MGMT 237H Quantitative Asset Management

Homework 4

Group 5:

Jie "Edward" Sheng (jie.sheng.2012@anderson.ucla.edu)

Anshul Maheshwari (anshul.maheshwari.2012@anderson.ucla.edu)

Chao Liang (chao.liang.2012@anderson.ucla.edu)

Li Zhang (li.zhang.2012@anderson.ucla.edu)

Problem Statement 1: Replicate Table 1 of Hirshleifer, Hou, Teoh and Zhang 2004 Paper on "Do Investors Overvalue Firms with Bloated Balance Sheets?"

Table 1 data is made up of fundamentals that are based on Compustat database. This database can be accessed using WRDS. These are company financials that are reported yearly. Because of company filing dates being different that fiscal year dates we need to make some adjustments to the data. These adjustments are listed below at appropriate places. Also the mapping of data, as described in the paper that was written in 2004, has changed. Following table summarizes the new mapping versus the earlier mapping and description.

Old	Description	New
1	Cash and short Term Investment	CHE
4	Current Asset	ACT
5	Current Liabilities	LCT
6	Total Asset	AT
9	Long Term Debt	DLTT
14	Depreciation and Amortization	DP
25	Shares Outstanding	CSHO
34	Debt included in current liabilities	DLC
38	Minor Interests	MIB
60	Book Value of Common Equity	CEQ
71	Income Tax Payable	TXP
130	Preferred Stocks	PSTK
178	Income From Continuing Operations	OIADP
199	Fiscal Year End Closing Price	PRCC_F

Following are the steps, assumptions and data related details that we encountered during our assignment:

Part 1 - COMPUSTAT Data Processing:

- Download data from fiscal year 1963 to 2000.
- Keep data only for following exchanges (COMPUSTAT code in brackets)
  NYSE (11)AMEX (12) and NASDAQ (14).
- Calculate one year lag of total assets.
- If the net operating assets fields are missing we set them to 0 (as discussed in the paper)
- We require Total assets, Book Value, Fiscal Year End Closing price, Number of Shares outstanding to be positive. Hence we filter out all records that do not meet this criterion.
- We calculate NOA, Earnings, Accruals, Cash Flows, Book Value, Market Value and Book/Market ratio.

- Drop the first row of each Stock that corresponds to the first year as we don't have lagged Total Assets for that year.
- Drop data for which NOA is missing.
- Rank the NOA in each year into deciles.
- Calculate the mean and median of each of the variables: NOA, Earnings, Accruals, Cash Flows, BV, MV and B/M
- Output the results to CSV files.

Part 2 - CRSP Data Processing:

Following are steps that we followed:

- Keep records that belong to following exchanges (CRSP Exchange Codes in Brackets): NYSE (1) AMEX (3) and NASDAQ (3)
- Merge CRSP with Compustat.
  - Prepare compustat for merge
  - Move month of report date forward by 4 months
  - The financial data will be used from this month and onward, as assumed in page 23 of the paper
- Delete the data with missing and negative returns
- Add PERMNO column to Compustat to help in merging CRSP and COMPUSTAT
- Merge CRSP and COMPUSTAT by matching PERMNO and YEAR
- We create a macro to move the data to previous year depending on the fiscal month and the report month when the company reports their data
- Delete the data for fiscal months for the first year in the data set as these returns should be used in previous year (as per logic described above)
- Rank the Stocks for each month every year into 10 deciles using NOA

Part 3 - Calculate Beta

Our methodology to calculate beta:

- We use three nested loops- 1) Decile 2) Year 3) Month
- As discussed in the paper we do a rolling regression of decile portfolio return on NYSE AMEX Equal Weighted Return
- The Rolling regression is for 60 months data
- We calculate beta of the each decile every month of all years
- The number of regressions required = ~30(years) * 12 (months) * 10 (deciles) = ~3600!
- The beta of each portfolio/decile is stored in its own table

- Take the mean and median of each decile to get the betas

Here is the replicated Table 1 based on our code and data selected:

Compustat: 151,754 records imported, 86,798 records used.

CRSP: 2,672,878 records imported, 1,220,092 records are used.

TABLE 1

Mean (Median) Values of Selected Characteristics for Decile Portfolios Sorted by NOA

Portfolio NOA Ranking

	Lowest	2	3	4	5	6	7	8	9	Highest
Panel A: Accounting Variables										
NOA	0.251*	0.490	0.589	0.654	0.705	0.751	0.798	0.854	0.943	1.485
	_0.264**_	_0.485_	_0.575_	_0.637_	_0.688_	_0.738_	_0.788_	_0.849_	_0.936_	_1.171_
Earnings	0.019	0.077	0.100	0.107	0.110	0.113	0.118	0.123	0.136	0.144
	_0.060_	_0.090_	_0.100_	_0.104_	_0.105_	_0.108_	_0.110_	_0.116_	_0.126_	_0.149_
Accruals	-0.075	-0.055	-0.044	-0.039	-0.033	-0.026	-0.019	-0.004	0.016	0.079
	_-0.050_	_-0.049_	_-0.044_	_-0.041_	_-0.037_	_-0.033_	_-0.026_	_-0.016_	_-0.001_	_0.033_
Cashflows	0.094	0.132	0.144	0.146	0.143	0.140	0.137	0.126	0.119	0.064
	_0.121_	_0.146_	_0.148_	_0.148_	_0.142_	_0.140_	_0.135_	_0.128_	_0.121_	_0.099_
BV ($m)	190	420	624	663	593	546	459	418	399	327
	_26_	_46_	_62_	_71_	_82_	_82_	_76_	_65_	_62_	_46_
Panel B: Asset Pricing Variables										
MV ($m)	650	1147	1806	1859	1458	1407	944	919	973	925
	_60_	_77_	_94_	_108_	_115_	_118_	_110_	_98_	_105_	_100_
B/M	1.354	1.725	4.464	4.187	1.494	0.913	0.886	0.844	2.553	5.901
	_0.462_	_0.607_	_0.658_	_0.681_	_0.706_	_0.709_	_0.707_	_0.676_	_0.604_	_0.486_
Beta	1.250	1.161	1.093	1.061	1.004	1.000	0.973	1.000	1.051	1.191
	_1.144_	_1.098_	_1.070_	_1.039_	_0.993_	_0.995_	_0.972_	_0.976_	_1.037_	_1.170_

* mean; ** median

Observations based on our results:

Comparing our results with those to the original paper, we observe that our results are quite close to the results published in the paper. NOA varies from the lowest of 25% to 148% in the highest decile. This suggests that high NOA firms might have had recent very rapid growth. Investors' perception on the sustainability of that growth comes into question. This confirms the finding of the author(s) finding on NOA.

There are several trends that confirm the other findings in the paper. First, from the lowest to highest NOA decile, the earning decreases. This earning performance also is coincident with the accruals across thee NOA decile. As also stated in the paper, we find a mixed sign of cash flow among NOA decile, and extreme (both high and low) NOA firms have the smallest size, as indicated by BV and B/M. Results of regression that gives us the Beta indicate that Beta is high at both ends of the decile portfolios. In the middle the Beta is either 1 or very close to 1. Combining our observations on MV, B/M, and beta, we confirm that "the extreme deciles seem to be small, possibly high growth orientated or overvalued, and risky firms", as concluded by the author(s).

The slight variations that we see in data can be attributed to following:

1) Quality of data set used for the original paper and our assignment
2) Number of years that have passed since the paper was originally written hence affecting the quality of data available now
3) Unstated assumptions by the author(s) while performing the analyses.

