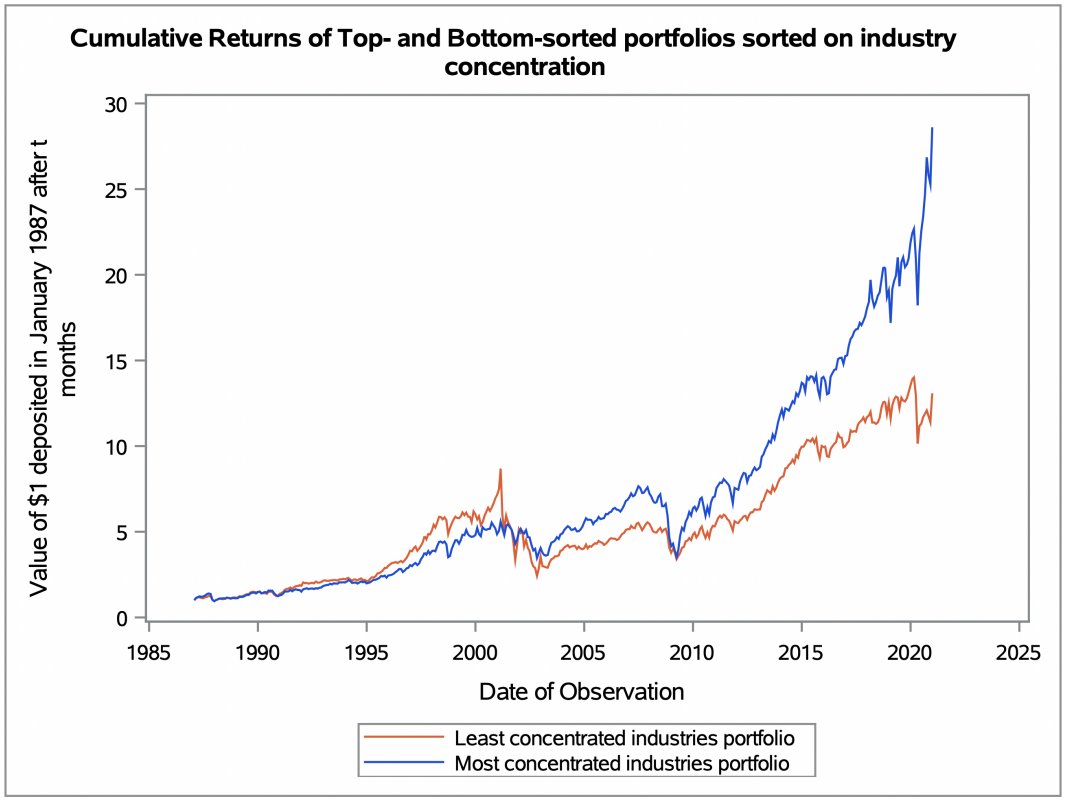

Figure 6: Cumulative Returns of concentration-ranked portfolios from sample extended back to 1986

This finding is both a vindication and rebuttal of Hou and Robinson (2006). Despite our initial discovery that goes against their results, we find that Hou and Robinson's conclusions do hold for the years of their original study when we look further back in time. When we look at the last twenty years, however, their results seem less accurate. This suggests that there has indeed been a shift in market composition in the United States since the beginning of the 21st century. It may have been a winning investment strategy to invest in companies in competitive industries in the second half of the 20th century. But, since 2001, the opposite seems to be true. Hence, while Hou

Parameter Estimates						
Variable	**Label**	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**
Intercept	Intercept	1	-0.00049833	0.00183	-0.27	0.7858
mktrf	Excess Return on the Market	1	0.13308	0.04270	3.12	0.0020
smb	Small-Minus-Big Return	1	0.14624	0.06030	2.43	0.0157
hml	High-Minus-Low Return	1	0.20970	0.06362	3.30	0.0011
umd	Momentum Factor	1	0.18269	0.04141	4.41	<.0001

Figure 7: Estimates from sample extended back to 1986

and Robinson may have been correct in the time context in which they studied, in the broader historical context their conclusions appear to be less firm.

Notably, in our updated model estimation (using the Carhart four-factor model), we do not find a statistically significant value for alpha. We claim that this is due to the shift that happens around 2000. Before that time the alpha value is negative, while after that time it is positive.

5.e Different HHIs

With our robustness checks, we wanted to test various HHIs other than the standard sales to calculate market share and concentration, similar to methods employed by Hou and Robinson. After using assets and equity data from COMPUSTAT and re-running our code, we were able to demonstrate that calculating HHI based on assets actually performed the best out of the three variables we tested to compute market concentration. (See Figure 8.) Hou and Robinson demonstrated that all three of these variables are highly correlated with one another and any can be used as the basis for the HHI.

5.f Recommendations for further research

Given our results, we believe the research on the link between stock returns and market concentration can be extended in at least three different directions.

First, considering the relative performance of high-market concentration and low-market concentration portfolios reversed around 2001–shortly after the dot-com recession – it is vital to analyze

Figure 8: Regression Results from HHIs Calculated using Sales, Assets, and Equity

the underlying cause of this change in performance. Why might the low-market concentration portfolio have outperformed the high-market concentration portfolio until 2000 (in line with Hou and Robinson's results) but then underperformed from 2000 onwards? What could be the underlying phenomena governing the relationship between returns and market concentration?

One reason may lie in the broader trends around market concentration itself. Note that the low-market concentration portfolios' cumulative returns were highest relative to the high-market concentration portfolio at the peak of the dot-com bubble. This could partly reflect a disproportionate impact of technology companies on asset returns, as technology companies were one of the driving sectors during the bubble. In addition, the dot-com technology sector was much less consolidated (and likely much less concentrated) than the technology sector is today. Furthermore, industry concentration has increased across the US economy in the last few decades (Bessen, 2017). Combining these facts could suggest a potential relationship between the relative performance of high- and low-market concentration portfolios and market concentration in the overall economy. However, a more detailed analysis of these dynamics is needed to support this hypothesis.

Another avenue for further research would be to extend this analysis across countries. This could help identify whether the mechanisms at play are unique to the US or pervasive throughout other regions and macroeconomic environments.

To increase the robustness of our study, a moving average HHI could be used instead of a single year. Hou and Robinson used a three-year moving average. We rebalanced our portfolios annually, but using a moving average model may remove some of the variation of industries in and out of the portfolios due to cycles.

Finally, testing alternative configurations of the constructed portfolios could also offer further insights. Our portfolios are weighted by market capitalization, but alternative weightings would impact the cumulative returns we find. Alternative configurations of the portfolios could also explore setting different cut-off points for the percentiles of industries we include in our portfolios (as ranked by market concentration), or the degree of granularity with which we define markets (using, for instance, four-digit instead of three-digit NAICS codes as the delimitations of markets whose concentrations we assess).

