Do Older Investors Make Better Investment Decisions?

George M. Korniotis and Alok Kumar

Review of Economics and Statistics, 93 (1), 244-265, 2011.

BROKERAGE DATA

We *use proprietary brokerage data* in this paper. The programs available for the replication of our results are listed below. The original brokerage data can be obtained from Professor Terrance Odean, University of California at Berkeley (odean@haas.berkeley.edu) by signing a non-disclosure agreement.

MATLAB PROGRAMS

If you need help in interpreting these programs, please contact Alok Kumar, University of Miami (akumar@miami.edu). Additional code that may be needed to run these programs can be obtained from this source as well.

```
%
% oldAgeAndPerfRegressionNEW2.m
% Regress perf measure on age and other vars.
% NEW2: just to have a working old version.
%
% AK, May 2009

clear all; warning off;

excludeCAInvestors = 0; excludeNYInvestors = 0;
excludeBottomSIR = 0;
excludeNonIRAInvestors = 0;
ConsiderInternetUsersOnly = 0;

% Sub samples
turnoverBasedGroup = 0;  % 0: None, 1: low turnover, 2: high turnover
minAge = 15; maxAge = 99;  % Age based sub sample
%minStartMonth = 1; maxStartMonth = 1;
minPortfSize = 000;
remZerosMF = 0;

useWealth = 0;
alphaCol = 3; minMonths = 12;
useCAPMAlpha = 0;
strtMonPerf = 1; lstMonPerf = 71;
%strtMonPerf = 13; lstMonPerf = 24;
%strtMonPerf = 25; lstMonPerf = 72;
%strtMonPerf = 37; lstMonPerf = 48;
%strtMonPerf = 49; lstMonPerf = 60;
%strtMonPerf = 48; lstMonPerf = 72;
oldAgeCutoff = 60; veryOldAgeCutoff = 70; YoungAgeCutoff = 30;
trimDistrib = 0; trimPct = 1;
useSigAlphas = 0;   % min t-value for alpha
modelNum = 9999.4;   % Main models: 9, 10, and 12.
% 1: LHS: Nstks, RHS: Age, Exper
% 2:             RHS: (1) + Other Control Vars
% 3: LHS: Turnover, RHS: (1)
% 4: LHS: Turnover, RHS: (2)
% 5: LHS: DY, RHS: (2)
% 6: LHS: MktTiming, RHS: (1)
% 69: LHS: MktTiming, RHS: (2)
% 7: LHS: SR, RHS: Age, Exper
% 8: LHS: Alpha, RHS: Age, Exper
% 9: LHS: SR, RHS: (7) + Control Vars
```

```
% 10: LHS: Alpha, RHS: (7) + Control Vars
% 11: LHS: MonRet, RHS: Age, Exper
% 12: LHS: MonRet, RHS: (7) + Control Vars
% 101: LHS: SR, RHS: Age but no exp (to match with Odean results)
% 102: LHS: Alpha, RHS: Age but no exp (to match with Odean results)
% 103: LHS: SR, RHS: Age + Turnover but no exp (to match with Odean results)
% 104: LHS: Alpha, RHS: Age + Turnover but no exp (to match with Odean results)
% 201: LHS: Self-Benchmark, RHS: Age + Exper
% 901: LHS: Char-Based Perf, RHS: Age, Exper
% 909: LHS: Char-Based Perf, RHS: (7) + Control Vars
% 1001: LHS: Char-Based Perf CHANGE, RHS: Age, Exper
% 1009: LHS: Char-Based Perf CHANGE, RHS: (7) + Control Vars
% 2001: LHS: Local bias, RHS: Age, Exper
% 2002: LHS: Local bias, RHS: Table IV vars
% 2009: LHS: Local bias, RHS: (7) + Control Vars
if modelNum == 1 | modelNum == 2
    regNum = 3;
elseif modelNum == 7 | modelNum == 9 | modelNum == 101 | modelNum == 103
    regNum = 1;
elseif modelNum == 8 | modelNum == 10 | modelNum == 102 | modelNum == 104
    regNum = 2;
elseif modelNum == 11 | modelNum == 12
    regNum = 5;
elseif modelNum == 6 | modelNum == 69
    regNum = 6;
elseif modelNum == 201 | modelNum == 202 | modelNum == 210
    regNum = 8;
elseif modelNum == 901 | modelNum == 909
    regNum = 9;
elseif modelNum == 1001 | modelNum == 1009
    regNum = 10;
elseif modelNum >= 2001 & modelNum <= 2009
    regNum = 11;
elseif modelNum == 99
    regNum = 99;
else
    regNum = 9;
end;
skillMeasure = regNum
% 1: SR, 2: Alpha, 3: DivMeasure, 4: PTBSD, 5: MeanMonRet, 6: MktTiming 9: Char adjusted returns
% 9.1: 9 with top 10% top performers removed
% 12: MutualFundPerf, 13: PerfDifferential
skillMeasure = 9
% Change in char adj ret
skillMeasure = 10


load hhChars hhc;
% hhc(9): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive
hhc(hhc==-999)=NaN; AGE = hhc(:,[1 8]); AGESQ = [AGE(:,1) AGE(:,2).^2];
% AGE FILTER
[r,c] = find(AGE(:,2)>=minAge & AGE(:,2)<=maxAge); AGE = AGE(r,:); hhc = hhc(r,:);

% Age-range dummies
Nage = max(size(AGE)); AGERANGE = zeros(Nage,5); AGERANGE(:,1) = AGE(:,1);
a = [0 40]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,2) = 1;
a = [40 50]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,3) = 1;
a = [50 65]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,4) = 1;
a = [65 95]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,5) = 1;

% Finer age range categories
%Nage = max(size(AGE)); AGERANGE = zeros(Nage,10); AGERANGE(:,1) = AGE(:,1);
%a = [0 30]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,2) = 1;
%a = [30 35]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,3) = 1;
%a = [35 40]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,4) = 1;
%a = [40 45]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,5) = 1;
%a = [45 50]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,6) = 1;
%a = [50 60]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,7) = 1;
%a = [60 65]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,8) = 1;
```

```
%a = [65 70]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,9) = 1;
%a = [70 100]; [r,c] = find(AGE(:,2)>=a(1) & AGE(:,2)<a(2)); AGERANGE(r,10) = 1;

% Age and Age-Squared using Legendre polynomials
Age = AGE(:,2); AgeMax = nanmax(Age); AgeMin = nanmin(Age); Age = (Age - AgeMin)/(AgeMax -
AgeMin);
L = 2*Age - 1; L2 = 6*Age.^2 - 6*Age + 1;
LEGPOLY = [AGE(:,1) L L2]; %fndCorrCoef([L L2]), fndPause;
% Correl = -0.23.

% Age and Age-Squared after demeaning
Age = AGE(:,2)-nanmean(AGE(:,2));
AGEANDSQ = [AGE(:,1) Age Age.^2]; %fndCorrCoef(AGEANDSQ(:,2:3)), fndPause;
AGECUBE = [AGE(:,1) Age.^3];
% Correl = 0.424.