Problem Statement 2: Replicate Table 4 of the same paper

Following are the steps, assumptions and data related details that we encountered during our assignment:

Part 1 – Calculate accumulated return in previous year and future 3 years for each stock

- Order data of each stock with year and month descending ranked; we use descending ranked to calculate future average return as SAS can only look backwards not forwards when processing data
- Rank observations for each stock with time in descending order, with most recent month labeled as line 1
- Use a macro to calculate the cumulative return in 1, 2 and 3 years after each current month for each stock, and put the result to column leadret1, leadret2, and leadret3.
- Delete the most recent 3 years data of each stock
- Rank the observations for each stock with time in ascending order, with first month labeled as line 1
- Use a macro to calculate the cumulative return in previous 1 year (2-12 month) and add the result to column 'lagret'
- Because we are lack of lag data, the first 12 months observations for each stock are deleted
- adjust returns to monthly average and delete data out of range
- Sort the data in each year and month into 5 groups based on market cap.
- In each group we got, sort the data into 5 groups based on Book Value/Market Value ratio
- In each group we got from last step, we further sort the data into 5 groups based on the accumulate return in previous year. Thus, we get 125 groups (5*5*5) in each month
- Calculate sum of market cap in each group we got, and add the result back to our data

Part 2 - Calculate the equal weighted and value weighted benchmarks

- For value weighted benchmark(equal weighted benchmark uses the same steps): calculate the weight as market cap/sum of market cap
- Calculate weighted average return in future 3 years for each of 125 groups
- Merge the return back to our data file
- Calculate adjusted return against benchmark
- Adjusted return for each observation is calculated as the difference between average return in future 3 years of each stock and the average return of the group that the stock belongs to

Part 3 - Calculate raw return and adjust return of each NOA decile using equal weight and value weight

- Separate each NOA decile into a sub database for return calculation
- Calculate sum of market cap in each decile, and then get the weight of each stock in each NOA decile
- Calculate weighted average return in next year and weighted average adjusted return in future 3 years for each NOA decile
- Merge the results of the 10 NOA deciles under two weighting method into two tables: one for equal weighted, the other for value weighted. Here we get the upper part of Table 4, i.e. the first 10 rows.

Part 4 - Calculate L-H Hedge portfolio

- Merge the weighted return in the first and last NOA decile by match year and month
- Calculate the difference of weighted return in next year and the difference of weighted adjusted return in future 3 years between the first NOA decile and last NOA decile. We do the same process for both equal weighted and market cap weighted portfolios.
- Add the result to the end of the table we got from the previous part
- Get result: Calculate the mean of the return in next year, the adjusted return in future 3 years for each NOA decile and each weighting method. Output the result.

Part 5 – Alpha Calculation

- We calculate market, Fama-French, 4 factor time series using the code from last homework
- We import risk-free rate from St. Louis Fed
- We then regress the future 1-3 yr return time series on market excessive return, Fama-French factors, and 4 factors to get the alpha and t-stat

Here is the replicated Table 4 based on our code and data selected:

Table 4

	Equal Weighted				***Value Weighted***			
Portfolio ranking	*raw_ew t+1*	*adj_ew t+1*	*adj_ew t+2*	*adj_ew t+3*	*raw_vw t+1*	*adj_vw t+1*	*adj_vw t+2*	*adj_vw t+3*
Lowest	0.0137	0.0025	0.0024	0.0020	0.0113	0.0011	0.0008	0.0008
	12.75*	***6.44***	***9.83***	***10.38***	***12.16***	***2.43***	***3.13***	***4.29***
2	0.0134	0.0019	0.0014	0.0013	0.0114	0.0009	0.0009	0.0010
	13.37	***7.16***	***7.37***	***8.54***	***15.49***	***3.82***	***5.15***	***7.14***
3	0.0130	0.0014	0.0011	0.0007	0.0113	0.0010	0.0008	0.0009
	14.94	***6.48***	***6.7***	***7.34***	***14.06***	***2.33***	***3.9***	***6.47***
4	0.0116	0.0003	0.0005	0.0004	0.0104	0.0003	0.0004	0.0004
	14.05	***2.09***	***4.37***	***4.61***	***17.62***	1.76	***2.28***	***2.57***
5	0.0112	0.0002	0.0002	0.0002	0.0100	0.0002	0.0003	0.0004
	13.95	1.5	1.8	1.91	***16.72***	1.36	***2.65***	***3.8***
6	0.0113	-0.0001	-0.0002	-0.0002	0.0103	0.0002	-0.0000	-0.0003
	14.09	-0.74	-2.09	***-2.28***	***16.19***	1.45	-0.33	***-2.53***
7	0.0103	-0.0004	-0.0001	0.0000	0.0084	-0.0005	-0.0001	0.0000
	13.04	***-3.36***	-1.41	0.12	***13.35***	***-3.12***	-1.28	0.39
8	0.0092	-0.00124	-0.00111	-0.00099	0.008731	-0.00017	-0.00038	-0.0004
	11.38	***-10.5***	***-9.47***	***-10.76***	***13.88***	-0.9	***-2.63***	***-2.95***
9	0.0085	-0.0013	-0.0012	-0.0010	0.0076	-0.0011	-0.0009	-0.0008
	10.31	***-7.21***	***-10.18***	***-10.72***	***13.87***	***-5.59***	***-5.64***	***-5.74***
Highest	0.0058	-0.0031	-0.0027	-0.0021	0.0055	-0.0031	-0.0027	-0.00239
	5.63	***-11.68***	***-13.58***	***-13.56***	***6.64***	***-10.26***	***-10.33***	***-10.77***
Hedge (L-H)	0.0079	0.0056	0.0051	0.0040	0.0058	0.0042	0.0035	0.0032
	11.83	***11.06***	***13.87***	***14.7***	***6***	***6.45***	***8.27***	***9.5***
CAPM α	0.0079	0.0056	0.0051	0.0040	0.0058	0.0042	0.0036	0.0033
	11.66	***10.99***	***13.77***	***14.58***	***5.91***	***6.46***	***8.34***	***9.56***
3 Factor α	0.0077	0.0055	0.0051	0.0040	0.0056	0.0041	0.0036	0.0032
	11.53	***10.88***	***13.62***	***14.37***	***5.67***	***6.21***	***8.22***	***9.53***
4 factor α	0.0075	0.0053	0.0048	0.0038	0.0054	0.0038	0.0034	0.0032
	10.61	***9.89***	***12.26***	***13.10***	***5.18***	***5.51***	***7.44***	***8.82***

* *t*-statistic

Observations based on our results:

Comparing our results with those to the original paper, we observe that our results are quite close to the results published in the paper. The slight variations that we see in data can be attributed to following:

- From our result, the average monthly adjusted equally weighted return spread between lowest and highest NOA decile is 0.56%; in year *t*+2 the effect is 0.51%; in *t*+3 it is 0.40%; the value weighted return spreads are around 0.3%-0.4%
- From the *t* value, we can see that the return spreads are very significant. The *t*-statistics for the L-H portfolio is very significant, which means the NOA L-H has the superior performance.
- When equal-weighted, NOA L-H return is more significant than the value-weighted. While for both equal and value-weighted results, NOA L-H performs very well.
- Even after adjusting the size, book to market and the momentum effect, the NOA L-H return is still significant.
- NOA L-H return is highest in the year *t*+1 compared with *t*+2 or *t*+3. NOA L-H return is still significant in 3 years.

These results indicate that there is a significant monotonic trend between NOA and future average return. Lower NOA tends to have higher future average return to *t*+3, while higher NOA tends to have lower future average return to *t*+3. The abnormal returns in both ends are statistically significant at 5% significance level even after the adjustment of size, B/M, and past year returns. When using headging portfolio to fit into CAPM, 3-factor, and 4-factor model, significant α is yield at 5% significance level.