6 Conclusion

This paper uses a three- and four-factor model to test the relationship between investing in highly concentrated industries and earning abnormal excess returns. Overall, we find that the relationship between market concentration and equity returns is more nuanced than the existing literature would suggest. While our findings agree with Hou and Robinson for the period from 1987 to 2001, but the environment has changed since then. Beginning in 2001, organizations in highly concentrated industries produced higher stock returns, a stark reversal of the relation observed by Hou and Robinson from 1963 to 2001.

We contend that this change in investment returns mirrors a broader change in the makeup of the United States economy. While theory from the late 20th century suggested that firms in competitive markets should offer investors a "competition premium," we find that things may be different in practice. As the U.S. economy has grown more concentrated over the last 20 to 30 years, it seems to have become more profitable to invest in firms that operate in less competitive markets.

Our findings are most similar to those of Applin (2019), who extended the analysis of Hou and Robinson beyond 2001 to include data for the years up to 2018. Applin finds that the relationship observed by Hou and Robinson begins to break down after 2002 when there is no longer a statistically significant link between less market concentration and consistent alpha generation.

As markets continue to evolve in the future, it will be necessary for investors to understand the interaction between market concentration and company performance, including equity returns. We hope our findings will help add to the ongoing discussion around how monopolies, competition, industrial organization, and investment strategies interact.

References

Adil, A., Marshall, S. (2018, September 11). The United States has a Market Concentration Problem [Review of The United States has a Market Concentration Problem]. Roosevelt Institute. https://rooseveltinstitute. org/publications/united-states-market-concentration-problem/

Applin, R. (2019). Industry Concentration and Average Stock Returns Revisited. https://people.wku.edu /david.zimmer/index_files/applin.pdf

Bessen, J. (2017). Information Technology and Industry Concentration Information Technology and Industry Concentration. https://scholarship.law.bu.edu/cgi/viewcontent.cgi?article=1269context=faculty_scholarship

Bustamante, M. C., Donangelo, A. (2017). Product Market Competition and Industry Returns. The Review of Financial Studies, 30(12), 4216–4266. https://doi.org/10.1093/rfs/hhx033

Ditta, S., Chakraborty, A. (2018). Industry Concentration and Stock Returns: Indian Evidence [Review of Industry Concentration and Stock Returns: Indian Evidence]. JIM QUEST Journal of Management and Technology, 14(1), 93–101. https://jaipuria.edu.in/jim/wp-content/uploads/2018/09/JIMQUES _Issue_2018.pdf

Gao, Mingze. (2020, May 24). Merge Compustat and CRSP. Mingze-Gao.com https://mingze-gao.com /posts/merge-compustat-and-crsp/gvkey-permno-link-table

Grullon, G., Larkin, Y., Michaely, R. (2019). Are US Industries Becoming More Concentrated?*. Review of Finance, 23(4), 697–743. https://doi.org/10.1093/rof/rfz007

Philippon, T. (2019).The Economics and Politics of Market Concentration. (n.d.). NBER. https://www. nber.org/reporter/2019number4/economics-and-politics-market-concentration

Rhoades, S. A. (1993, March). The Herfindahl-Hirschman Index [Review of The Herfindahl-Hirschman Index]. Federal Reserve Bank of St. Louis; FRASER. https://fraser.stlouisfed.org/files/docs/publicatio /pages/1990-1994/33101_1990-1994.pdf

Sharma, V. (2010). Stock returns and product market competition: beyond industry concentration. Review of Quantitative Finance and Accounting, 37(3), 283–299. https://doi.org/10.1007/s11156-010-0205-0

Appendix 1:

Baseline SAS Code with Fama-French Three-Factor Model Regression

```
/*******************************************/
/*Financial Economics Project - Spring 2022*/
/*Authors: Christopher Babcock, Eric Mosher,
Santiago Olalquiaga C., and Arne Severijns*/
/*******************************************/
/*The following code is our baseline code where we clean and merge Compustat
Fundamental (COMPFUNDA) and Center for Research in Security Prices (CRSP) databases
from 2001 to 2020. COMPFUNDA provides metrics we use like companies' sales data and
industry category - North American Industry Classification System (NAICS) Code and
CRSP provides stock returns for those companies. We sort companies into 96 different
industry categories by 3-digit NAICS codes. We then calculate a Herfindahl-Hirschman
Index (HHI) through annual company sales in each industry category. A high HHI
corresponds to an industry that is highly concentrated or dominated by one or few
companies. A low HHI corresponds to an industry that is competitive between many
different companies. All companies in each category each year will have the same HHI.
We then rank all 96 categories (and the companies in those categories) by their HHI.
We then split the HHI into quintiles and invest in companies with the top 20% of HHIs
and short companies in the bottom 20% of HHIs. We then test if this portfolio
produces a significant intercept alpha from OLS regression, indicating this strategy
produces abnormal returns for the investor.*/
/*******************************************/


/*******************************************/
/*Step 1: Merge the CRSP and COMPUSTAT tables*/
/*******************************************/
```

```
/*1.1: Create a table containing the information for linking CRSP and Compustat data*/

%let beg_yr = 2000;
%let end_yr = 2020;

proc sql;
    create table lnk as
    select *
    from crsp.ccmxpf_lnkhist
    where
        /* See below for a description of the link types */
        linktype in ("LU", "LC") and
        /* Extend the period to deal with fiscal year issues */
        /* Note that the ".B" and ".E" missing value codes represent the   */
        /* earliest possible beginning date and latest possible end date   */
        /* of the Link Date range, respectively.                           */
        (&end_yr+1 >= year(linkdt) or linkdt = .B) and
        (&beg_yr-1 <= year(linkenddt) or linkenddt = .E)
        /* primary link assigned by Compustat or CRSP */
        and linkprim in ("P", "C")
    order by
        gvkey, linkdt;
quit;

/*1.2: Create a new table using COMPUSTAT data and add to it the linking information
from the table above*/

/*The first commands create a table "my data" drawing the variables from lnk and drawing
the specified variables from com.funda.
```

```
The commands below equate the two gvkey variables from the two tables, conditioning
on the dates of the observations.*/

proc sql;
    create table mydata as
    select *
    from lnk, comp.funda (keep=gvkey fyear datadate cusip conm sale naicsh) as cst
    where lnk.gvkey = cst.gvkey
    and (&beg_yr <= fyear <= &end_yr)
    and (linkdt <= cst.datadate or linkdt = .B)
    and (cst.datadate <= linkenddt or linkenddt = .E)
    and naicsh>99
    and sale is not missing;
quit;

/*1.3: Add a variable for 3-digit NAICS to "mydata"*/
data mydata;
    SET mydata;
    naic3Digit = substrn(naicsh,1,3);
run;

/*1.4: Create a new table from the data on CRSP, adding a variable that extracts
the year from the date variable*/
data CRSP_new;
    set CRSP.MSF;
    fyear = year(DATE);
run;

/*1.5: The code below creates a table merging the CRSP data from the table above
to the "my data" table that contains COMPUSTAT
```

```
The merging takes places on the PERMNO, PERMCO and YEAR variables.*/
proc sql;
    create table Merged_table as
    select a.* , b.*
    from mydata a
    inner join CRSP_new b
    on a.LPERMNO = b.PERMNO and a.LPERMCO= b.PERMCO and a.fyear = b.fyear;
quit;

/*1.6: Add a variable for each company's market capitalization*/
data Merged_table;
    SET Merged_table;
    marketCap = abs(prc*shrout);
    marketCap = COALESCE(marketCap,0);
run;

/*1.7: Remove duplicate rows*/
proc sort data=Merged_table;
    by lpermco date;
run;

proc sort data=Merged_table out=Merged_table_2 nodupkey;
    by conm date;
run;

proc sort data=Merged_table_2;
    by lpermco date;
run;
```

```
/*******************/
/*STEP 2: CREATE HHI*/
/*******************/

/*2.1: Create Ind_sales variable. Command cluster creates table "Merged_IND" pulling
all data from "Merged_table" (the asterisk in the "select" command line selects all
variables) and creates a new variable "Ind_sale" that sums sales grouped by
naic3Digit and fyear*/

proc sql;
    create table Merged_IND as
    /*The following three lines of code:
    (1) specify the source of the data for the table ("from");
    (2) with the asterisk, select all the existing variables the source table; and
    (3) create a new variable with the sum function, and label it ("as")*/
    select *
          ,sum(sale) as Ind_Sales
    from Merged_table_2
    /*The following two lines of code condition the sum operation to define the new
    variable from above, by grouping based on two
    condition variables (naic3Digit and fyear)*/
    group by naic3Digit, fyear
    order by naic3Digit, fyear;
quit;

/*2.2: Create Market share variable. Using the same command structure as in 2.1, we
create a new table containing a new variable that equals the market share of each
company for each year. The market share is calculated as the yearly sales of each
company divided by the yearly sales of the corresponding industry, using the variable
calculated in step 2.1. Note that markets are defined by the 3-Digit
```

```
level of NAICS codes.*/
proc sql;
    create table Merged_IND_2 as
    select *, divide(sale,Ind_Sales) as Market_Share
    from Merged_IND;
quit;

/*2.3: Square the market share variable*/
proc sql;
    create table Merged_IND_3 as
    select *, Market_Share**2 as Market_Share_sqrd
    from Merged_IND_2;
quit;

/*2.4: Calculate Herfindahl Hirschman index (HHI) variable. Using the same command
structure from 2.1, we create a new table that includes the HHI variable, which
is calculated as summation of yearly market share for all firms in each market,
where markets are defined by the 3-Digit
level of NAICS codes.*/
proc sql;
    create table Merged_IND_4 as
    select *, sum(Market_Share_sqrd) as HHI
    from Merged_IND_3
    group by naic3Digit, fyear
    order by naic3Digit, fyear;
quit;


/************************************************/
/*STEP 3: RANK DATA BY HHI VALUES and introduce lag*/
```

```
/************************************************/

/*3.1: Create ranking for HHI */

/*We need to create a ranking for all the HHI values within each year (i.e., each
year will be a so-called 'BY' group, within which sorting will occur). To
do so, we first sort the data by fyear, which is the variable on which the
'BY' groups are sorted.*/

proc sort data=Merged_IND_4;
    by fyear;
run;

/*Rank industries within each year based on their HHI values. The specifications in
the code give the ranking as a fraction, and create a separate ranking for each
'BY' group, defined on the fyear variable.*/

proc rank data=Merged_IND_4 out=Merged_Ranked ties=high fraction;
    by fyear;
    var HHI;
    ranks rank;
run;

/*Sort resulting data set by HHI rank */

proc sort data=Merged_Ranked out=Merged_Ranked;
    by fyear rank;
run;
```

```
/*3.2: Repeat removal of duplicate observations from the data.*/
proc sort data=Merged_Ranked;
    by lpermco date;
run;