PSIZE = hhc(:,[1 2]);

% Very old dummy
Nage = max(size(AGE)); VOLDDUMMY = [AGE(:,1) zeros(Nage,1)];
[r,c] = find(AGE(:,2)>=veryOldAgeCutoff); VOLDDUMMY(r,2) = 1;

% Very young dummy
Nage = max(size(AGE)); YOUNGDUMMY = [AGE(:,1) zeros(Nage,1)];
[r,c] = find(AGE(:,2)>=YoungAgeCutoff); YOUNGDUMMY(r,2) = 1;

% TAX Dummy
% Taxable vs TDA accounts
load AccTypeAndSegment.txt;   % contains: HHNum, AccNum, AccType, ClientSegment
atype = AccTypeAndSegment; clear AccTypeAndSegment;   % 158031 x 4
% (1) Cash: 42565, (2) Margin: 49737, (3) IRA: 64416, (4) Keogh: 1299, (5) Unknown: 14 (Code is -
1)

taxHH = unique(atype(:,1)); Nhh = max(size(taxHH));

% Taxable HH dummy: HH with ONLY taxable accounts
[r,c] = find(atype(:,3)==1|atype(:,3)==2);  selTaxHH = unique(atype(r,1));
[r,c] = find(atype(:,3)==3|atype(:,3)==4); selNonTaxHH = unique(atype(r,1));
selTaxHH = setdiff(selTaxHH,selNonTaxHH); Ntax = max(size(selTaxHH))

% Non-Taxable HH dummy: HH with ONLY tax-exempt accounts
[r,c] = find(atype(:,3)==3|atype(:,3)==4); selNonTaxHH = unique(atype(r,1));
[r,c] = find(atype(:,3)==1|atype(:,3)==2);  selTaxHH = unique(atype(r,1));
selNoTaxHH = setdiff(selNonTaxHH,selTaxHH); Nnotax = max(size(selNoTaxHH))

Nhh = max(size(hhc)); TDADummy = zeros(Nhh,2); TDADummy(:,1) = hhc(:,1);
idx = ismember(hhc(:,1),selNoTaxHH); TDADummy(idx,2) = 1;

% Age and race interaction
% New Census Info
load HHNumZipLevelCensusInfo HHCensus;
% (1)HHNum (2)Zipcode (3)ForeignBorn (4)EduBachHigh (5)Income (6)White (7)Black (8)Hispanic;
CENSUSINFO = HHCensus(:,[1 3 4 5 6 7 8]);
EDU = HHCensus(:,[1 4]); FORBORN = HHCensus(:,[1 3]);
blackRat = HHCensus(:,7)./HHCensus(:,6); hispRat = HHCensus(:,8)./HHCensus(:,6);
RACE = [HHCensus(:,1) blackRat hispRat];
% Identify zipcodes with large concentration of AfAm and Hisp
Nhh = max(size(RACE)); RACEDUMMIES = [RACE(:,1) zeros(Nhh,2)];
afamCutoff = prctile(RACE(:,2),75); [r,c] = find(RACE(:,2)>=afamCutoff); RACEDUMMIES(r,2) = 1;
hispCutoff = prctile(RACE(:,3),75); [r,c] = find(RACE(:,3)>=hispCutoff); RACEDUMMIES(r,3) = 1;

RACEAGE = fndCombineMatrices(RACEDUMMIES(:,[1 3]),AGE);
RACEAGE = [RACEAGE(:,1) RACEAGE(:,2).*RACEAGE(:,3)];

RACEAGE2 = fndCombineMatrices(RACEDUMMIES(:,[1 2]),AGE);
RACEAGE2 = [RACEAGE2(:,1) RACEAGE2(:,2).*RACEAGE2(:,3)];

Nage = max(size(AGE)); OLDDUMMY = zeros(Nage,2); OLDDUMMY(:,1) = AGE(:,1);
[r,c] = find(AGE(:,2)>=oldAgeCutoff); OLDDUMMY(r,2) = 1;

RETDUMMY = zeros(Nage,2); RETDUMMY(:,1) = AGE(:,1);
```

```
[r,c] = find(hhc(:,7)==8); RETDUMMY(r,2) = 1;

% Job code dummies
jcd1 = hhc(:,7); jcd2 = hhc(:,7); jcd3 = hhc(:,7);
jcd1(jcd1==1|jcd1==2)=-1; jcd1(jcd1>=0)=0; jcd1(jcd1==-1)=1;   % Professional
%jcd2(jcd2==3|jcd2==4|jcd2==5|jcd2==6|jcd2==7)=-1; jcd2(jcd2>=0)=0; jcd2(jcd2==-1)=1;   % Non-
Professional
Nhh = max(size(hhc)); PROFDUMMY = zeros(Nhh,2); PROFDUMMY(:,1) = hhc(:,1); PROFDUMMY(:,2) = jcd1;
fndPause;


icd = hhc(:,6);
icd(icd == 1) = 7.5;
icd(icd == 2) = 17.5;
icd(icd == 3) = 25;
icd(icd == 4) = 35;
icd(icd == 5) = 45;
icd(icd == 6) = 62.5;
icd(icd == 7) = 87.5;
icd(icd == 8) = 112.5;
icd(icd == 9) = 250;    % ref: Ivkovich:Weisbenner
INC = [hhc(:,1) icd];
SIR = [hhc(:,1) hhc(:,2)./(1000*hhc(:,6))];
% Low income
Ninc = max(size(INC)); LOWINC = zeros(Ninc,2); LOWINC(:,1) = INC(:,1);
[r,c] = find(INC(:,2)<=40); LOWINC(r,2) = 1;


load hhChars htc hhgrps10 hhc5min; hhc = hhc5min; hhgrps = hhgrps10; clear hhc5min hhgrps10;
hhc = sortrows(hhc,1); htc = sortrows(htc,1);
hhc = [hhc htc(:,2:6)];
turn = nanmean(hhc(:,[10 11])')';  TURN = [hhc(:,1) turn];
MFUNDS = [hhc(:,1) hhc(:,12)]; MFUNDS(isnan(MFUNDS)) = 0;

% Taxable vs TDA accounts
load AccTypeAndSegment.txt;   % contains: HHNum, AccNum, AccType, ClientSegment
atype = AccTypeAndSegment; clear AccTypeAndSegment;   % 158031 x 4
% (1) Cash: 42565, (2) Margin: 49737, (3) IRA: 64416, (4) Keogh: 1299, (5) Unknown: 14 (Code is -
1)

taxHH = unique(atype(:,1)); Nhh = max(size(taxHH));
TDADUMMY = zeros(Nhh,3); TDADUMMY(:,1) = taxHH;

% Taxable HH dummy: HH with ONLY taxable accounts
[r,c] = find(atype(:,3)==1|atype(:,3)==2);  selTaxHH = unique(atype(r,1));
[r,c] = find(atype(:,3)==3|atype(:,3)==4); selNonTaxHH = unique(atype(r,1));
selTaxHH = setdiff(selTaxHH,selNonTaxHH);
idx = ismember(taxHH,selTaxHH); TDADUMMY(idx,2)=1;

% Non-Taxable HH dummy: HH with ONLY tax-exempt accounts
[r,c] = find(atype(:,3)==3|atype(:,3)==4); selNonTaxHH = unique(atype(r,1));
[r,c] = find(atype(:,3)==1|atype(:,3)==2);  selTaxHH = unique(atype(r,1));
selNoTaxHH = setdiff(selNonTaxHH,selTaxHH);
idx = ismember(taxHH,selNoTaxHH); TDADUMMY(idx,3)=1;

% Has at least on TDA account
ONETDA = zeros(Nhh,2); ONETDA(:,1) = taxHH;
idx = ismember(taxHH,selTaxHH); ONETDA(idx,2) = 1;

% Has at least on TDA account
MARGINACC = zeros(Nhh,2); MARGINACC(:,1) = taxHH;
[r,c] = find(atype(:,3)==2); MarginHH = unique(atype(r,1));
idx = ismember(taxHH,MarginHH);  MARGINACC(idx,2) = 1;

% PORTFOLIO DIVIDEND YIELD
load invMonPortfDivYldBIG DY allHH;
dy = DY(:,1:72); DY = [allHH 4*nanmean(100*dy')'];   % Annual dividend yield

load invPortfPerfAllHH ipc;   % returns ipc
% ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
```

```
%              (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep (ex
post Sharpe ratio);
%              (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta, (23)
Tval: FFAlpha,
%              (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28) Rbarsq:
CAPM Reg,
%              (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
SR = ipc(:,[1 16]); %RAWRET = ipc(:,[1 8]);

if skillMeasure == 1
    % PORTFOLIO PERFORMANCE
    % Get performance data
    load invPortfPerfAllHH ipc;   % returns ipc
    % ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
    %               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep
(ex post Sharpe ratio);
    %               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta,
(23) Tval: FFAlpha,
    %               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28)
Rbarsq: CAPM Reg,
    %               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
    PERF = ipc(:,[1 16]);
elseif skillMeasure == 2
    % DOMESTIC FF Estimates
    % Fama-French estimates (portfolio risk measures)
    if useCAPMAlpha == 1
        load invFamaFrenchEachHH capmAlph; alph = capmAlph;
    else
        load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
    end;
    %load invFamaFrenchEachHHNew alphWithLiq; alph = alphWithLiq; clear alphWithLiq;
    %load invFamaFrenchEachHHUsingTotRet2 alph8 alph4; alph = alph4;
    %load invFamaFrenchEachHHUsingMFRet alph8 alph4; alph = alph8;
    %[r,c] = find(alph(:,3)<-10); alph(r,3) = NaN; [r,c] = find(alph(:,3)>10); alph(r,3) = NaN;

    [r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
    PERF = alph(r,[1 alphaCol]);   % HHNum, AlphaDom
    if useSigAlphas > 0
        [r,c] = find(abs(alph(:,8))>=useSigAlphas); Nr = max(size(r));
        PERF = alph(r,[1 alphaCol]);   % HHNum, AlphaDom
    end;
elseif skillMeasure == 3
    if modelNum == 1 | modelNum == 2
        % Diversification
        load hhChars hdivcAllP;   % 41039 x 6
        PERF = hdivcAllP(:,[1 3]);
        % hdivcAllP contains: (1)HHNum (2)Nobs (3)Nstks (4)Sqw (5)NormVar
        % (6)AvgCorrel
    else
        % Diversification
        load hhChars hdivcAllP;   % 41039 x 6
        PERF = hdivcAllP(:,[1 6]);
        % hdivcAllP contains: (1)HHNum (2)Nobs (3)Nstks (4)Sqw (5)NormVar
        % (6)AvgCorrel
    end;
elseif skillMeasure == 4
    % Informed in domestic setting?
    minTrades = 3; locCutoff = 250;
    winSize = 21;   % (1)5 (2)10 (3)21 (4)42 (5)63 (6)84 (7)105 (8)126 (9)252
    infoFile = ['C:\1cornell\2research\1behfin\51local\2data\OverConfiEachHH_Win'
num2str(winSize) '_Dist' num2str(locCutoff) '.mat'];
    load(infoFile,'OC','NTR','allHH');
    %OC = 100*OC(:,:,ocCol)); % OC = NaN*zeros(Nhh,6,9); NTR = NaN*zeros(Nhh,6);
    % OC: ExPostBuyLoc, ExPostSellLoc, ExPostBuyNonLoc, ExPostSellNonLoc, ExPostBuyAll,
ExPostSellAll
    for i = 1:6
        [r,c] = find(NTR(:,i)<minTrades); OC(r,i)=NaN;
    end;

    % Get info measure
    info = [allHH OC(:,1)-OC(:,2) OC(:,3)-OC(:,4) OC(:,5)-OC(:,6)];
```

```
    % HHNum, InfoMeasureLoc, InfoMeasureNonLoc, InfoMeasureAll
    PERF = info(:,[1 3]);
elseif skillMeasure == 5
    % PORTFOLIO PERFORMANCE
    % Get performance data
    load invPortfPerfAllHH ipc;   % returns ipc
    % ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
    %               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep
(ex post Sharpe ratio);
    %               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta,
(23) Tval: FFAlpha,
    %               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28)
Rbarsq: CAPM Reg,
    %               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
    PERF = ipc(:,[1 8]);

    % Actual Returns
    load invMonPortfRetALLBIG R allHH;
    %load invMonPortfRetBIG R allHH;   % 55277 x 72
    PERF = [allHH nanmean(R(:,strtMonPerf:lstMonPerf)')'];
    %PERF = [allHH nanstd(R(:,strtMonPerf:lstMonPerf)')'];
elseif skillMeasure == 6   % Market timing
    load invMktTimingEachHH mtm allHH;
    % (1)rsr (2)esr (3)jenalph (4)tJenAlph (5)gh(1) (6)Tgh1 (7)gh2 (8)Tgh2 (9)beta2HM
(10)tBeta2HM
    % (11)beta2TM (12)tBeta2TM (13)beta (14)ffalph (15)tFFAlph
    %[r,c] = find(abs(mtm(:,12))>=1.65); mtm = mtm(r,:); allHH = allHH(r,1); PERF = [allHH
mtm(:,11)];
    %[r,c] = find(abs(mtm(:,10))>=1.65); mtm = mtm(r,:); allHH = allHH(r,1); PERF = [allHH
mtm(:,9)];
    PERF = [allHH mtm(:,11)];
elseif skillMeasure == 7   % Tax skill
    load invDecTaxLossSelling.mat TaxSkill; % HHNum, Nact, Npap
    TSKILL = [TaxSkill(:,1) TaxSkill(:,2)./TaxSkill(:,3)];

    Nhh = max(size(hhc)); PERF = [hhc(:,1) zeros(Nhh,1)];
    idx = ismember(hhc(:,1),TaxSkill(:,1));
    PERF(idx,2) = 1;
    %idx2 = ismember(TSKILL(:,1),hhc(:,1)); PERF(idx,2) = TSKILL(idx2,2);
elseif skillMeasure == 8
    % SELF BENCHMARK
    % Actual Returns
    load invMonPortfRetBIG R allHH;   % 55277 x 72
    Ract = R;
    load invMonPortfRetSelfBIG R allHH;   % 55277 x 72
    Rself = R;
    R = Ract - Rself; clear Ract Rself;