Appendix: SAS code

```
libname cleandat "C:\SAS Data\Data";
libname result "C:\SAS Data\Output";

/* Clean out log */
dm "out;clear;log;clear;";
%let raw_1 = Compustat_HW4;
%let raw_2 = CRSP_HW4;
%let raw_3 = NYSEAMEX;
%let fstartyr = 1963;  /* define start year of compustat data */
%let fendyr = 2000;    /* define end year of compustat data */
%let startyr = 1964;   /* define start year of crsp data */
%let startm = 6;     /* define start month of crsp data */
%let endyr = 2002;         /* define end year of crsp data */

/* input raw data */
/* input compustat */
/* select NYSE/AMEX/NASDAQ stocks
11 - NYSE, 12 - AMEX, 14 - NASDAQ */
data compustat(drop = CONM SEQ IB REVT);
      set cleandat.&raw_1;
      if EXCHG= 11 or EXCHG= 12 or EXCHG= 14;
run;

/* input crsp */
/* select NYSE/AMEX/NASDAQ stocks: 1 - NYSE, 2 - AMEX, 3 - NASDAQ */
data crsp; set cleandat.&raw_2;
      if EXCHCD = 1 or EXCHCD = 2 or EXCHCD = 3;
      month = month(date);
      year = year(date);
run;

/* input NYSE/AMEX return */
data NYSEAMEX(keep = year month mktret);
      set cleandat.&raw_3;
      year = year(caldt);
      month = month(caldt);
      rename ewretd = mktret;
run;



/***********************************************/
/* Table 1, Part I NOA and other fundamentals */
/***********************************************/

/* Input Compustat */
/*
Old   Description                       New
1       Cash and short Term Investment    CHE
4       Current Asset                       ACT
5       Current Liabilities               LCT
6       Total Asset                         AT
9       Long Term Debt                  DLTT
14      Depreciation and Amortization         DP
```

```
25      Shares Outstanding                CSHO
34      Debt included in current liabilities DLC
38      Minor Interests                 MIB
60      Book Value of Common Equity     CEQ
71      Income Tax Payable                TXP
130     Preferred Stocks                      PSTK
178     Income From Continuing Operations OIADP
199     Fiscal Year End Closing Price         PRCC_F
*/

/* ********************************************************* */

/* clean compustat data */
data compustat (drop = gvkey_char);
      set compustat (rename = (gvkey = gvkey_char));
      cusip = substr(cusip,1,8);    /* reshape cusip to match crsp */
      gvkey = gvkey_char*1;   /* change gvkey to number */
      month = month(datadate);
      year = year(datadate);
      lagAT = lag(AT);  /* get total asset of last year */
    rename CEQ = BV;  /* book value of common equity */
run;

/* if some variables are missing, they can be reasonalby assumed
to be 0. see page 17 */
data compustat; set Compustat;
      if missing(DLC) then DLC = 0;
      if missing(TXP) then TXP = 0;
      if missing(DLTT) then DLTT = 0;
      if missing(MIB) then MIB = 0;
      if missing(PSTK) then PSTK = 0;
run;

/* if some variables are missing or negative, they must be dropped */
data compustat; set Compustat;
      if missing(CHE) = 0 and missing(ACT) = 0 and missing(LCT) = 0
         and missing(AT) = 0 and missing(DP) = 0 and missing(CSHO) = 0
         and missing(BV) = 0 and missing(OIADP) = 0 and missing(PRCC_F) = 0
         and missing(lagAT) = 0
         and AT > 0 and lagAT > 0 and CSHO > 0 and BV > 0 and PRCC_F > 0;
run;

/* ********************************************************* */

/* calculate fundamentals, meanings of variables see above chart and page 44 of the
paper*/
data compustat; set compustat;
    OA = AT-CHE;
    OL = AT-DLC-DLTT-MIB-PSTK-BV;
      RawNOA = OA-OL;
    NOA = RawNOA/lagAT;
      Earnings = OIADP/lagAT;
      Rawaccruals = (ACT-lag(ACT))-(CHE-lag(CHE))-(LCT-lag(LCT))
                          +(DLC-lag(DLC))+(TXP-lag(TXP))-DP;
      /* be cautious to use lag function as the first line of each gvkey is wrong
         will be adjusted later */
      Accruals = Rawaccruals/lagAT;
      Cashflows = Earnings-Accruals;
    MV = PRCC_F*CSHO;   /* market cap */
      Cash = CHE/lagAT;
      Equity = BV/lagAT;
      Debt = NOA-(Equity+Cash);
      BM = BV/MV; /* book to market */
run;

/* clean NOA and first line problem */
data compustat; set compustat; if missing(NOA) = 0; run;
/* delete the first row of each gvkey because its lag numbers are wrong */
proc sort data = compustat; by by gvkey year; run;
data compustat; set compustat; by gvkey year;
      if first.gvkey and first.year then delete;
run;

/* output necessary variables for Table 1 Part I calculation */
/* table_1 is for the calculation of Table 1 Part I */
data table_1 (keep = gvkey datadate fyear year month TIC cusip NOA
                               Earnings Accruals Cashflows BV MV BM);
      set compustat;
      if fyear >= &fstartyr and fyear <= &fendyr;
      /* use the same data range with the paper after we clean the data */
run;

/* ******************************************************** */

/* rank by NOA in each fiscal year */
proc sort data = table_1; by fyear; run;

proc rank data = table_1 out = table_1 group = 10;
      var NOA; by fyear;
      ranks NOA_rank;
run;

/* calculate mean and median stat of each NOA decile */
proc sort data = table_1; by NOA_rank; run;

proc means data = table_1 noprint;
      var NOA Earnings Accruals Cashflows BV MV BM;
      by NOA_rank;
      output out = meanstat mean = meanNOA meanEarning meanAccruals meanCashflows
meanBV meanMV meanBM;
run;

proc means data = table_1 noprint;
      var NOA Earnings Accruals Cashflows BV MV BM;
      by NOA_rank;
      output out = medianstat median = medianNOA medianEarning medianAccruals
medianCashflows medianBV medianMV medianBM;
run;

/* Output results */
proc export data = meanstat outfile = "C:\SAS Data\Output\HW4_meanstat.csv" DBMS =
csv replace; run;

```

```
proc export data = medianstat outfile = "C:\SAS Data\Output\HW4_medianstat.csv"
DBMS = csv replace; run;



/********************************************/
/* Table 1, Part II Beta of each NOA decile */
/********************************************/

/* merge compustat with crsp */
/* crspmerge is used to merge with compustat */
data crspmerge (keep = permno cusip date year month ret);
      set crsp;
      if missing(dlret) = 0 then ret = dlret;
      if missing(ret) or ret < -1 then delete;
run;

proc sort data = crspmerge; by permno year month; run;

/* prepare compustat for merge, compustatmerge is used to merge with crspmerge */
/* move month of report date forward 4 month, the finnancial data will be used
since this month and onward, as assumed in pate 23 */
data compustatmerge (keep = gvkey datadate fyear year fmonth cusip NOA MV BM);
      set compustat;
      if month >= 1 and month <= 8 then month = month+4;
      else if month >= 9 and month <= 12 then do;
            month = month - 8;
            year = year+1;
      end;
      rename month = fmonth;  /* this is the month financial statement used by
investor */
run;

/* Add permnos to compustat by matching cusips */
data cusips (keep = permno cusip year); set crsp;
      if month = 1;
run;

proc sort data = cusips; by cusip year; run;
proc sort data = compustatmerge; by cusip year; run;

data compustatmerge;
      merge compustatmerge(in = k) cusips;
      by cusip year;
      if k;
run;

/* merge crspmerge and compustatmerge by permno
we think the reason not use cusip to merge is that cusip is not perfectly number
fin_ret contains financial data and returns */
proc sort data = compustatmerge; by permno year; run;
data fin_ret;
      merge crspmerge compustatmerge(in = k);
      by permno year;
      if k;
run;