proc sort data=Merged_Ranked out=Merged_Ranked_2 nodupkey;
    by conm date;
run;

proc sort data=Merged_Ranked_2;
    by fyear rank conm;
run;


/*3.3: Lag the HHI and rank variables by 12 months for each observation. The HHI
information is contemporaneous to stock prices, so in order to construct portfolios
that are truly realistic, we need to do the sorting based on HHI value rankings
from the preceding year.To do so, we lag the HHI and rank variables by 12 months.*/

proc sort data=Merged_Ranked_2;
    by permno date fyear;
run;

proc rank data=Merged_Ranked_2 out=Merged_Ranked_3 ties=low;
    by permno;
    var date;
    ranks time_rank2;
run;

proc sort data=Merged_Ranked_3 out=Merged_Ranked_4 nodupkey;
```

```
    by permno time_rank2;
run;

/*The statements below define the data set as a data panel, with permno and time_rank2
as the entity ID and time-stamp variables, and lag the HHI and rank variables for
one year (12 months) based on the time variable defined.*/

proc panel data=Merged_Ranked_4;
    id permno time_rank2;
    lag HHI(12) / out=HHI_lag;
run;

proc panel data=HHI_lag;
    id permno time_rank2;
    lag rank(12) / out=rank_lag;
run;

proc panel data=rank_lag;
    id permno time_rank2;
    lag marketCap(12) / out=marketCap_lag;
run;

/*Sorting again*/
proc sort data=marketCap_lag;
    by fyear rank conm;
run;


/*3.4: Identifiy the top and bottom 20% of firms based on their ranking of market
concnetration. The selection is executed on the preceding year's rank
```

```
(i.e., 'rank_12') because the portfolio construction needs to be done on
past information.*/

data Data_Indexed_1;
    set marketCap_lag;
    INDEX=.;
    if rank_12 <= .2 then INDEX=1; /*TOP: least concentrated industries*/
    if rank_12 >= .8 then INDEX=2; /*BOTTOM: most concentrated industries*/
run;

proc sort data=Data_Indexed_1 out=Data_Indexed_2;
    by fyear rank conm;
run;



/*************************************************************/
/*STEP 4: Calculate and plot returns of long/short portfolios*/
/*************************************************************/

/*4.1: Calculate value-weighted monthly returns for the portfolios constructed on the
HHI rank. A brief explanation of the code folllows.*/

/*The following command defines the macro-variable weight as the weight based on
marketCap_12.*/

%let weight = weight marketCap_12;

/*This cluster of commands:
(i) calculates the mean monthly return (VAR statement) of portfolios of assets,
(ii) where each portfolio is constructed for every month pooling assets according to
```

```
whether those assets belong to the top 20% or bottom 20% of industry
concentration (WHERE statement).
(iii) These means are calculated separately for each fiscal year (BY statement) and
(iv) the means are weighted according to the preceding year's value of the firm's
market capitalisation (WEIGHT statement).
(v) All of these weighted mean returns are then put into a table (OUTPUT statement)*/

proc means data =Data_Indexed_2;
    by fyear;
    weight marketCap_12;
    where INDEX in (1,2);
    var ret;
    class date INDEX;&weight.;
    output out=Weighted_Returns_1 (keep= date INDEX ret_mean ) mean = / autoname ;
run;

/*The following command sorts the table from above.*/

proc sort data=Weighted_Returns_1;
    by date INDEX;
run;

/*Create a table deleting the observations that
(a) are undated;
(b) are from the year 2000; and
(c) have blank values in the INDEX variable (i.e., the returns of the non-top
or bottom portfolios).*/

proc sql;
    create table Weighted_Returns_2 as
```

```
    select *
    from Weighted_Returns_1
    where DATE>=15006
    AND (INDEX = 1 or INDEX=2);
quit;

/*4.2: Transpose the portfolio returns data for better manipulation and graphing.*/
proc transpose data =Weighted_Returns_2 out =Weighted_Returns_3
(rename = ( _1=return_top _2=return_bottom) drop = _label_ );
    by date;
    id INDEX;
    var ret_mean;
run;

/*4.3: Calculate the difference in mean return for the top and bottom portfolios.*/
data Trans_Weight_Ret_Diff;
    set Weighted_Returns_3;
    where _name_='RET_Mean' ;
    ret_diff = return_bottom - return_top;
    drop _name_ ;
run;


/*4.4: Calculate cumulative returns for portfolios*/

/*The statement below calculates the cumulative return of the two portfolios, by:
(i) constructing a on-observation (i.e. one-month) lag of the portfolio's return;
(ii) create cumulative return variables for both portfolios and, using a conditional
loop, assign a value of 1 to the "starting cumulative return" (if...then statement);
(iii) formulate the cumulative return for the two portfolios for every subsequent
```

```
period (else do statement).*/

data Cumulative_Returns;
    set Trans_Weight_Ret_Diff;
    lag_ret_top=lag(return_top);
    lag_ret_bot=lag(return_bottom);
    retain cum1 cum2;
    if _N_= 1 then do;
    cum_top=1;
    cum_bottom=1;
    cum1 = cum_top;
    cum2 = cum_bottom;
    ret_cum=cum1-cum2;
    end ;
    else do ;
    cum_top = (1 + lag_ret_top) * cum1;
    cum_bottom = (1 + lag_ret_bot) * cum2;
    cum1 = cum_top;
    cum2 = cum_bottom;
    ret_cum=cum1-cum2;
    end;
run;

/*4.5: Plot cumulative returns*/

/*Suppress date and set landscape orientation of PDF output*/

options nodate orientation=landscape;

/*Plot cumulative returns of portfolios*/
```

```
title "Cumulative Returns of Top- and Bottom-sorted portfolios
sorted on industry concentration";
proc sgplot
    data=Cumulative_Returns;
    series x=date y=cum_top /
        lineattrs=(color=CXEA5728 thickness=1pt)
        legendlabel="Least concentrated industries portfolio";
    series x=date y=cum_bottom /
        lineattrs=(color=CX004FDB thickness=1pt)
        legendlabel="Most concentrated industries portfolio";
    yaxis label="Value of $1 deposited in January 2001 after t months";
run;



/*****************************************/
/*STEP 5: Run OLS regression to find alpha*/
/*****************************************/

/*5.1 Simple T-Test*/
proc ttest data = Trans_Weight_Ret_Diff;
    Title ' T-Test ';
    var ret_diff;
run;

/*5.2 Reproduce table with difference in mean portfolio returns and format to
merge with Fama-French factors data.*/

proc sort data=Trans_Weight_Ret_Diff out=Returns_for_regression;
    by date;
```

```
run;

data Returns_for_regression;
    set Returns_for_regression;
    format date YYMMN.;
    informat date best32.;
run;

proc rank data=Returns_for_regression out=Returns_for_regression ties=dense;
    var date;
    ranks time_rank;
run;

/*5.3: Import and format Fama-French factors data*/

/*Import Fama-French file*/

%web_drop_table(FF);

FILENAME FF '/home/columbia/sjo2125/FF_final.csv';

PROC IMPORT DATAFILE=FF
    DBMS=CSV
    OUT=FF;
    GETNAMES=YES;
RUN;

PROC CONTENTS DATA=FF;
RUN;
```

```
%web_open_table(FF);

/*Clean and format factors data; define date rank to merge with returns table.*/

Data FF;
set FF;
    date_char = Put(date, 6.);
    date = input(date_char, YYMMN6.);
    format date YYMMN.;
Run;

proc rank data=FF out=FF ties=low;
    var date;
    ranks time_rank;
run;



/*5.4: Merge Portfolio returns table with Fama-French factors table.*/
proc sql;
    create table final_regression as
    select L.*, R.*
    from Returns_for_regression as L
    left join  FF as R
    on L.time_rank = R.time_rank;
quit;

/*5.5: Run basic OLS regression.*/
proc reg;
    Title "OLS Regression";
    model ret_diff=mktrf smb hml;
```

```
quit;

/*5.5: Run OLS regression with White error correction.*/
proc model data =final_regression;
    Title "Results with White Corrections" ;
    ret_diff = a1 + b1*mktrf + b2*smb + b3*hml;
    fit ret_diff / ols hccme= 1 ;
quit;
```