    PERF = [allHH nanmean(R(:,strtMonPerf:lstMonPerf)')'];
elseif skillMeasure == 9   % Char-Based Excess Ret
    load invMonCharAdjPortfRetBIG R allHH;   % Nhh x 72
    load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
    [r,c] = find(alph(:,2)>=minMonths); selHH = alph(r,1);
    idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(r,1);
    PERF = [allHH nanmean(R(:,strtMonPerf:lstMonPerf)')']; clear R;
elseif skillMeasure == 9.1   % Char-Based Excess Ret: REMOVE TOP k%
    load invMonCharAdjPortfRetBIG R allHH;   % Nhh x 72
    load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
    [r,c] = find(alph(:,2)>=minMonths); selHH = alph(r,1);
    idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(r,1);
    PERF = [allHH nanmean(R(:,strtMonPerf:lstMonPerf)')']; clear R;
    PerfCutoff = prctile(PERF(:,2),95); [r,c] = find(PERF(:,2)>=PerfCutoff); PERF(r,2) = NaN;
    PerfCutoff = prctile(PERF(:,2),5); [r,c] = find(PERF(:,2)<=PerfCutoff); PERF(r,2) = NaN;
elseif skillMeasure == 10   % CHANGE in Char-Based Excess Ret
    load invMonCharAdjPortfRetBIG R allHH;   % Nhh x 72
    %load NetHouseholdReturns2 R1 R2 R3 R4 AllHH;   R = R2'; allHH = AllHH;  % 72 x Nhh
    %load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
    %[r,c] = find(alph(:,2)>=minMonths); selHH = alph(r,1);
    %idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(r,1);
    PERF = [allHH nanmean(R(:,37:72)')'-nanmean(R(:,1:36)')']; clear R;
    %PERF = [allHH nanmean(R(:,25:48)')'-nanmean(R(:,1:24)')']; clear R;
    %PERF = [allHH nanmean(R(:,49:72)')'-nanmean(R(:,1:24)')']; clear R;
```

```
elseif skillMeasure == 11
    % LOCAL BIAS
    load LocalBias LB; % HHNum, LBOld, LBNew
    PERF = LB(:,[1 2]);
    load AdjustedDispEffect ADE; PERF = ADE;
    load VolatilityRedn VREDN; PERF = VREDN;

    % DOMESTIC FF Estimates
    minMonths = 12; minMonthsFor = 12;
    % Fama-French estimates (portfolio risk measures)
    load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
    [r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
    fprintf(1, '%d investors have reliable FF estimates.\n', Nr);
    DOMALPHA = alph(r,[1 3]);   % HHNum, AlphaDom

    inFile = ['C:\1cornell\2research\1behfin\7intl\2data\invForeignPortfRetNewBIG.mat'];
    load(inFile,'totAlphas','totHH','SRtot');   % [b' tval' rbar2 n]
    [r,c] = find(totAlphas(:,12)>=minMonthsFor);
    alphTot = [totHH(r,1) totAlphas(r,1)];   % HHNum, AlphaTot


    % COMBINE Performance Measures
    Z1 = fndCombineMatrices(DOMALPHA,alphTot);   % HHNum, AlphaDom, AlphaTot
    N = max(size(Z1)); fprintf(1, '%d investors with reliable domestic and total FF
estimates.\n', N);
    PERF = [Z1(:,1) Z1(:,2)];


    % DOMESTIC Sharpe ratios
    % Get performance data
    load invPortfPerfAllHH ipc;   % returns ipc
    % ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
    %               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep
(ex post Sharpe ratio);
    %               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta,
(23) Tval: FFAlpha,
    %               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28)
Rbarsq: CAPM Reg,
    %               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
    DOMSR = ipc(:,[1 16]);

    % TOTAL (Domestic + Foreign) FF Estimates + Sharpe Ratios
    inFile = ['c:\1cornell\2research\1behfin\7intl\2data\invForeignPortfRetNewBIG.mat'];
    load(inFile,'totAlphas','totHH','SRtot');   % [b' tval' rbar2 n]
    Z2 = fndCombineMatrices(DOMSR,[totHH SRtot]);   % HHNum, SRDom, SRTot
    N = max(size(Z2)); fprintf(1, '%d investors with reliable domestic and total FF
estimates.\n', N);
    PERF = [Z2(:,1) (Z2(:,3) - Z2(:,2))];


    %load VolatilityRedn VREDN; PERF = VREDN;

    minMonths = 18;
    load LocNonlocAlphasEachHH locRes nonlocRes;
    [r,c] = find(locRes(:,2)>=minMonths);
    PERF = [locRes(r,1) locRes(r,3) nonlocRes(r,3)];   % HHNum, LocAlpha, NonLocAlpha
    PERF = PERF(:,[1 3]);
elseif skillMeasure == 12   % MF Ret
    minMonths = 1; perfNum = 3;   % 1: MeanRet, 2: Volatility, 3: Mean/Volatility
    load invMonMutualFundRetHH.txt; R = invMonMutualFundRetHH; clear invMonMutualFundRetHH;
    % HHNum, MeanRet, StdDevRet, NumCases
    [r,c] = find(R(:,4)<minMonths); R(r,:) = [];
    if perfNum == 1
        MFHH = [R(:,1) 100*R(:,2)];
    elseif perfNum == 2
        MFHH = [R(:,1) R(:,3)];
    else
        MFHH = [R(:,1) 100*R(:,2)./R(:,3)];
    end;
    PERF = MFHH;
```

```
    % Get HH chars
    % Portfolio chars + Trading chars: combine them
    load hhChars HHC;
    hhc(hhc==-999)=NaN;
    load MeanMutualFundsPosHH.txt;   % HHNum, MeanPosition
    MFHOLD = MeanMutualFundsPosHH; clear MeanMutualFundsPosHH;

    Z = fndMergeMatrices(hhc(:,[1 2]),MFHOLD); Z(isnan(Z))=0;
    MFWEIGHT = [Z(:,1) 100*Z(:,3)./(Z(:,2)+Z(:,3))];

elseif skillMeasure == 13   % Perf Diff
    load c:\1cornell\2research\1behfin\27oldinv\2data\InvPerfDiffAcrossPeriods.mat PERFDIFF; PERF
= PERFDIFF;
end;
%PERF = hdivcAllP(:,[1 6]);

load invFamaFrenchEachHH alph;   % Nhh x 12: % NNHum, Ncases, b0,... b4, t0, ..., t4