```

```
/* delete data that does no have valid permno (cannot be identified) */
data fin_ret; set fin_ret; if missing(permno) = 0; run;

/* adjust financial data. when merge, all physical year has same financial data
however, investors only use the financial data after fmonth (when financial data
of that year come out, before that month, invesor use last year financial data */
proc sort data = fin_ret; by permno year month; run;
%macro fin_adj;
%do iter = 1 %to 11;
data fin_ret; set fin_ret;
      if fmonth = &iter+1 and month < fmonth then do;
      /* if month < fmonth, investor need to use last year data */
            NOA = lag&iter(NOA);
            MV = lag&iter(MV);      /* market cap in the end of fiscal year */
            BM = lag&iter(BM);      /* book to market in the end of fiscal year */
      end;
%end;
%mend;
%fin_adj;

/* delete the data that is not correct */
data fin_ret; set fin_ret;
      by permno year month;
      if (first.permno or first.year) and month < fmonth then delete;
      /* during the first year and before the fmonth, investor has no financial
data */
      if missing(NOA) then delete;
      if missing(MV) or MV <= 0 then delete;
      if missing(BM) or BM <=0 then delete;
      /* BV (book value) and BM (book to market) are from compustat and are in
      the end of fiscal year, as stated by the paper */
run;

/* after clean the data, use the data in the data range as stated in the paper
from now fin_ret contains cleaned return and financial data with financial data
at the right month that the investor should begin to use */
data fin_ret; set fin_ret; if year >= &startyr and year <= &endyr; run;
data fin_ret; set fin_ret; if year = &startyr and month < &startm then delete; run;

/* ********************************************************* */

/* rank NOA for each month */
proc sort data = fin_ret; by year month; run;
proc rank data = fin_ret out = fin_ret group = 10;
      var NOA; by year month;
      ranks NOA_rank;
run;

proc sort data = NYSEAMEX; by year month; run;

/* betacal is for beta calculation */
data betacal(keep = permno year month NOA_rank ret); set fin_ret; run;

/* sort betacal in descending order for id identification (prior 60mon return) */
proc sort data = betacal; by permno descending year month; run;

/* this macro is to calculate the beta of each NOA decile in each month. the idea
```

```
is in each month, identify the stocks that belongs to on NOA decile and then
retrieve
the previous 60mon returns. use these returns to get equal weighted historical
return and
regress against NYSE/AMEX equal weighted index to get the beta of that NOA decile
in that month */
/* WARNING: This is a big loop and will take a lot of time.  Please reduce the loop
of
rank, year, and m to save time if you just want to test (e.g. set y = 1980 to 1981)
*/
%macro beta_cal;
%do rank = 0 %to 9;     /* loop of NOA decile */

data NOA_&rank._beta; run;
/* create empty dataset to store beta of each month for that NOA decile */

%do y = &startyr+5 %to &endyr;
/* %do y = &startyr+5 %to &endyr; */
      %do m = 1 %to 12; /* loop of year and month */

      data betacal; set betacal;
            by permno descending year month;
            retain id;
            /* use id to identify the returns that we want to use for regression */
            if year = &y and month = &m and NOA_rank = &rank then id = 1;
            else if first.permno = 0 then id = id+1;
            else if first.permno then id = 61;
      run;

      /* drop id to prevent interfere to the next calculation */
      /* id < 60 are the recent 60 historical returns that we will use for the
regression
         to get the beta of this NOA decile in this month */
      data beta_sub(drop = id); set betacal; if missing(id) = 0 and id <= 60; run;
      data betacal(drop = id); set betacal; run;

      proc sort data = beta_sub; by year month; run;

      /* calculate monthly equal weighted return of NOA portfolio */
      proc means data = beta_sub noprint;
            var ret; by year month;
            output out = return mean = NOAret;
      run;

      /* merge NOA portfolio return with NYSE/AMEX equal weighted return in
regression time period */
      data return (drop = _TYPE_ _FREQ_);
            merge return(in = k) NYSEAMEX;
            by year month;
            if k;
      run;

      /* delete data that is obviously wrong */
      data return; set return;
            if (&y-year)*12+(&m-month) < 0 or (&y-year)*12+(&m-month) >= 60
                  or missing(NOAret) or missing (mktret) then delete;
      run;

      /* regress NOA portfolio on NYSE/AMEX index, regression results are stored in
est dataset */
      proc reg data = return outest = est noprint;
            model NOAret = mktret;
      run;
      quit; /* stop the regression once it is done to speed up the performance */

      /* store all beta in all month for this NOA decile in one dataset
(NOA_&rank.beta)
      the beta (coefficient of mktret) is store as mktret variable in est dataset
*/
      data NOA_&rank._beta; set NOA_&rank._beta est(keep = mktret); run;
      %end;
%end;

/* rename the beta variable */
data NOA_&rank._beta; set NOA_&rank._beta;
      rename mktret = NOA_&rank;
run;
data NOA_&rank._beta; set NOA_&rank._beta;
      if missing(NOA_&rank) = 0;
      line = _n_;
run;
%end;
%mend;
%beta_cal;

/* ********************************************************** */

/* NOA_beta stores beta of each month for all NOA decile */
data NOA_beta; set NOA_0_beta; run;
%macro beta_merge;
%do iter = 1 %to 9;
data NOA_beta;
      merge NOA_beta NOA_&iter._beta;
      by line;
run;
%end;
%mend;
%beta_merge;

proc means data = NOA_beta noprint;
      var NOA_0 NOA_1 NOA_2 NOA_3 NOA_4 NOA_5 NOA_6 NOA_7 NOA_8 NOA_9;
      /* these are the betas for each NOA decile */
      output out = betamean mean = NOA_0 NOA_1 NOA_2 NOA_3 NOA_4 NOA_5 NOA_6 NOA_7
NOA_8 NOA_9;
run;

proc means data = NOA_beta noprint;
      var NOA_0 NOA_1 NOA_2 NOA_3 NOA_4 NOA_5 NOA_6 NOA_7 NOA_8 NOA_9;
      output out = betamedian median = NOA_0 NOA_1 NOA_2 NOA_3 NOA_4 NOA_5 NOA_6
NOA_7 NOA_8 NOA_9;
run;

/* Output results */
```

```
proc export data = betamean outfile = "C:\SAS Data\Output\HW4_betamean.csv" DBMS =
csv replace; run;
proc export data = betamedian outfile = "C:\SAS Data\Output\HW4_betamedian.csv"
DBMS = csv replace; run;



/**************************************************************/
/* Table 4, Part I. Abnormal returns of NOA decile portfolios */
/**************************************************************/

/* calculate next 1yr, 2yr, 3yr return for each NOA decile - leadret1 leadret2
leadret3 */
/* fin_ret is created in Table 1, Part II, sort descending to calculate future
average return */
proc sort data = fin_ret; by permno descending year month; run;

data fin_ret; set fin_ret;
      by permno descending year month;
      leadret1 = 0; leadret2 = 0; leadret3 = 0;
      retain line;
      if first.permno then line = 1;
      else line = line+1;
      /* use line to control time step */
run;

/* This macro is used to calculate next 1yr, 2yr, 3yr return - leadret1 leadret2
leadret3 */
%macro lead_cal;
%do iter = 1 %to 36;
data fin_ret; set fin_ret;
      by permno descending year month;
      if &iter <= 12 then leadret1 = leadret1+lag&iter(ret);
      /* accumulative return from t+1 to t+12 month */
      if &iter <= 24 then leadret2 = leadret2+lag&iter(ret);
      /* accumulative return from t+1 to t+24 month */
      leadret3 = leadret3+lag&iter(ret);
      /* accumulative return from t+1 to t+36 month */
%end;
%mend;
%lead_cal;

/* the first 36 lines of each stock are wrong, as in these time period there are no
sufficient data for t+3 return */
data fin_ret (drop = line); set fin_ret; if line <= 36 then delete; run;

/* perform similar calculation to calculate cumulative returns from t-2 to t-12
month */
proc sort data = fin_ret; by permno year month; run;

data fin_ret; set fin_ret;
      by permno year month;
      lagret = 0;
      retain line;
      if first.permno then line = 1;
      else line = line+1;
      /* use line to control the calculation of PR1YR */
```

```
run;

/* This macro is used to calculate cumulative returns from t-2 to t-12 month */
%macro lag_cal;
%do iter = 2 %to 12;
data fin_ret; set fin_ret;
      by permno year month;
      lagret = (1+lagret)*(1+lag&iter(ret))-1;
      /* cumulative return from t-2 to t-12 month */
%end;
%mend;
%lag_cal;

/* the first 12 lines are calculated wrong */
data fin_ret (drop = line); set fin_ret; if line <= 12 then delete; run;

/* adjust returns to monthly average and delete data out of range */
data fin_ret;
      set fin_ret;
      leadret1 = leadret1/12; /* future average is athimetic average */
      leadret2 = leadret2/24;
      leadret3 = leadret3/36;
run;

/* ********************************************************* */

/* rank size, book/mkt, and PR1YR of each month, size(MV) and b/m(BM)
are value in each fiscal year end */
/* each month has 125 groups, this may lead to the problem that some
group only have one stock */
proc sort data = fin_ret; by year month; run;
proc rank data = fin_ret out = fin_ret group = 5;
      var MV; by year month;
      ranks size_rank;
run;

proc sort data = fin_ret; by year month size_rank; run;
proc rank data = fin_ret out = fin_ret group = 5;
      var BM; by year month size_rank;
      ranks BM_rank;
run;

proc sort data = fin_ret; by year month size_rank BM_rank; run;
proc rank data = fin_ret out = fin_ret group = 5;
      var lagret; by year month size_rank BM_rank;
      ranks ret_rank;
run;

/* calculate sum mktcap of 125 groups */
proc sort data = fin_ret; by year month size_rank BM_rank ret_rank; run;

proc means data = fin_ret noprint;
            var MV; by year month size_rank BM_rank ret_rank;
            output out = sumstat sum = sumcap;
run;

data fin_ret;
```

```
      merge fin_ret sumstat(drop = _TYPE_);
      by year month size_rank BM_rank ret_rank;
run;

/* ********************************************************* */

/* calculate S/BM/Mom benchmark return for t+1, t+2, t+3 */
/* This macro calculates the equal-weighted and value weighted benchmarks S/BM/Mom
*/
%macro benchmark_ret(datain = );
/* use datain to control input data set */
%do code = 1 %to 2;
      data retcomp; set &datain; run;

      data retcomp; set retcomp;
            if &code = 1 then wt = MV/sumcap;
            else if &code = 2 then wt = 1/_FREQ_;
      run;

      /* Calculate benchmark weighted returns */
      proc sort data = retcomp; by year month size_rank BM_rank ret_rank; run;

      proc means data = retcomp noprint;
            var leadret1 leadret2 leadret3;
            weight wt;
            by year month size_rank BM_rank ret_rank;
            output out = return mean = bmkret1_&code bmkret2_&code bmkret3_&code;
      run;

      /* Output Returns */
      data return(drop = _TYPE_ _FREQ_); set return;
            if missing(bmkret1_&code) = 0 and missing(bmkret2_&code) = 0 and
               missing(bmkret3_&code) = 0 and missing(month) = 0;
      run;

      data &datain;
            merge &datain return;
            by year month size_rank BM_rank ret_rank;
      run;
%end;
%mend;
%benchmark_ret(datain = fin_ret);

/* calculate adjusted return against benchmark */
data fin_ret (drop = sumcap _FREQ_); set fin_ret;
      /* _1 is value/cap weighted */
      adjret1_1 = leadret1-bmkret1_1;
      adjret2_1 = leadret2-bmkret2_1;
      adjret3_1 = leadret3-bmkret3_1;
      /* _2 is equal weighted */
      adjret1_2 = leadret1-bmkret1_2;
      adjret2_2 = leadret2-bmkret2_2;
      adjret3_2 = leadret3-bmkret3_2;
      if _FREQ_ <= 1 then delete;
      /* some groups only have one stocks, it is excluded */
run;