Appendix 2:

Baseline SAS Code with Carhart Four-Factor Model Regression

```
/*********************************************/
/*Financial Economics Project - Spring 2022*/
/*Authors: Christopher Babcock, Eric Mosher,
Santiago Olalquiaga C., and Arne Severijns*/
/*********************************************/
/*The following code will replace Step 5 in the previous code, Appendix 1. Instead of
using the Fama-French Three-Factor model for regression, this code adds the momentum
factor to form the Carhart Four-Factor model. This in turn, leads to increased
significance in the factor variables, (besides smb) and worsens significance of our
alpha intercept. This is discussed above in our paper.*/
/*********************************************/


/*****************************************/
/*STEP 5: Run OLS regression to find alpha*/
/*****************************************/

/*STEP 5: Test Difference between Top and Bottom Portfolio Returns*/

/*5.1 Simple T-Test*/
proc ttest data = Trans_Weight_Ret_Diff;
    Title ' T-Test ';
    var ret_diff;
run;

/*5.2 OLS Regressions*/
```

```
/*Testing abnormal return using 4-factor model with Fama-French and Momentum factors*/
proc sql;
    create table ff4
    as select a.*, b.mktrf, b.smb, b.hml, b.umd, b.rf
    from Trans_Weight_Ret_Diff as a, ff.factors_monthly as b
    where put(a.date, yymmn6.)=put(b.date, yymmn6.)
    order by a.date;
quit;

proc reg;
    Title "OLS Regression";
    model ret_diff=mktrf smb hml umd;
quit;
```

Appendix 3:

Expanded Dates of Study from 1986 - 2020

```
/*********************************************/
/*Financial Economics Project - Spring 2022*/
/*Authors: Christopher Babcock, Eric Mosher,
Santiago Olalquiaga C., and Arne Severijns*/
/*********************************************/
/*The following code builds on the previous two appendices in that it uses a Carhart
four-factor model and extends the beginning date range from 2000 to 1986, the earliest
date in the base COMPFUNDA database. This enriches our results, confirming Hou and
Robinson's findings while showing that the investment strategy reversed from when their
paper was published from more competitive companies providing higher returns to more
concentrated companies providing higher returns after approximately 2002. Since the
financial crisis in 2008/2009, this gap has continued to widen, where more concentrated
industries are consistently producing higher returns
than competitive industries.*/
/*********************************************/

/*********************************************/
/*Step 1: Merge the CRSP and COMPUSTAT tables*/
/*********************************************/

/*1.1: Create a table containing the information for linking CRSP and Compustat data*/

%let beg_yr = 1986;
%let end_yr = 2020;

proc sql;
    create table lnk as
```

```
    select *
    from crsp.ccmxpf_lnkhist
    where
        /* See below for a description of the link types */
        linktype in ("LU", "LC") and
        /* Extend the period to deal with fiscal year issues */
        /* Note that the ".B" and ".E" missing value codes represent the   */
        /* earliest possible beginning date and latest possible end date   */
        /* of the Link Date range, respectively.                           */
        (&end_yr+1 >= year(linkdt) or linkdt = .B) and
        (&beg_yr-1 <= year(linkenddt) or linkenddt = .E)
        /* primary link assigned by Compustat or CRSP */
        and linkprim in ("P", "C")
    order by
        gvkey, linkdt;
quit;

/*1.2: Create a new table using COMPUSTAT data and add to it the linking
information from the table above*/

/*The first commands create a table "my data" drawing the variables from lnk and
drawing the specified variables from compfunda. The commands below equate the two
gvkey variables from the two tables, conditioning on the dates of the observations.*/

proc sql;
    create table mydata as
    select *
    from lnk, comp.funda (keep=gvkey fyear datadate cusip conm sale naicsh) as cst
    where lnk.gvkey = cst.gvkey
    and (&beg_yr <= fyear <= &end_yr)
```

```
    and (linkdt <= cst.datadate or linkdt = .B)
    and (cst.datadate <= linkenddt or linkenddt = .E)
    and naicsh>99
    and sale is not missing;
quit;

/*1.3: Add a variable for 3-digit NAICS to "mydata"*/
data mydata;
    SET mydata;
    naic3Digit = substrn(naicsh,1,3);
run;

/*1.4: Create a new table from the data on CRSP, adding a variable that extracts
the year from the date variable*/

data CRSP_new;
    set CRSP.MSF;
    fyear = year(DATE);
run;

/*1.5: The code below creates a table merging the CRSP data from the table above to the
"my data" table that contains COMPUSTAT. The merging takes places on the PERMNO,
PERMCO and YEAR variables.*/

proc sql;
    create table Merged_table as
    select a.* , b.*
    from mydata a
    inner join CRSP_new b
    on a.LPERMNO = b.PERMNO and a.LPERMCO= b.PERMCO and a.fyear = b.fyear ;
```

```
quit;

/*1.6: Add a variable for each company's market capitalization*/
data Merged_table;
    SET Merged_table;
    marketCap = abs(prc*shrout);
    marketCap = COALESCE(marketCap,0);
run;

/*1.7: Remove duplicate rows*/
proc sort data=Merged_table;
    by lpermco date;
run;

proc sort data=Merged_table out=Merged_table_2 nodupkey;
    by conm date;
run;

proc sort data=Merged_table_2;
    by lpermco date;
run;



/********************/
/*STEP 2: CREATE HHI*/
/********************/

/*2.1: Create Ind_sales variable. Command cluster creates table "Merged_IND" pulling
all data from "Merged_table" (the asterisk in the "select" command line selects all
variables) and creates a new variable "Ind_sale" that sums sales grouped by
```

```
naic3Digit and fyear*/

proc sql;
    create table Merged_IND as
    /*The following three lines of code (1) specify the source of the data for the
    table ("from"); (2) with the asterisk,
    select all the existing variables the source table; and (3) create a new variable
    with the sum function, and label it ("as")*/
    select *, sum(sale) as Ind_Sales
    from Merged_table_2
    /*The following two lines of code condition the sum operation to define the new
    variable from above, by grouping based on two condition variables (naic3Digit
    and fyear)*/
    group by naic3Digit, fyear
    order by naic3Digit, fyear;
quit;

/*2.2: Create Market share variable. Using the same command structure as in 2.1,
we create a new table containing a new variable that equals the market share of
each company for each year. The market share is calculated as the yearly sales of
each company divided by the yearly sales of the corresponding industry, using the
variable calculated in step 2.1. Note that markets are defined by the 3-Digit level
of NAICS codes.*/

proc sql;
    create table Merged_IND_2 as
    select *, divide(sale,Ind_Sales) as Market_Share
    from Merged_IND;
quit;
```

```
/*2.3: Square the market share variable*/
proc sql;
    create table Merged_IND_3 as
    select *, Market_Share**2 as Market_Share_sqrd
    from Merged_IND_2;
quit;

/*2.4: Calculate Herfindahl{Hirschman index (HHI) variable. Using the same command
structure from 2.1, we create a new table that includes the HHI variable, which is
calculated as summation of yearly market share for all firms in each market, where
markets are defined by the 3-Digit level of NAICS codes.*/

proc sql;
    create table Merged_IND_4 as
    select *, sum(Market_Share_sqrd) as HHI
    from Merged_IND_3
    group by naic3Digit, fyear
    order by naic3Digit, fyear;
quit;



/************************************************/
/*STEP 3: RANK DATA BY HHI VALUES and introduce lag*/
/************************************************/

/*3.1: Create ranking for HHI */

/*We need to create a ranking for all the HHI values within each year (i.e., each
year will be a so-called 'BY' group, within which sorting will occur). To do so,
we first sort the data by fyear, which is the variable on which the 'BY' groups
```

```
are sorted.*/

proc sort data=Merged_IND_4;
    by fyear;
run;

/*Rank industries within each year based on their HHI values. The specifications in
the code give the ranking as a fraction, and create a separate ranking for each 'BY'
group, defined on the fyear variable.*/

proc rank data=Merged_IND_4 out=Merged_Ranked ties=high fraction;
    by fyear;
    var HHI;
    ranks rank;
run;

/*Sort resulting data set by HHI rank */

proc sort data=Merged_Ranked out=Merged_Ranked;
    by fyear rank;
run;