minMonths = 24;
[r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
fprintf(1, '%d investors have reliable FF estimates.\n', Nr);
FACTEXPO = alph(r,[1 4:7]);   % HHNum, Factor exposures

% INVESTMENT EXPER
load hhChars EXPER;   % HHNum, InvExperience

% New Census Info
load HHNumZipLevelCensusInfo HHCensus;
% (1)HHNum (2)Zipcode (3)ForeignBorn (4)EduBachHigh (5)Income (6)White (7)Black (8)Hispanic;
CENSUSINFO = HHCensus(:,[1 3 4 5 6 7 8]);
EDU = HHCensus(:,[1 5]);
% Low edu
Nedu = max(size(EDU)); LOWEDU = zeros(Nedu,2); LOWEDU(:,1) = EDU(:,1);
eduLowCutoff = prctile(EDU(:,2),25); [r,c] = find(EDU(:,2)<=eduLowCutoff); LOWEDU(r,2) = 1;

% NASDAQ and Industry concentration
load NasdaqIndusConc ndqIndusConc allHH;
% NasdaqDummy + FF-Industry Conc
INDUSCONC = [allHH ndqIndusConc]; clear ndqIndusConc;

% GENDER
load infobaseWithGender.txt; gender = infobaseWithGender(:,[1 12]);
[r,c] = find(gender(:,1)==2869090); gender(r(1),:)=[];
[r,c] = find(gender(:,1)==3916722); gender(r(1),:)=[];
load hhChars ahmap;
[r,c] = find(ahmap(:,1)==2869090); ahmap(r(1),:)=[];
[r,c] = find(ahmap(:,1)==3916722); ahmap(r(1),:)=[];
[r,c] = find(ahmap(:,1)==3844354); ahmap(r(1),:)=[];
[r,c] = find(ahmap(:,1)==8827950); ahmap(r(1),:)=[];
gender = fndCombineMatrices(gender,ahmap); gender = gender(:,[3 2]);
[r,c] = find(gender(:,1)==8862950); gender(r(1),:)=[];
% gender contains: HHNum, Gender (0: Female, 1: Male, -1: Unknown)
gender(gender==-1)=NaN; GENDER = gender;

% Age and gender
GENDERAGE = fndCombineMatrices(GENDER,AGE);
GENDERAGE = [GENDERAGE(:,1) GENDERAGE(:,2).*GENDERAGE(:,3)];

load TaxRateIncomeNetWorth2.txt; x = TaxRateIncomeNetWorth2; clear TaxRateIncomeNetWorth2;
% x contains: HHNum, TaxRate, Income, NetWorth
WEALTH = x(:,[1 4]);
[r,c] = find(WEALTH(:,2)>0); WEALTH = WEALTH(r,[1 2]);

% Idiosyncratic risk
load invIdioSyncRiskEachHH.mat IR;
% IR contains: HHNum, Nmonths, IRusingCAPM, IRusingFF, TotalRisk
IR = IR(:,[1 4]);

% Interaction terms
% Old and less educated
Nage = max(size(AGE)); OLDEDU = zeros(Nage,2); OLDEDU(:,1) = AGE(:,1);
[r1,c] = find(AGE(:,2)>=oldAgeCutoff);
```

```
eduLowCutoff = prctile(EDU(:,2),25); [r2,c] = find(EDU(:,2)>=eduLowCutoff);
r = intersect(r1,r2); OLDEDU(r,2) = 1;

% Old and poor
Nage = max(size(AGE)); OLDINC = zeros(Nage,2); OLDINC(:,1) = AGE(:,1);
[r1,c] = find(AGE(:,2)>=oldAgeCutoff);
wealthLowCutoff = prctile(WEALTH(:,2),25); [r2,c] = find(WEALTH(:,2)>=wealthLowCutoff);
r = intersect(r1,r2); OLDINC(r,2) = 1;

% Old and TDADummy
Nage = max(size(AGE)); OLDTDA = zeros(Nage,2); OLDTDA(:,1) = AGE(:,1);
[r1,c] = find(AGE(:,2)>=oldAgeCutoff);
[r2,c] = find(TDADummy(:,2)==1);
r = intersect(r1,r2); OLDTDA(r,2) = 1;


% DIVERSIFICATION
load hhChars hdivcAllP;   % 41039 x 6
DIV = [hdivcAllP(:,1) hdivcAllP(:,3)];
% hdivcAllP contains: HHNum, Nobs, Nstks, Sqw, NormVar, AvgCorrel

% LOCAL BIAS
load LocalBias LB; % HHNum, LBOld, LBNew
LB = LB(:,[1 2]);

% DE
%load DispEffectEachHHNew Z; % z = [allHH(i,1) Nxx Nyy (4)propAct (5)propExp];
%DE = [Z(:,1) 100*(Z(:,5)-Z(:,4))./Z(:,5)]; DE(isinf(DE)) = NaN;
load AdjustedDispEffect ADE;
load PeerAdjDispEffect2 PADE; deHH = PADE(:,1); DE = [deHH PADE(:,2)];   % HHNum, AdjDE

% OVERCONFIDENCE
% Dummy = 1 if in the top turnover quintile and bottom performance quintile
% Get performance data
load invPortfPerfAllHH ipc;   % returns ipc
% ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
%               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep (ex
post Sharpe ratio);
%               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta, (23)
Tval: FFAlpha,
%               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28) Rbarsq:
CAPM Reg,
%               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
% Demographics
% Cross-sectional regression
% Portfolio chars + Trading chars: combine them
load hhChars htc hhgrps10 hhc5min; hhc = hhc5min; hhgrps = hhgrps10; clear hhc5min hhgrps10;
hhc = sortrows(hhc,1); htc = sortrows(htc,1);
hhc = [hhc htc(:,2:6)];
% htc contains: (1)HHNum (2)Nstks (3)NbuyTrades (4)NsellTrades (5)NdaysActive (6)dysBetTrades
% hhgrps(7): HHNum, Ntr(buys), Ntr(sells), Groups for BUYS (2: 21d, 63d), Groups for SELLS (2:
21d, 63d)
% hhc(16): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive, (10)PurchTurn, (11)SalesTurn, (12) MFOwnerProp
%         (13)Nstks (14)NbuyTrades (15)NsellTrades (16)NdaysActive
%         (17)dysBetTrades (18) PortfSize/Income ratio
hhc(hhc==-99)=NaN; hhc(hhc==-999)=NaN;
turn = nanmean(hhc(:,[10 11])')';  TURN = [hhc(:,1) turn];
Nhh = max(size(ipc)); OCProxy = zeros(Nhh,2); OCProxy(:,1) = ipc(:,1);
[perfgrps,perfbkpts] = fndAssignToDeciles(ipc(:,[1 16]),33); lowperfhh = perfgrps{1}; highperfhh
= perfgrps{3};
[turngrps,perfbkpts] = fndAssignToDeciles(TURN,33); lowturnhh = turngrps{1}; highturnhh =
turngrps{3};
idx = ismember(lowperfhh,highturnhh); selhh = lowperfhh(idx,1); Nochh = max(size(selhh));
idx = ismember(OCProxy(:,1),selhh); OCProxy(idx,2) = 1;   % Low Perf + High Turn
OVERCONFI = OCProxy;   % HHNum, OC

k = 126;