```

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

/* calculate raw return and adjust return of each NOA decile */
/* monthly NOA rank are set in Table 1 Part II */

/* this macro is listed first and will be used in the next macro */
/* calculate the equal weighted and value weighted average future return of the NOA
decile */
/* there are several weighted return macros in this file, we don't have time to
consolidate it in to one as we did last time */
%macro NOA_ret(datain = , rank = );
/* use datain to control input data set */
%do code = 1 %to 2;
      data retcomp; set &datain; run;

      /* select weighting method */
      data retcomp; set retcomp;
            if &code = 1 then wt = MV/sumcap;
            else if &code = 2 then wt = 1/_FREQ_;
      run;

      proc sort data = retcomp; by year month; run;

      /* calculate monthly weighted returns */
      proc means data = retcomp noprint;
            var leadret1 adjret1_&code adjret2_&code adjret3_&code;
            weight wt; by year month;
            output out = return mean = leadret1 adjret1 adjret2 adjret3;
      run;

      /* output returns */
      data ret_&datain._&code(drop = _TYPE_ _FREQ_);
            set return; NOA_rank = &rank;
            if missing(leadret1) = 0 and missing(adjret1) = 0 and
               missing(adjret2) = 0 and missing(adjret3) = 0 and
               missing(month) = 0;
      run;
%end;
%mend;

/* seperate each NOA decile in to each sub database and calculate return */
%macro NOA_decile;
%do iter = 0 %to 9;
      data NOA_&iter; set fin_ret;
            if NOA_rank = &iter;
      run;

      proc sort data = NOA_&iter; by year month; run;

      proc means data = NOA_&iter noprint;
            var MV; by year month;
            output out = sumstat sum = sumcap;
      run;

      data NOA_&iter;
            merge NOA_&iter sumstat(drop = _TYPE_);
            by year month;
      run;

      %NOA_ret(datain = NOA_&iter, rank = &iter);
%end;
%mend;
%NOA_decile;

/* ********************************************************* */

/* vertical merge return data */
data NOA_cap; set ret_NOA_0_1; run; /* cap use all _1 */
data NOA_eq; set ret_NOA_0_2; run;  /* eq use all _2 */

%macro NOA_merge;
%do iter = 1 %to 9;
      data NOA_cap; set NOA_cap ret_NOA_&iter._1; run;
      data NOA_eq; set NOA_eq ret_NOA_&iter._2; run;
%end;
%mend;
%NOA_merge;

/* calculate L-H for eq and cap, assign it to NOA_rank = 10 */
/* cap weighted */
data ret_NOA_9_1; set ret_NOA_9_1;
      rename leadret1 = leadret1_9;
      rename adjret1 = adjret1_9;
      rename adjret2 = adjret2_9;
      rename adjret3 = adjret3_9;
run;

data ret_NOA_10_1;
      merge ret_NOA_0_1 (drop = NOA_rank) ret_NOA_9_1 (drop = NOA_rank);
      by year month;
run;

data ret_NOA_10_1 (drop = leadret1_9 adjret1_9 adjret2_9 adjret3_9);
      set ret_NOA_10_1;
      leadret1 = leadret1-leadret1_9;
      adjret1 = adjret1-adjret1_9;
      adjret2 = adjret2-adjret2_9;
      adjret3 = adjret3-adjret3_9;
      NOA_rank = 10;
run;
/* vertical merge NOA_10_1 to NOA_cap */
data NOA_cap; set NOA_cap ret_NOA_10_1; run;

/* equal weighted */
data ret_NOA_9_2;
      set ret_NOA_9_2;
      rename leadret1 = leadret1_9;
      rename adjret1 = adjret1_9;
      rename adjret2 = adjret2_9;
      rename adjret3 = adjret3_9;
run;

data ret_NOA_10_2;
      merge ret_NOA_0_2 (drop = NOA_rank) ret_NOA_9_2 (drop = NOA_rank);
      by year month;
run;

data ret_NOA_10_2 (drop = leadret1_9 adjret1_9 adjret2_9 adjret3_9);
      set ret_NOA_10_2;
      leadret1 = leadret1-leadret1_9;
      adjret1 = adjret1-adjret1_9;
      adjret2 = adjret2-adjret2_9;
      adjret3 = adjret3-adjret3_9;
      NOA_rank = 10;
run;
/* vertical merge NOA_10_2 to NOA_eq */
data NOA_eq; set NOA_eq ret_NOA_10_2; run;

/* ********************************************************* */

/* calculate the average weighted returns of each NOA decile */
proc sort data = NOA_cap; by NOA_rank year month; run;
proc means data = NOA_cap noprint;
      var leadret1 adjret1 adjret2 adjret3;
      by NOA_rank;
      output out = table4_cap mean = leadret1 adjret1 adjret2 adjret3;
run;
proc export data = table4_cap outfile =
            "C:\SAS Data\Output\HW4_table4_cap.csv" DBMS = csv replace; run;

proc sort data = NOA_eq; by NOA_rank year month; run;
proc means data = NOA_eq noprint;
      var leadret1 adjret1 adjret2 adjret3;
      by NOA_rank;
      output out = table4_eq mean = leadret1 adjret1 adjret2 adjret3;
run;
proc export data = table4_eq outfile =
            "C:\SAS Data\Output\HW4_table4_eq.csv" DBMS = csv replace; run;