/*3.2: Repeat removal of duplicate observations from the data.*/

proc sort data=Merged_Ranked;
    by lpermco date;
run;

proc sort data=Merged_Ranked out=Merged_Ranked_2 nodupkey;
```

```
    by conm date;
run;

proc sort data=Merged_Ranked_2;
    by fyear rank conm;
run;


/*3.3: Lag the HHI and rank variables by 12 months for each observation. The HHI
information is contemporaneous to stock prices, so in order to construct portfolios
that are truly realistic, we need to do the sorting based on HHI value rankings from
the preceding year. To do so, we lag the HHI and rank variables by 12 months.*/

proc sort data=Merged_Ranked_2;
    by permno date fyear;
run;

proc rank data=Merged_Ranked_2 out=Merged_Ranked_3 ties=low;
    by permno;
    var date;
    ranks time_rank2;
run;


proc sort data=Merged_Ranked_3 out=Merged_Ranked_4 nodupkey;
    by permno time_rank2;
run;

/*The statements below define the data set as a data panel, with permno and
time_rank2 as the entity ID and time-stamp variables, and lag the HHI and rank
```

```
variables for one year (12 months) based on the time variable defined.*/

proc panel data=Merged_Ranked_4;
    id permno time_rank2;
    lag HHI(12) / out=HHI_lag;
run;

proc panel data=HHI_lag;
    id permno time_rank2;
    lag rank(12) / out=rank_lag;
run;

proc panel data=rank_lag;
    id permno time_rank2;
    lag marketCap(12) / out=marketCap_lag;
run;

/*Sorting again*/
proc sort data=marketCap_lag;
    by fyear rank conm;
run;


/*3.4: Identifiy the top and bottom 20% of firms based on their ranking of market
concentration. The selection is executed on the preceding year's rank (i.e., 'rank_12')
because the portfolio construction needs to be done on past information.*/

data Data_Indexed_1;
    set marketCap_lag;
    INDEX=.;
```

```
    if rank_12 <= .2 then INDEX=1; /*TOP: least concentrated industries*/
    if rank_12 >= .8 then INDEX=2; /*BOTTOM: most concentrated industries*/
run;

proc sort data=Data_Indexed_1 out=Data_Indexed_2;
    by fyear rank conm;
run;


/*************************************************************/
/*STEP 4: Calculate and plot returns of long/short portfolios*/
/*************************************************************/

/*4.1: Calculate value-weighted monthly returns for the portfolios constructed on
the HHI rank. A brief explanation of the code folllows.*/

/*The following command defines the macro-variable weight as the weight based on
marketCap_12.*/

%let weight = weight marketCap_12;

/*This cluster of commands:
(i) calculates the mean monthly return (VAR statement) of portfolios of assets,
(ii)where each portfolio is constructed for every month pooling assets according to
whether those assets belong to the top 20% or bottom 20% of industry concentration
(WHERE statement).
(iii) These means are calculated separately for each fiscal year (BY statement) and
(iv) the means are weighted according to the preceding year's value
of the firm's market capitalisation (WEIGHT statement).
(v) All of these weighted mean returns are then put into a table (OUTPUT statement)*/
```

```
proc means data =Data_Indexed_2;
    by fyear;
    weight marketCap_12;
    where INDEX in (1,2);
    var ret;
    class date INDEX;&weight.;
    output out=Weighted_Returns_1 (keep= date INDEX ret_mean ) mean = / autoname ;
run;

/*The following command sorts the table from above.*/

proc sort data=Weighted_Returns_1;
    by date INDEX;
run;

/*Create a table deleting the observations that
(a) are undated;
(b) are from the year 1987 (Weighted Returns are lagged a year, so start at 1987 since
our beginning date is January 1986; and
(c) have blank values in the INDEX variable (i.e., the returns of the non-top or
bottom portfolios).*/

proc sql;
    create table Weighted_Returns_2 as
    select *
    from Weighted_Returns_1
    where DATE>= 9862
    AND (INDEX = 1 or INDEX=2);
```

```
quit;

/*4.2: Transpose the portfolio returns data for better manipulation and graphing.*/

proc transpose data =Weighted_Returns_2 out =Weighted_Returns_3 (rename =
( _1=return_top _2=return_bottom) drop = _label_ );
    by date;
    id INDEX;
    var ret_mean;
run;

/*4.3: Calculate the difference in mean return for the top and bottom portfolios.*/
data Trans_Weight_Ret_Diff;
    set Weighted_Returns_3;
    where _name_='RET_Mean' ;
    ret_diff = return_bottom - return_top;
    drop _name_ ;
run;


/*4.4: Calculate cumulative returns for portfolios*/

/*The statement below calculates the cumulative return of the two portfolios, by:
(i) constructing a on-observation (i.e. one-month) lag of the portfolio's return;
(ii) create cumulative return variables for both portfolios and, using a
conditional loop, assign a value of 1 to the "starting cumulative return"
(if...then statement);
(iii) formulate the cumulative return for the two portfolios for every subsequent
period (else do statement).*/
```

```
data Cumulative_Returns;
    set Trans_Weight_Ret_Diff;
    lag_ret_top=lag(return_top);
    lag_ret_bot=lag(return_bottom);
    retain cum1 cum2;
    if _N_= 1 then do;
    cum_top=1;
    cum_bottom=1;
    cum1 = cum_top;
    cum2 = cum_bottom;
    ret_cum=cum1-cum2;
    end ;
    else do ;
    cum_top = (1 + lag_ret_top) * cum1;
    cum_bottom = (1 + lag_ret_bot) * cum2;
    cum1 = cum_top;
    cum2 = cum_bottom;
    ret_cum=cum1-cum2;
    end;
run;

/*4.5: Plot cumulative returns*/

/*Suppress date and set landscape orientation of PDF output*/

options nodate orientation=landscape;

/*Plot cumulative returns of portfolios*/

title "Cumulative Returns of Top- and Bottom-sorted portfolios
```

```
sorted on industry concentration";
proc sgplot
    data=Cumulative_Returns;
    series x=date y=cum_top /
        lineattrs=(color=CXEA5728 thickness=1pt)
        legendlabel="Least concentrated industries portfolio";
    series x=date y=cum_bottom /
        lineattrs=(color=CX004FDB thickness=1pt)
        legendlabel="Most concentrated industries portfolio";
    yaxis label="Value of $1 deposited in January 1987 after t months";
run;



/*****************************************/
/*STEP 5: Run OLS regression to find alpha*/
/*****************************************/

/*STEP 5: Test Difference between Top and Bottom Portfolio Returns*/

/*5.1 Simple T-Test*/
proc ttest data = Trans_Weight_Ret_Diff;
    Title ' T-Test ';
    var ret_diff;
run;

/*5.2 OLS Regressions*/
/*Testing abnormal return using 4-factor model with Fama-French and Momentum factors*/
proc sql;
    create table ff4
    as select a.*, b.mktrf, b.smb, b.hml, b.umd, b.rf
```

```
    from Trans_Weight_Ret_Diff as a, ff.factors_monthly as b
    where put(a.date, yymmn6.)=put(b.date, yymmn6.)
    order by a.date;
quit;

proc reg;
    Title "OLS Regression";
    model ret_diff=mktrf smb hml umd;
quit;

/*5.3 White Correction*/
proc model data = ff4;
    Title ' Results with White Corrections';
    ret_diff = a1 + b1*mktrf + b2*smb + b3*hml + b4*umd;
    fit ret_diff / ols hccme= 1 ;
quit;
```