inFile = ['c:\1cornell\2research\1behfin\64amtosh\2data\IPOTrades_k' num2str(k) '.mat'];
load(inFile,'tphh');   % HHNum, Nbuytrades, Nselltrades, BuyVol, SellVol
```

```
% DEMOGRAPHIC DATA
% ========================================================================
% Get HH chars
% Portfolio chars + Trading chars: combine them
load hhChars hhc;
% hhc(16): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive
hhc(hhc==-999)=NaN;
% Convert income codes into mid-points of income range
icd = hhc(:,6);
icd(icd == 1) = 7.5;
icd(icd == 2) = 17.5;
icd(icd == 3) = 25;
icd(icd == 4) = 35;
icd(icd == 5) = 45;
icd(icd == 6) = 62.5;
icd(icd == 7) = 87.5;
icd(icd == 8) = 112.5;
icd(icd == 9) = 250;    % ref: Ivkovich:Weisbenner
hhc(:,6) = icd;
DEMOG = hhc(:,[1 8 6 2]);

% Participation dummy
Ndemoghh = max(size(DEMOG));
PART = zeros(Ndemoghh,2); PART(:,1) = DEMOG(:,1);
idx = ismember(PART(:,1),tphh(:,1)); PART(idx,2) = 1;
idx = ismember(PART(:,1),tphh(:,1)); idx2 = ismember(tphh(:,1),PART(:,1)); PART(idx,2) =
tphh(idx2,4)/1000;


% LOCATION
% Metro-Non-Metro status
load MetroNonMetroStatusEachHH GP;   % HHNum, Metro-Non-Metro Status
% MetroNonMetroStatusEachHH2: with 20 metros
load FinCenterStatusEachHH GP;
%load NewYorkStatusEachHH GP;

Nmetrohh = max(size(GP));
METRO = zeros(Nmetrohh,2); METRO(:,1) = GP(:,1);
met = GP(:,2); met(met>1)=0;  % Metro dummy
remote = GP(:,2); remote(remote==1)=0; remote(remote==2)=0; remote(remote==3)=1;
METRO(:,2) = met; %METRO(:,3) = remote;

% Define various age related dummies
% Old and poor
Nage = max(size(AGE)); AGEDUMMIES = [AGE(:,1) zeros(Nage,7)];
%RACEAGE = fndCombineMatrices(RACEDUMMIES(:,[1 3]),AGE);
LowIncDummy = [zeros(Nage,1)]; [r,c] = find(INC(:,2)<=20); LowIncDummy(r,1) = 1;
dummy = [OLDDUMMY(:,2).*LowIncDummy]; AGEDUMMIES(:,2) = dummy;
dummy = [AGE(:,2).*LowIncDummy]; AGEDUMMIES(:,2) = dummy;

% Old and uneducated
[eduhh,edubkpts] = fndAssignToDeciles(EDU,33);
idx = ismember(AGEDUMMIES(:,1),eduhh{1}); AGEDUMMIES(idx,3) = 1;
lessEduDummy = AGEDUMMIES(:,3);
dummy = [AGE(:,2).*lessEduDummy]; AGEDUMMIES(:,3) = dummy;

% Old and high turnover
[turnhh,turnbkpts] = fndAssignToDeciles(TURN,20);
idx = ismember(AGEDUMMIES(:,1),turnhh{5}); AGEDUMMIES(idx,4) = 1;
highTurnDummy = AGEDUMMIES(:,4);
dummy = [AGE(:,2).*highTurnDummy]; AGEDUMMIES(:,4) = dummy;

% Old and less experienced
[experhh,experbkpts] = fndAssignToDeciles(EXPER,20);
idx = ismember(AGEDUMMIES(:,1),experhh{5}); AGEDUMMIES(idx,5) = 1;
lessExpDummy = AGEDUMMIES(:,5);
dummy = [AGE(:,2).*lessExpDummy]; AGEDUMMIES(:,5) = dummy;
```

```
% Old and Male
[r,c] = find(GENDER(:,2)==1); maleHH = GENDER(r,1);
idx = ismember(AGEDUMMIES(:,1),maleHH); AGEDUMMIES(idx,6) = 1;
maleDummy = AGEDUMMIES(:,6);
dummy = [AGE(:,2).*maleDummy]; AGEDUMMIES(:,6) = dummy;

% Old and Metro
[r,c] = find(METRO(:,2)==1); metroHH = METRO(r,1);
idx = ismember(AGEDUMMIES(:,1),metroHH); AGEDUMMIES(idx,7) = 1;
metroDummy = AGEDUMMIES(:,7);
dummy = [AGE(:,2).*metroDummy]; AGEDUMMIES(:,7) = dummy;

% Old and Professional
[r,c] = find(PROFDUMMY(:,2)==1); profHH = METRO(r,1);
idx = ismember(AGEDUMMIES(:,1),profHH); AGEDUMMIES(idx,8) = 1;
profDummy = AGEDUMMIES(:,8);
dummy = [AGE(:,2).*profDummy]; AGEDUMMIES(:,8) = dummy;

% Trades per asset class
load hhChars hhTrdsPerAssetClass;   % HHNum, NumClass, HHNum, NumClass2
TPAC = hhTrdsPerAssetClass(:,[1 4]);

% Get knowledge and exp
load KnowledgeAndExpHH2.txt; kae = KnowledgeAndExpHH2; clear KnowledgeAndExpHH2;
% HHNum, KnowledgeNum (0-4), ExperienceNum (0-4)
% 0: Missing data, 1: NoExp, 2: LittleExp, 3: GoodExp, 4: ExtensiveExp
Nhh = max(size(kae)); KAEDUMMIES = zeros(Nhh,3); KAEDUMMIES(:,1) = kae(:,1);
%[r,c] = find(kae(:,2)==3); KAEDUMMIES(r,2) = 1;
[r,c] = find(kae(:,2)==4); KAEDUMMIES(r,2) = 1;
[r,c] = find(kae(:,2)==0); KAEDUMMIES(r,2) = NaN;
%[r,c] = find(kae(:,3)==3); KAEDUMMIES(r,3) = 1;
[r,c] = find(kae(:,3)==4); KAEDUMMIES(r,3) = 1;
[r,c] = find(kae(:,3)==0); KAEDUMMIES(r,3) = NaN;

Nage = max(size(AGE)); HIGHEXP = zeros(Nage,2); HIGHEXP(:,1) = AGE(:,1);
[expgrps,b] = fndAssignToDeciles(EXPER,10); highexphh = expgrps{5};
idx = ismember(AGE(:,1),highexphh); HIGHEXP(idx,2) = 1;


% TDA*Age Dummy
Nage = max(size(AGE)); AgeDummyInteract = [AGE(:,1) zeros(Nage,1)];
[r,c] = find(TDADUMMY(:,3)==1); tdaHH = TDADUMMY(r,1);
idx = ismember(AgeDummyInteract(:,1),tdaHH); AgeDummyInteract(idx,2) = 1;
AgeDummyInteract(:,2) = [AGE(:,2).*AgeDummyInteract(:,2)];

load infoBaseDivStudy.txt; hhc = infoBaseDivStudy;
% contains: (1)HHNum, (2)IncCode, (3)JobCode, (4)SpouseJobCode, (5)HHAge, (6)SpouseAge,
%           (7)ChildrenBelow18, (8)ChildrenAbove18, (9)MaritalStatus, (10)RentOrOwn,
(11)NumAdults
% RentOrOwn: 0 => Rent, 1 => Own, -1: Unknown.
own = hhc(:,[1 9]); own(own==-1)=NaN;
numAdults = hhc(:,11); numAdults(numAdults==-1)=NaN; famSz = [(hhc(:,7)+hhc(:,8)+numAdults)];
marStatus = hhc(:,9); marStatus(marStatus==-1)=NaN; marStatus = [hhc(:,1) marStatus];
MARITALSTATUS = marStatus; FAMSZ = [hhc(:,1) famSz];


if excludeCAInvestors == 1 | excludeNYInvestors == 1
    % STATE filter
    % Get zipcode-state map
    % CA: 6, NY: 36.