/* ********************************************************* */

/* calculate the t-stat */
proc ttest data = NOA_cap;
      var leadret1 adjret1 adjret2 adjret3;
      by NOA_rank;
      ods output Ttests = NOA_cap_ttest;
run;

proc export data = NOA_cap_ttest outfile =
            "C:\SAS Data\Output\HW4_NOA_cap_ttest.csv" DBMS = csv replace; run;

proc ttest data = NOA_eq;
      var leadret1 adjret1 adjret2 adjret3;
      by NOA_rank;
      ods output Ttests = NOA_eq_ttest;
run;

proc export data = NOA_eq_ttest outfile =
            "C:\SAS Data\Output\HW4_NOA_eq_ttest.csv" DBMS = csv replace; run;
```

```

/***************************/
/* Table 4, Part II Alphas */
/***************************/

/* construct market/FAMA-French/four factor portforlio */
/* FAMA-French/four factor portforlio are constructed using code of last hw */
data compustat(drop = datadate gvkey_char);
      set cleandat.&raw_1(rename = (gvkey = gvkey_char));
      rename ib = income;
      rename seq = book;
      rename fyear = year;
      rename tic = comp_ticker;
      cusip = substr(cusip,1,8);
      gvkey = gvkey_char*1;
      if fyear >= &startyr-2 and fyear <= &endyr-2;
run;

/* Input CRSP */
data crsp(drop = date dlret prc shrout); set cleandat.&raw_2;
      month = month(date);
      year = year(date);
      if missing(dlret) = 0 then ret = dlret;
      if ret < -1 then delete;
      mktcap = abs(prc)*abs(shrout);
      if year >= &startyr-2 and year <= &endyr;
run;

/* Add permnos to compustat by matching cusips */
data cusips(keep = permno cusip ticker year); set crsp;
      if month = 1;
run;

proc sort data = cusips; by cusip year; run;
proc sort data = compustat; by cusip year; run;

data compustat;
      merge compustat(in = k) cusips;
      by cusip year;
      if k;
      year = year+2;
run;

/* Add Market cap (Dec of y-1) data to financial data */
data mktcap_dec(keep = year permno mktcap_dec); set crsp;
      if month = 12;
      year = year+1;
      rename mktcap = mktcap_dec;
run;

proc sort data = mktcap_dec; by year permno; run;
proc sort data = compustat; by year permno; run;

data universe;
      merge compustat(in = k) mktcap_dec;
      by year permno;
      if k;
```

```
run;

proc means data = universe;
      var mktcap_dec book income;
      by year;
      output out = sumstat sum = sumcap sumbook sumincome;
run;

data universe;
      merge universe sumstat(drop = _TYPE_);
      by year;
run;

/* Calculate book/cap value */
data universe; set universe; BKMK = book/mktcap_dec; run;

/* Rank market cap and book/cap */
proc rank data = universe out = universe group = 10;
      var mktcap_dec; by year;
      ranks cap_rank;
run;

proc rank data = universe out = universe group = 10;
      var BKMK; by year;
      ranks BKMK_rank;
run;

/* Merge universe with return data from CRSP */
proc sort data = crsp(keep = permno year month ret retx) out = crsp_subset;
      by permno year;
run;
proc sort data = universe; by permno year; run;

data universe;
      merge crsp_subset universe(in = k);
      by permno year;
      if k;
run;

/* ********************************************************* */

/* this macro calculates weighted average return */
%macro ret_cal(datain = , start_code = , end_code = , rebal = , rebal_mon = );
/* use datain to control input data set */
/* use start_code and end_code to control weighting method */
/* code 1-cap weighted; 2-equal weighted; 3-earning weighted; 4-book weighted */
/* rebal control rebalance frequency. 1-annually; 2-monthly */
/* rebal_mon control on which month to perform annually rebalance */

%do code = &start_code %to &end_code;
      data retcomp; set &datain; run;

      data retcomp; set retcomp;
            if &code = 1 then weight = mktcap_dec/sumcap;
            else if &code = 2 then weight = 1/_FREQ_;
            else if &code = 3 then weight = income/sumincome;
            else if &code = 4 then weight = book/sumbook;
      run;

      /* Calculate dynamic weights */
      proc sort data = retcomp; by permno year month; run;

      data retcomp; set retcomp;
            by permno year month;
            lagretx = lag(retx);
            if first.permno then lagretx = 0;
      run;

      /* Calculate rebalance */
      %if &rebal = 1 %then %do;
      /* &rebal = 1,Annual rebalance */
            data retcomp; set retcomp;
                  by permno year month;
                  retain dyn_wt;
                  if first.permno or month = &rebal_mon then dyn_wt = weight;
                  /* rebalance in the rebal_mon month */
                  else dyn_wt = dyn_wt*(1+lagretx);   /* Rebalance of the portfolio
*/
            run;
      %end;
      %else %if &rebal = 2 %then %do;
      /* &rebal = 2, Monthly rebalance */
            data retcomp; set retcomp;
                  dyn_wt = weight;
            run;
      %end;

      /* Calculate Returns */
      proc sort data = retcomp; by year month; run;

      proc means data = retcomp noprint;
            var dyn_wt; by year month;
            output out = sumwt sum = sumwt;
      run;

      data retcomp(drop = _TYPE_ _FREQ_ sumwt);
            merge sumwt retcomp; by year month;
            dyn_wt = dyn_wt/sumwt;
      run;

      proc means data = retcomp noprint;
            var ret retx;
            weight dyn_wt;
            by year month;
            output out = return mean = TR_&datain PR_&datain;
      run;

      /* Output Returns */
      data ret_&datain._&code._&rebal(drop = _TYPE_ _FREQ_);
            set return;
            if missing(TR_&datain) = 0 and missing(month) = 0;
      run;
%end;
%mend;
```

```

/* Market/CRSP annual rebalance */
%ret_cal(datain = universe, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 1);

/* Get market return */
proc export data = ret_universe_1_1 outfile = "C:\SAS Data\Output\mkt_ret.csv" DBMS
= csv replace; run;

/* ****************************************************** */

/* Fama-French */
/* FF portfolio, drop data that do not have marketcap or book data in T, T-1, and
T-2 or do not have active market return */
data ff; set universe;
      if missing(mktcap_dec) = 0 and missing(book) = 0 and
         missing(lag1(mktcap_dec)) = 0 and missing(lag1(book)) = 0 and
         missing(lag2(mktcap_dec)) = 0 and missing(lag2(book)) = 0 and
         missing(ret) = 0;
run;

/* Create big-growth, big-neutral, big-value, small-growth, small-neutral, small-
value portfolio */
data ff_bg; set ff; if cap_rank >= 8 and BKMK_rank >= 0 and BKMK_rank <= 2; run;
data ff_bn; set ff; if cap_rank >= 8 and BKMK_rank >= 3 and BKMK_rank <= 6; run;
data ff_bv; set ff; if cap_rank >= 8 and BKMK_rank >= 7 and BKMK_rank <= 9; run;
data ff_sg; set ff; if cap_rank >= 0 and cap_rank <= 7 and BKMK_rank >= 0 and
BKMK_rank <= 2; run;
data ff_sn; set ff; if cap_rank >= 0 and cap_rank <= 7 and BKMK_rank >= 3 and
BKMK_rank <= 6; run;
data ff_sv; set ff; if cap_rank >= 0 and cap_rank <= 7 and BKMK_rank >= 7 and
BKMK_rank <= 9; run;

/* Size and Value portfolios are reconstituted annually at the end of June, returns
are cap weighted */
%ret_cal(datain = ff_bg, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 6);
%ret_cal(datain = ff_bn, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 6);
%ret_cal(datain = ff_bv, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 6);
%ret_cal(datain = ff_sg, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 6);
%ret_cal(datain = ff_sn, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 6);
%ret_cal(datain = ff_sv, start_code = 1, end_code = 1, rebal = 1, rebal_mon = 6);

/* Momentum */
/* Momentum portfolio, drop data that does not have mktcap and return value */
/* first 12 month are dropped later */
data mom; set universe;
      if missing(mktcap_dec) = 0 and missing(ret) = 0;
run;

proc sort data = mom; by permno year month; run;

data mom; set mom;
      by permno year month;
      cum_ret = 0;
      retain line;
      if first.permno then line = 1;
      else line = line+1;
      /* use line to control the calculation of PR1YR */
```

```
run;