Appendix 4:

Equal Weighting Decile Strategy: Top and Bottom 10%

```
/*********************************************/
/*Financial Economics Project - Spring 2022*/
/*Authors: Christopher Babcock, Eric Mosher,
Santiago Olalquiaga C., and Arne Severijns*/
/*********************************************/
/*The following code builds on the previous three appendices in that it uses a Carhart
four-factor model and data range from 1986 to 2020. The weighting on this portfolio
change from top/bottom 20% in the previous models to top/bottom 10% in this model.*/
/*********************************************/

/*********************************************/
/*Step 1: Merge the CRSP and COMPUSTAT tables*/
/*********************************************/

/*1.1: Create a table containing the information for linking CRSP and Compustat data*/

%let beg_yr = 1986;
%let end_yr = 2020;

proc sql;
    create table lnk as
    select *
    from crsp.ccmxpf_lnkhist
    where
        /* See below for a description of the link types */
        linktype in ("LU", "LC") and
        /* Extend the period to deal with fiscal year issues */
```

```
        /* Note that the ".B" and ".E" missing value codes represent the   */
        /* earliest possible beginning date and latest possible end date   */
        /* of the Link Date range, respectively.                           */
        (&end_yr+1 >= year(linkdt) or linkdt = .B) and
        (&beg_yr-1 <= year(linkenddt) or linkenddt = .E)
        /* primary link assigned by Compustat or CRSP */
        and linkprim in ("P", "C")
    order by
        gvkey, linkdt;
quit;

/*1.2: Create a new table using COMPUSTAT data and add to it the linking
information from the table above*/

/*The first commands create a table "my data" drawing the variables from lnk and
drawing the specified variables from compfunda. The commands below equate the two
gvkey variables from the two tables, conditioning on the dates of the observations.*/

proc sql;
    create table mydata as
    select *
    from lnk, comp.funda (keep=gvkey fyear datadate cusip conm sale naicsh) as cst
    where lnk.gvkey = cst.gvkey
    and (&beg_yr <= fyear <= &end_yr)
    and (linkdt <= cst.datadate or linkdt = .B)
    and (cst.datadate <= linkenddt or linkenddt = .E)
    and naicsh>99
    and sale is not missing;
quit;
```

```
/*1.3: Add a variable for 3-digit NAICS to "mydata"*/
data mydata;
    SET mydata;
    naic3Digit = substrn(naicsh,1,3);
run;

/*1.4: Create a new table from the data on CRSP, adding a variable that extracts
the year from the date variable*/

data CRSP_new;
    set CRSP.MSF;
    fyear = year(DATE);
run;

/*1.5: The code below creates a table merging the CRSP data from the table above to the
"my data" table that contains COMPUSTAT. The merging takes places on the PERMNO,
PERMCO and YEAR variables.*/

proc sql;
    create table Merged_table as
    select a.* , b.*
    from mydata a
    inner join CRSP_new b
    on a.LPERMNO = b.PERMNO and a.LPERMCO= b.PERMCO and a.fyear = b.fyear ;
quit;

/*1.6: Add a variable for each company's market capitalization*/
data Merged_table;
    SET Merged_table;
    marketCap = abs(prc*shrout);
```

```
    marketCap = COALESCE(marketCap,0);
run;

/*1.7: Remove duplicate rows*/
proc sort data=Merged_table;
    by lpermco date;
run;

proc sort data=Merged_table out=Merged_table_2 nodupkey;
    by conm date;
run;

proc sort data=Merged_table_2;
    by lpermco date;
run;



/********************/
/*STEP 2: CREATE HHI*/
/********************/

/*2.1: Create Ind_sales variable. Command cluster creates table "Merged_IND" pulling
all data from "Merged_table" (the asterisk in the "select" command line selects all
variables) and creates a new variable "Ind_sale" that sums sales grouped by
naic3Digit and fyear*/

proc sql;
    create table Merged_IND as
    /*The following three lines of code (1) specify the source of the data for the
    table ("from"); (2) with the asterisk,
```

```
    select all the existing variables the source table; and (3) create a new variable
    with the sum function, and label it ("as")*/
    select *, sum(sale) as Ind_Sales
    from Merged_table_2
    /*The following two lines of code condition the sum operation to define the new
    variable from above, by grouping based on two condition variables (naic3Digit
    and fyear)*/
    group by naic3Digit, fyear
    order by naic3Digit, fyear;
quit;

/*2.2: Create Market share variable. Using the same command structure as in 2.1,
we create a new table containing a new variable that equals the market share of
each company for each year. The market share is calculated as the yearly sales of
each company divided by the yearly sales of the corresponding industry, using the
variable calculated in step 2.1. Note that markets are defined by the 3-Digit level
of NAICS codes.*/

proc sql;
    create table Merged_IND_2 as
    select *, divide(sale,Ind_Sales) as Market_Share
    from Merged_IND;
quit;

/*2.3: Square the market share variable*/
proc sql;
    create table Merged_IND_3 as
    select *, Market_Share**2 as Market_Share_sqrd
    from Merged_IND_2;
quit;
```

```
/*2.4: Calculate Herfindahl{Hirschman index (HHI) variable. Using the same command
structure from 2.1, we create a new table that includes the HHI variable, which is
calculated as summation of yearly market share for all firms in each market, where
markets are defined by the 3-Digit level of NAICS codes.*/

proc sql;
    create table Merged_IND_4 as
    select *, sum(Market_Share_sqrd) as HHI
    from Merged_IND_3
    group by naic3Digit, fyear
    order by naic3Digit, fyear;
quit;



/*************************************************/
/*STEP 3: RANK DATA BY HHI VALUES and introduce lag*/
/*************************************************/

/*3.1: Create ranking for HHI */

/*We need to create a ranking for all the HHI values within each year (i.e., each
year will be a so-called 'BY' group, within which sorting will occur). To do so,
we first sort the data by fyear, which is the variable on which the 'BY' groups
are sorted.*/

proc sort data=Merged_IND_4;
    by fyear;
run;
```

```
/*Rank industries within each year based on their HHI values. The specifications in
the code give the ranking as a fraction, and create a separate ranking for each 'BY'
group, defined on the fyear variable.*/

proc rank data=Merged_IND_4 out=Merged_Ranked ties=mean fraction;
    by fyear;
    var HHI;
    ranks rank;
run;

/*Sort resulting data set by HHI rank */

proc sort data=Merged_Ranked out=Merged_Ranked;
    by fyear rank;
run;



/*3.2: Repeat removal of duplicate observations from the data.*/

proc sort data=Merged_Ranked;
    by lpermco date;
run;

proc sort data=Merged_Ranked out=Merged_Ranked_2 nodupkey;
    by conm date;
run;

proc sort data=Merged_Ranked_2;
    by fyear rank conm;
run;
```

```
/*3.3: Lag the HHI and rank variables by 12 months for each observation. The HHI
information is contemporaneous to stock prices, so in order to construct portfolios
that are truly realistic, we need to do the sorting based on HHI value rankings from
the preceding year. To do so, we lag the HHI and rank variables by 12 months.*/

proc sort data=Merged_Ranked_2;
    by permno date fyear;
run;

proc rank data=Merged_Ranked_2 out=Merged_Ranked_3 ties=mean;
    by permno;
    var date;
    ranks time_rank2;
run;


proc sort data=Merged_Ranked_3 out=Merged_Ranked_4 nodupkey;
    by permno time_rank2;
run;

/*The statements below define the data set as a data panel, with permno and
time_rank2 as the entity ID and time-stamp variables, and lag the HHI and rank
variables for one year (12 months) based on the time variable defined.*/

proc panel data=Merged_Ranked_4;
    id permno time_rank2;
    lag HHI(12) / out=HHI_lag;
run;
```

```
proc panel data=HHI_lag;
    id permno time_rank2;
    lag rank(12) / out=rank_lag;
run;

proc panel data=rank_lag;
    id permno time_rank2;
    lag marketCap(12) / out=marketCap_lag;
run;

/*Sorting again*/
proc sort data=marketCap_lag;
    by fyear rank conm;
run;


/*3.4: Identifiy the top and bottom 10% of firms based on their ranking of market
concentration. The selection is executed on the preceding year's rank (i.e., 'rank_12')
because the portfolio construction needs to be done on past information.*/

data Data_Indexed_1;
    set marketCap_lag;
    INDEX=.;
    if rank_12 <= .1 then INDEX=1; /*TOP: least concentrated industries*/
    if rank_12 >= .9 then INDEX=2; /*BOTTOM: most concentrated industries*/
run;

proc sort data=Data_Indexed_1 out=Data_Indexed_2;
    by fyear rank conm;
```

```
run;


/*************************************************************/
/*STEP 4: Calculate and plot returns of long/short portfolios*/
/*************************************************************/

/*4.1: Calculate value-weighted monthly returns for the portfolios constructed on
the HHI rank. A brief explanation of the code folllows.*/

/*The following command defines the macro-variable weight as the weight based on
marketCap_12.*/

%let weight = weight marketCap_12;

/*This cluster of commands:
(i) calculates the mean monthly return (VAR statement) of portfolios of assets,
(ii)where each portfolio is constructed for every month pooling assets according to
whether those assets belong to the top 10% or bottom 10% of industry concentration
(WHERE statement).
(iii) These means are calculated separately for each fiscal year (BY statement) and
(iv) the means are weighted according to the preceding year's value
of the firm's market capitalisation (WEIGHT statement).
(v) All of these weighted mean returns are then put into a table (OUTPUT statement)*/


proc means data =Data_Indexed_2;
    by fyear;
    weight marketCap_12;
    where INDEX in (1,2);
```

```
    var ret;
    class date INDEX;&weight.;
    output out=Weighted_Returns_1 (keep= date INDEX ret_mean ) mean = / autoname ;
run;

/*The following command sorts the table from above.*/

proc sort data=Weighted_Returns_1;
    by date INDEX;
run;

/*Create a table deleting the observations that
(a) are undated;
(b) are from the year 1987 (Weighted Returns are lagged a year, so start at 1987 since
our beginning date is January 1986; and
(c) have blank values in the INDEX variable (i.e., the returns of the non-top or
bottom portfolios).*/

proc sql;
    create table Weighted_Returns_2 as
    select *
    from Weighted_Returns_1
    where DATE>= 9862
    AND (INDEX = 1 or INDEX=2);
quit;

/*4.2: Transpose the portfolio returns data for better manipulation and graphing.*/

proc transpose data =Weighted_Returns_2 out =Weighted_Returns_3 (rename =
( _1=return_top _2=return_bottom) drop = _label_ );
```

```
    by date;
    id INDEX;
    var ret_mean;
run;

/*4.3: Calculate the difference in mean return for the top and bottom portfolios.*/
data Trans_Weight_Ret_Diff;
    set Weighted_Returns_3;
    where _name_='RET_Mean' ;
    ret_diff = return_bottom - return_top;
    drop _name_ ;
run;


/*4.4: Calculate cumulative returns for portfolios*/

/*The statement below calculates the cumulative return of the two portfolios, by:
(i) constructing a on-observation (i.e. one-month) lag of the portfolio's return;
(ii) create cumulative return variables for both portfolios and, using a
conditional loop, assign a value of 1 to the "starting cumulative return"
(if...then statement);
(iii) formulate the cumulative return for the two portfolios for every subsequent
period (else do statement).*/

data Cumulative_Returns;
    set Trans_Weight_Ret_Diff;
    lag_ret_top=lag(return_top);
    lag_ret_bot=lag(return_bottom);
    retain cum1 cum2;
    if _N_= 1 then do;
```

```
    cum_top=1;
    cum_bottom=1;
    cum1 = cum_top;
    cum2 = cum_bottom;
    ret_cum=cum1-cum2;
    end ;
    else do ;
    cum_top = (1 + lag_ret_top) * cum1;
    cum_bottom = (1 + lag_ret_bot) * cum2;
    cum1 = cum_top;
    cum2 = cum_bottom;
    ret_cum=cum1-cum2;
    end;
run;

/*4.5: Plot cumulative returns*/

/*Suppress date and set landscape orientation of PDF output*/

options nodate orientation=landscape;

/*Plot cumulative returns of portfolios*/

title "Cumulative Returns of Top-10% and Bottom-10% sorted portfolios
sorted on industry concentration";
proc sgplot
    data=Cumulative_Returns;
    series x=date y=cum_top /
        lineattrs=(color=CXEA5728 thickness=1pt)
        legendlabel="Least concentrated industries portfolio";
```

```
    series x=date y=cum_bottom /
        lineattrs=(color=CX004FDB thickness=1pt)
        legendlabel="Most concentrated industries portfolio";
    yaxis label="Value of $1 deposited in January 1987 after t months";
run;



/*****************************************/
/*STEP 5: Run OLS regression to find alpha*/
/*****************************************/

/*STEP 5: Test Difference between Top and Bottom Portfolio Returns*/

/*5.1 Simple T-Test*/
proc ttest data = Trans_Weight_Ret_Diff;
    Title ' T-Test ';
    var ret_diff;
run;

/*5.2 OLS Regressions*/
/*Testing abnormal return using 4-factor model with Fama-French and Momentum factors*/
proc sql;
    create table ff4
    as select a.*, b.mktrf, b.smb, b.hml, b.umd, b.rf
    from Trans_Weight_Ret_Diff as a, ff.factors_monthly as b
    where put(a.date, yymmn6.)=put(b.date, yymmn6.)
    order by a.date;
quit;

proc reg;
```

```
    Title "OLS Regression";
    model ret_diff=mktrf smb hml umd;
quit;

/*5.3 White Correction*/
proc model data = ff4;
    Title ' Results with White Corrections';
    ret_diff = a1 + b1*mktrf + b2*smb + b3*hml + b4*umd;
    fit ret_diff / ols hccme= 1 ;
quit;
```