    load ZipcodeStateMap zsMap;   % Zipcode, StateCode
    Nzip = max(size(zsMap)); fprintf(1, '%d zipcodes.\n', Nzip);
    load HHNumAndZipcode hhNumZip;   % HHNum, Zipcode (2 households have duplicate entries:
2869090 and 3916722)
    [r,c] = find(hhNumZip(:,1)==2869090); hhNumZip(r(1),:)=[]; [r,c] =
find(hhNumZip(:,1)==3916722); hhNumZip(r(1),:)=[];
    % => 55431 valid HH

    if excludeCAInvestors == 1
        [r,c] = find(zsMap(:,2)==6); stateZip = zsMap(r,1); Nzip = max(size(r));
```

```
        idx = ismember(hhNumZip(:,2),stateZip); selHH = hhNumZip(idx,1);
    else
        selHH = [];
    end;

    if excludeNYInvestors == 1
        [r,c] = find(zsMap(:,2)==36); stateZip = zsMap(r,1); Nzip = max(size(r));
        idx = ismember(hhNumZip(:,2),stateZip); selHH2 = hhNumZip(idx,1);
        selHH = [selHH; selHH2];
    end;
    idx = ismember(AGE(:,1),selHH); [r,c] = find(idx==0);
    AGE = AGE(r,:); Nz = max(size(AGE));
    fprintf(1, '%d investors remain after the state filter.\n', Nz);
end;

if excludeBottomSIR == 1   % exclude play money accounts
    [sirhh,sirbkpts] = fndAssignToDeciles(SIR,33);  sirbkpts
    bottomSIR = sirhh{1};

    % Strict cutoff
    %[r,c] = find(SIR(:,2)<1); bottomSIR = SIR(r,1);

    idx = ismember(AGE(:,1),bottomSIR); [r,c] = find(idx==0);
    AGE = AGE(r,:); Nz = max(size(AGE));
    fprintf(1, '%d investors remain after the SIR filter.\n', Nz);
end;

if excludeNonIRAInvestors == 1
    % TAX Dummy
    % Taxable vs TDA accounts
    load AccTypeAndSegment.txt;   % contains: HHNum, AccNum, AccType, ClientSegment
    atype = AccTypeAndSegment; clear AccTypeAndSegment;   % 158031 x 4
    % (1) Cash: 42565, (2) Margin: 49737, (3) IRA: 64416, (4) Keogh: 1299, (5) Unknown: 14 (Code
is -1)

    taxHH = unique(atype(:,1)); Nhh = max(size(taxHH));
    % Households with IRAs
    [r,c] = find(atype(:,3)==3|atype(:,3)==4); selHH = unique(atype(r,1));

    idx = ismember(AGE(:,1),selHH); [r,c] = find(idx==0);
    AGE = AGE(r,:); Nz = max(size(AGE));
    fprintf(1, '%d investors with IRA accounts.\n', Nz);

end;


if minPortfSize > 0
    [r,c] = find(PSIZE(:,2)>=minPortfSize); selHH = PSIZE(r,1);
    idx = ismember(AGE(:,1),selHH); AGE = AGE(idx,:); Nz = max(size(AGE));
    fprintf(1, '%d investors remain after the PORTF SIZE filter.\n', Nz);
end;

% Other sub samples based on portf size, turnover, etc.
if turnoverBasedGroup > 0   % consider sub sample based on turnover
    [turnhh,turnbkpts] = fndAssignToDeciles(TURN,33);  turnbkpts
    if turnoverBasedGroup == 1  % low turnover

        selHHTurn = turnhh{1};
    elseif turnoverBasedGroup == 2   % high turnover
        selHHTurn = turnhh{3};
    end;

    idx = ismember(AGE(:,1),selHHTurn); [r,c] = find(idx==0);
    AGE = AGE(r,:); Nz = max(size(AGE));
    fprintf(1, '%d investors remain after the TURNOVER filter.\n', Nz);
end;

AgeAccOpenDateSubSample = 0;   % 1: Low, 2: Med, 3: High

% Sub sample based on age on acc opening date
if AgeAccOpenDateSubSample > 0
```

```
    Z = fndCombineMatrices(AGE,EXPER);
    AgeAccOpenDate = Z(:,2)-Z(:,3)/365;
    [turnhh,turnbkpts] = fndAssignToDeciles([Z(:,1) AgeAccOpenDate],33);
    selHHTurn = turnhh{AgeAccOpenDateSubSample};

    idx = ismember(AGE(:,1),selHHTurn); [r,c] = find(idx==0);
    AGE = AGE(r,:); Nz = max(size(AGE));
    fprintf(1, '%d investors remain after the TURNOVER filter.\n', Nz); turnbkpts, fndPause;
end;
Z = fndCombineMatrices(AGE,EXPER);
AGEOPENDATE = [Z(:,1) Z(:,2)-Z(:,3)/365];


% Sub sample based on first month with returns
load invMonPortfRetBIG R allHH;   % Nhh x 72
R = R'; Nhh = max(size(R));

FROW = zeros(Nhh,1);
for i = 1:Nhh
    [r,c] = find(isnan(R(:,i))==0); FROW(i,1) = min(r);
end;

if useWealth == 1
    INC = WEALTH;
end;

% AGE * Experience Interaction terms
% Old and less experienced
Nage = max(size(AGE)); AGEEXP = zeros(Nage,5); AGEEXP(:,1) = AGE(:,1);
[agegrps,a] = fndAssignToDeciles(AGE,20); younghh = agegrps{1}; oldhh = agegrps{5};
[expgrps,b] = fndAssignToDeciles(EXPER,20); lowexphh = expgrps{1}; highexphh = expgrps{5};
% Young + Less Exp
idx = ismember(younghh,lowexphh); selhh = younghh(idx,1);  sum(idx), idx =
ismember(AGEEXP(:,1),selhh); AGEEXP(idx,2) = 1;
% Young + More Exp (MARKET SAVVY)
idx = ismember(younghh,highexphh); selhh = younghh(idx,1);  sum(idx), idx =
ismember(AGEEXP(:,1),selhh); AGEEXP(idx,3) = 1;
% Old + Less Exp
idx = ismember(oldhh,lowexphh); selhh = oldhh(idx,1);  sum(idx), idx =
ismember(AGEEXP(:,1),selhh); AGEEXP(idx,4) = 1;
% Old + More Exp
idx = ismember(oldhh,highexphh); selhh = oldhh(idx,1);  sum(idx), idx =
ismember(AGEEXP(:,1),selhh); AGEEXP(idx,5) = 1;
AGEEXP = AGEEXP(:,[1 2 4]);   % Less exp interactions
%AGEEXP = AGEEXP(:,[1 3 5]);   % More exp interactions

% NASDAQ and Industry concentration
load NasdaqIndusConc ndqIndusConc allHH;
% NasdaqDummy + 48 FF-Industry
NDQINDUS = [allHH ndqIndusConc];

% Predicted cognitive ability
load c:\1cornell\2research\1behfin\PredictedCogAbility.mat COGABP;
% COGAB * Experience Interaction terms
% Old and less experienced
Nage = max(size(COGABP)); COGABEXP = zeros(Nage,5); COGABEXP(:,1) = COGABP(:,1);
[agegrps,a] = fndAssignToDeciles(COGABP,20); younghh = agegrps{1}; oldhh = agegrps{5};