/* This macro is used to calculate cumulative returns from t-2 to t-12 */
%macro cum_cal;
%do iter = 2 %to 12;
data mom; set mom;
      by permno year month;
      cum_ret = (1+cum_ret)*(1+lag&iter(ret))-1;
      /* accumulative return from t-2 to t-12 */
      %end;
%mend cum_cal;
%cum_cal;

data mom; set mom; if line >= 13; run;
/* drop the first 12 month */

proc sort data = mom; by year month; run;
proc rank data = mom out = mom group = 10;
      var cum_ret; by year month;
      ranks ret_rank;
run;

/* Create big-up, big-down, small-up, small-down portfolio */
data mom_bd; set mom; if cap_rank >= 8 and ret_rank >= 0 and ret_rank <= 2; run;
data mom_bu; set mom; if cap_rank >= 8 and ret_rank >= 7 and ret_rank <= 9; run;
data mom_sd; set mom; if cap_rank >= 0 and cap_rank <= 7 and ret_rank >= 0 and
ret_rank <= 2; run;
data mom_su; set mom; if cap_rank >= 0 and cap_rank <= 7 and ret_rank >= 7 and
ret_rank <= 9; run;

/* After getting the portfolio, redo the monthly rebalanced, cap weighting return
*/
%ret_cal(datain = mom_bd, start_code = 1, end_code = 1, rebal = 2, rebal_mon = 1);
%ret_cal(datain = mom_bu, start_code = 1, end_code = 1, rebal = 2, rebal_mon = 1);
%ret_cal(datain = mom_sd, start_code = 1, end_code = 1, rebal = 2, rebal_mon = 1);
%ret_cal(datain = mom_su, start_code = 1, end_code = 1, rebal = 2, rebal_mon = 1);

/* Consolidate sub-FF and sub-Momentum portfolio */
/* Merge calculation results */
data ff_ret;
      merge ret_ff_bg_1_1 ret_ff_bn_1_1 ret_ff_bv_1_1
              ret_ff_sg_1_1 ret_ff_sn_1_1 ret_ff_sv_1_1;
              /* these datasets are generated by ret_cal macro */
      by year month;
run;

data mom_ret;
      merge ret_mom_bu_1_2 ret_mom_bd_1_2 ret_mom_su_1_2 ret_mom_sd_1_2;
      /* these datasets are generated by ret_cal macro */
      by year month;
run;

/* Calculate SMB, HML, MOM */
data ff_ret; set ff_ret;
      smb = 1/3*(TR_ff_sv+TR_ff_sn+TR_ff_sg)-1/3*(TR_ff_bv+TR_ff_bn+TR_ff_bg);
      hml = 1/2*(TR_ff_bv+TR_ff_sv)-1/2*(TR_ff_bg+TR_ff_sg);
run;
```

```

data mom_ret; set mom_ret;
      mom = 1/2*(TR_mom_bu+TR_mom_su)-1/2*(TR_mom_bd+TR_mom_sd);
run;

data ffm (keep = year month smb hml mom);
      merge ff_ret mom_ret;
      by year month;
run;

proc export data = ffm outfile = "C:\SAS Data\Output\ffm.csv" DBMS = csv replace;
run;

/* ******************************************************** */

/* calculate alpha using hedge NOA portfolio in Table 2 Part I */
/* get rf from outside */
proc import datafile = "C:\SAS Data\Data\riskfree.csv"
      out = rf DBMS = csv replace;
run;

data rf; set rf;
      year = substr(date,5,6);
      month = substr(date,11,2);
run;

data rf(drop = date year_char month_char);
      set rf(rename = (year = year_char month = month_char));
      year = year_char*1;
      month = month_char*1;
      rf = rf/100;
      if year >= &startyr and year <= &endyr;
run;

/* This macro is to do the regression on CAPM, FF, and 4-factor. It returns alpha
and t-stat */
/* 1 - value weighted; 2 - equal weighted */
%macro factor_cal;
%do iter = 1 %to 2;

data factor_&iter; merge ret_NOA_10_&iter rf; by year month; run;
data factor_&iter; merge factor_&iter ret_universe_1_1; by year month; run;
data factor_&iter; merge factor_&iter ffm; by year month; run;

data factor_&iter; set factor_&iter;
      if missing(leadret1) or missing(adjret1) or missing(adjret2) or
missing(adjret3)
         or missing(rf) or missing(TR_universe) or missing(smb) or missing(hml)
         or missing(mom) then delete;
run;

/* mktex is the market excessive return */
data factor_&iter; set factor_&iter; mktex = TR_universe-rf;

data alpha_&iter; run;

/* regress CAPM */
```

```
proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model leadret1 = mktex;
run; quit; /* stop the regression once it is done to speed up the performance */
data alpha_&iter; merge alpha_&iter est(keep = _TYPE_ intercept); rename intercept
= lead1_CAPM; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret1 = mktex;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj1_CAPM; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret2 = mktex;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj2_CAPM; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret3 = mktex;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj3_CAPM; run;

/* regress FF */
proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model leadret1 = mktex smb hml;
run; quit; /* stop the regression once it is done to speed up the performance */
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
lead1_FF; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret1 = mktex smb hml;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj1_FF; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret2 = mktex smb hml;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj2_FF; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret3 = mktex smb hml;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj3_FF; run;

/* regress FF+Mom */
proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model leadret1 = mktex smb hml mom;
run; quit; /* stop the regression once it is done to speed up the performance */
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
lead1_FFM; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
```

```
      model adjret1 = mktex smb hml mom;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj1_FFM; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret2 = mktex smb hml mom;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj2_FFM; run;

proc reg data = factor_&iter outest = est TABLEOUT noprint;
      model adjret3 = mktex smb hml mom;
run; quit;
data alpha_&iter; merge alpha_&iter est(keep = intercept); rename intercept =
adj3_FFM; run;

proc export data = alpha_&iter outfile = "C:\SAS Data\Output\HW4_alpha_&iter..csv"
DBMS = csv replace; run;
%end;
%mend;
%factor_cal;

/* Thank you for reading 1104 lines of code */
```

MGMT 237H Homework 4

Group 5

Jie "Edward" Sheng (jie.sheng.2012@anderson.ucla.edu)

Anshul Maheshwari (anshul.maheshwari.2012@anderson.ucla.edu)

Chao Liang (chao.liang.2012@anderson.ucla.edu)

Li Zhang (li.zhang.2012@anderson.ucla.edu)

PART I. REPLICATE TABLE 1

Mapping COMPUSTAT Data

Old	Description	New
1	Cash and short Term Investment	CHE
4	Current Asset	ACT
5	Current Liabilities	LCT
6	Total Asset	AT
9	Long Term Debt	DLTT
14	Depreciation and Amortization	DP
25	Shares Outstanding	CSHO
34	Debt included in current liabilities	DLC
38	Minor Interests	MIB
60	Book Value of Common Equity	CEQ
71	Income Tax Payable	TXP
130	Preferred Stocks	PSTK
178	Income From Continuing Operations	OIADP
199	Fiscal Year End Closing Price	PRCC_F

COMPUSTAT Data Processing

- Download data from fiscal year 1963 to 2000
- Keep data only for following exchanges (COMPUSTAT code in brackets) NYSE (11) AMEX (12) and NASDAQ (14)
- Calculate NOA, Earnings, Accruals, Cash Flows, Book Value, Market Value and Book/Market ratio
- Rank the NOA in each year into 10 deciles
- Calculate the mean and median of each of the variables: NOA, Earnings, Accruals, Cash Flows, BV, MV and B/M
- Output the results to CSV files.

CRSP Data Processing:

- Keep records that belong to following exchanges NYSE (1) AMEX (3) NASDAQ (3)
- Merge CRSP with Compustat.
  - Move month of report date forward by 4 months
  - The financial data will be used from this month and onward, as assumed in page 23 of the paper
- Merge CRSP and COMPUSTAT by matching PERMNO and YEAR

CRSP Data Processing:

- Create a macro to move the data to previous year depending on the fiscal month and the report month when the company reports their data
- Again rank the Stocks for each month every year into 10 deciles using NOA

Calculate Beta

- We use three nested loops- 1) Decile 2) Year 3) Month
- As discussed in the paper we do a rolling regression of decile portfolio return on NYSE AMEX Equal Weighted Return
- The Rolling regression is for 60 months data
- We calculate beta of the each decile every month of all years
- The number of regressions required = 30(years) * 12 (months) * 10 (deciles) = 3600!
- The beta of each portfolio/decile is stored in its own table
- Take the mean and median of each decile to get the betas

TABLE 1

Mean (Median) Values of Selected Characteristics for Decile Portfolios Sorted by NOA

	Portfolio NOA Ranking									
	Lowest	2	3	4	5	6	7	8	9	Highest
Panel A: Accounting Variables										
NOA	0.251	0.490	0.589	0.654	0.705	0.751	0.798	0.854	0.943	1.485
	0.264	0.485	0.575	0.637	0.688	0.738	0.788	0.849	0.936	1.171
Earnings	0.019	0.077	0.100	0.107	0.110	0.113	0.118	0.123	0.136	0.144
	0.060	0.090	0.100	0.104	0.105	0.108	0.110	0.116	0.126	0.149
Accruals	-0.075	-0.055	-0.044	-0.039	-0.033	-0.026	-0.019	-0.004	0.016	0.079
	-0.050	-0.049	-0.044	-0.041	-0.037	-0.033	-0.026	-0.016	-0.001	0.033
Cashflows	0.094	0.132	0.144	0.146	0.143	0.140	0.137	0.126	0.119	0.064
	0.121	0.146	0.148	0.148	0.142	0.140	0.135	0.128	0.121	0.099
BV ($m)	189.747	419.748	624.380	663.079	593.382	545.562	458.733	417.945	399.358	326.829
	26.014	45.773	61.786	71.017	82.387	81.701	76.459	65.470	61.910	45.847
Panel B: Asset Pricing Variables										
MV ($m)	650.456	1146.725	1805.905	1858.996	1458.045	1406.722	943.530	919.260	972.525	925.095
	59.837	76.529	93.709	108.112	115.433	117.837	110.250	98.205	105.491	99.804
B/M	1.354	1.725	4.464	4.187	1.494	0.913	0.886	0.844	2.553	5.901
	0.462	0.607	0.658	0.681	0.706	0.709	0.707	0.676	0.604	0.486
Beta	1.250	1.161	1.093	1.061	1.004	1.000	0.973	1.000	1.051	1.191
	1.144	1.098	1.070	1.039	0.993	0.995	0.972	0.976	1.037	1.170

Observations based on results:

- Our results are quite close to the results published in the paper.
- Beta is high at both ends of the decile portfolios
- In the middle the Beta is either 1 or very close to 1
- This table also indicates that extreme (low and high) NOA firms have the smallest size
- measured either by book value of equity or market value of equity and the lowest book to market ratios.
- NOA varies from the lowest of 25% to 148% in the highest decile.
- This confirms the finding of the author(s) that high NOA stocks are over-valued

Observations based on results:

The slight variations that we see in data can be attributed to following:

- Quality of data set used for the original paper and our assignment
- Number of years that have passed since the paper was originally written hence affecting the quality of data available now
- Unstated assumptions by the author(s) while performing the analyses.

PART II. REPLICATE TABLE 4

Average future 3 yr return

- Use macros to calculate the cumulative return in future 1, 2 and 3 years and previous 1 year
- Sort the data in each year and month into 125 groups based on market cap, B/M ratio and previous year return.

Benchmark and adjusted return

- Calculate weighted average return for each of 125 groups using both value weighted method and equal weighted method
- Calculate adjusted return against benchmark
- Difference between average return in future 3 years of each particular stock and of the group the stock belongs to
- Calculate raw return and adjusted return of each NOA decile using equal weight and value weight
- Separate each NOA decile into a sub database for return calculation
- Calculate weighted average adjusted return in future 3 years for each NOA decile

Hedge L-H portfolio

- Merge the weighted return in the first and last NOA decile by matching year and month
- Calculate the difference of weighted return in next year, the adjusted return in future 3 years between the two NOA deciles for both equal weighted and value weighted portfolios.
- Add the result to the end of the table we got from the previous step

CAPM, 3 factor, 4 factor alpha calculation

- We calculate market, Fama-French, 4 factor time series using the code from last homework
- We import risk-free rate from St. Louis Fed
- We then regress the future 1-3 yr return time series on market excessive return, Fama-French factors, and 4 factors to get the alpha and t-stat

Observations based on results:

- From our result, the average monthly adjusted equally weighted return spread between lowest and highest NOA deciles is 0.56%; in year t+2 the effect is 0.51%; in t+3 it is 0.40%; the value weighted return spreads are around 0.3%-0.4%
- From the t value, we can see that the return spreads are very significant

Table 4

	Equal Weighted					Value Weighted			
Portfolio ranking	raw_ew t+1	adj_ew t+1	adj_ew t+2	adj_ew t+3		raw_vw t+1	adj_vw t+1	adj_vw t+2	adj_vw t+3
Lowest	0.013712	0.002533	0.002389	0.00196		0.011342	0.001129	0.000843	0.000835
	12.75	6.44	9.83	10.38		12.16	2.43	3.13	4.29
2	0.013358	0.001873	0.001416	0.00125		0.011448	0.000877	0.000891	0.000988
	13.37	7.16	7.37	8.54		15.49	3.82	5.15	7.14
3	0.01306	0.001408	0.001051	0.000708		0.011344	0.000994	0.000794	0.000855
	14.94	6.48	6.7	7.34		14.06	2.33	3.9	6.47
4	0.011588	0.000313	0.000478	0.000395		0.01046	0.00032	0.000429	0.000373
	14.05	2.09	4.37	4.61		17.62	1.76	2.28	2.57
5	0.011237	0.000227	0.000177	0.000166		0.010008	0.000237	0.000328	0.000396
	13.95	1.5	1.8	1.91		16.72	1.36	2.65	3.8
6	0.011317	-0.00011	-0.00022	-0.0002		0.010253	0.000244	-4.6E-05	-0.00031
	14.09	-0.74	-2.09	-2.28		16.19	1.45	-0.33	-2.53
7	0.010279	-0.00041	-0.00012	9.70E-06		0.008417	-0.00049	-0.00014	3.66E-05
	13.04	-3.36	-1.41	0.12		13.35	-3.12	-1.28	0.39
8	0.009189	-0.00124	-0.00111	-0.00099		0.008731	-0.00017	-0.00038	-0.00036
	11.38	-10.5	-9.47	-10.76		13.88	-0.9	-2.63	-2.95
9	0.008462	-0.0013	-0.00119	-0.00101		0.007599	-0.00106	-0.00086	-0.00083
	10.31	-7.21	-10.18	-10.72		13.87	-5.59	-5.64	-5.74
Highest	0.005798	-0.00308	-0.0027	-0.00205		0.005496	-0.00305	-0.00271	-0.00239
	5.63	-11.68	-13.58	-13.56		6.64	-10.26	-10.33	-10.77
Hedge(L-H)	0.007914	0.005616	0.005086	0.004014		0.005846	0.004175	0.003549	0.003225
	11.83	11.06	13.87	14.7		6	6.45	8.27	9.5

Table 4								
	Equal Weighted				Value Weighted			
Portfolio	*raw_ew*	*adj_ew*	*adj_ew*	*adj_ew*	*raw_vw*	*adj_vw*	*adj_vw*	*adj_vw*
ranking	*t+1*	*t+1*	*t+2*	*t+3*	*t+1*	*t+1*	*t+2*	*t+3*
CAPM α	0.0079	0.0056	0.0051	0.0040	0.0058	0.0042	0.0036	0.0033
	11.66	10.99	13.77	14.58	5.91	6.46	8.34	9.56
3 Factor α	0.0077	0.0055	0.0051	0.0040	0.0056	0.0041	0.0036	0.0032
	11.53	10.88	13.62	14.37	5.67	6.21	8.22	9.53
4 factor α	0.0075	0.0053	0.0048	0.0038	0.0054	0.0038	0.0034	0.0032
	10.61	9.89	12.26	13.10	5.18	5.51	7.44	8.82

Conclusions

- *Replication*
  - Table 1 and Table 4 successfully replicated
  - NOA trading strategy might be profitable

Thank You