[expgrps,b] = fndAssignToDeciles(EXPER,20); lowexphh = expgrps{1}; highexphh = expgrps{5};
% Young + Less Exp
idx = ismember(younghh,lowexphh); selhh = younghh(idx,1);  sum(idx), idx =
ismember(COGABEXP(:,1),selhh); COGABEXP(idx,2) = 1;
% Young + More Exp (MARKET SAVVY)
idx = ismember(younghh,highexphh); selhh = younghh(idx,1);  sum(idx), idx =
ismember(COGABEXP(:,1),selhh); COGABEXP(idx,3) = 1;
% Old + Less Exp
idx = ismember(oldhh,lowexphh); selhh = oldhh(idx,1);  sum(idx), idx =
ismember(COGABEXP(:,1),selhh); COGABEXP(idx,4) = 1;
% Old + More Exp
idx = ismember(oldhh,highexphh); selhh = oldhh(idx,1);  sum(idx), idx =
ismember(COGABEXP(:,1),selhh); COGABEXP(idx,5) = 1;
COGABEXP = COGABEXP(:,[1 2 4]);   % Less exp interactions
```

```
%COGABEXP = COGABEXP(:,[1 3 5]);   % More exp interactions

load c:\1cornell\2research\1behfin\27oldinv\2data\MsaSocialCapitalEachHH.mat SOCCAP;
% HHNum, SocialCapital, MinDist
SOCCAP(:,3) = [];


% Old and High Exp
LowHighPctl = 20; LowGrp = 1; HighGrp = 5;
Z = fndCombineMatrices(AGE,EXPER);
OLDLOWEXP = [Z(:,1) fndInteract(Z(:,2),Z(:,3),HighGrp,LowGrp,LowHighPctl)];
OLDHIGHEXP = [Z(:,1) fndInteract(Z(:,2),Z(:,3),HighGrp,HighGrp,LowHighPctl)];
OLDDummy = fndGetHighOrLowDummy(Z(:,2),LowHighPctl,HighGrp);

% NET Performance measures
load NetHouseholdReturns2 R1 R2 R3 R4 AllHH;   % 72 x Nhh
NetPerf = nanmean(R2)';
%NetPerf = nanmean(R2)'./nanstd(R2)';
%PERF = [AllHH NetPerf];

% NET Alpha measures
load FFAlphasUsingNetReturns.mat FFRAW FFNET AllHH;
MinObs = 36;
[r,c] = find(FFNET(:,12)>=MinObs);
%PERF = [AllHH(r,1) FFNET(r,1)];

%PERF = ONETDA;
%PERF = MARGINACC;
%PERF = TPAC;

% SOPHISTICATION
% Investor sophistication data
% Short-sellers
load hhChars sslrsHH;   % HHNum, NumShortPos

% Option traders
load hhChars optTraders;   % HHNum
% Dummy vars
locPerfHH = hhc(:,1);
Nlochh = max(size(locPerfHH));
ISOPHD = zeros(Nlochh,3);   % HH, Short-sellers, Option-traders
idx = ismember(locPerfHH,sslrsHH(:,1)); ISOPHD(idx,2) = 1;
idx = ismember(locPerfHH,optTraders); ISOPHD(idx,3) = 1;
ISOPHD(:,1) = locPerfHH;
%PERF = ISOPHD(:,[1 2]);

load MeanMonthlyExpensesHH.txt; MFEXP = MeanMonthlyExpensesHH; clear MeanMonthlyExpensesHH;
% HHNum, B1, ExpRatio, FrontEndLoad
%PERF = MFEXP(:,[1 3]);

load invMonMutualFundRetHH.txt; R = invMonMutualFundRetHH; clear invMonMutualFundRetHH;
% HHNum, MeanRet, StdDevRet, NumCases
%PERF = [R(:,1) 100*R(:,2)./R(:,3)];

load MeanMutualFundsPosHH.txt;   % HHNum, MeanPosition
MFHOLD = MeanMutualFundsPosHH; clear MeanMutualFundsPosHH;
Nmfinvestors = max(size(MFHOLD)); fprintf(1, '%d mutual fund investors.\n', Nmfinvestors);
Z = fndMergeMatrices(hhc(:,[1 2]),MFHOLD); Z(isnan(Z))=0;
%PERF = [Z(:,1) 100*Z(:,3)./(Z(:,2)+Z(:,3))];

%PERF = INDUSCONC;
%PERF = IR;

load intlForPosAndTradesNewHH WTFOR;   % HHNum, PosForExclMF, PosForMF, PosForAll
%PERF = WTFOR(:,[1 4]);

load TradeClustering ATC; TC = ATC;
load TradeClusteringUsingStksAndSize ATC; TC = ATC;

% Info measure for FOREIGN stocks
minTrades = 2; infoCol = 6;
```

```
load PostTradeRetForStks FORINFOHHBuy FORINFOHHSell allForHH NTR;
[r,c] = find(NTR(:,1) >= minTrades); FORINFOHHBuy = [allForHH(r,1) FORINFOHHBuy(r,infoCol)];
[r,c] = find(NTR(:,2) >= minTrades); FORINFOHHSell = [allForHH(r,1) FORINFOHHSell(r,infoCol)];
FORINFOHH = fndCombineMatrices(FORINFOHHBuy,FORINFOHHSell);
FORINFOHH = [FORINFOHH(:,1) 100*(FORINFOHH(:,2)-FORINFOHH(:,3))];
N = max(size(FORINFOHH)); fprintf(1, '%d investors with foreign info measure.\n', N);
FORINFOHH(:,2) = fndTruncateDistrib(FORINFOHH(:,2),0.25);


% HH with internet use
load ListHHWithInternetUse.txt;
InternetHH = ListHHWithInternetUse; clear ListHHWithInternetUse;
Nhh = max(size(hhc)); INTERNETHH = zeros(Nhh,2); INTERNETHH(:,1) = hhc(:,1);
idx = ismember(hhc(:,1), InternetHH); INTERNETHH(idx,2) = 1;
%PERF = INTERNETHH;

if ConsiderInternetUsersOnly == 1   % Only look at Internet users
    idx = ismember(AGE(:,1),InternetHH); [r,c] = find(idx==1);
    AGE = AGE(r,:); Nz = max(size(AGE));
    fprintf(1, '%d investors remain after the INTERNET filter.\n', Nz);
end;



if modelNum == 1 | modelNum == 6 | modelNum == 201 | modelNum == 901 | modelNum == 1001 |
modelNum == 2001
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    y = Z(:,3); x = Z(:,[2 4]);
elseif modelNum == 2
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract

    y = 100*Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 12:15 19]);
elseif modelNum == 2.1
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
```

```
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    Z = fndCombineMatrices(Z,VOLDDUMMY);
    Z = fndCombineMatrices(Z,LOWINC);
    Z = fndCombineMatrices(Z,LOWEDU);
    Z = fndCombineMatrices(Z,RACEDUMMIES);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract

    y = 100*Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 12:15 19 20 21 22 23 24]);
    y = 100*Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 12:15 19 20 21 22 23 24 25]);
elseif modelNum == 2.2
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    Z = fndCombineMatrices(Z,VOLDDUMMY);
    Z = fndCombineMatrices(Z,LOWINC);
    Z = fndCombineMatrices(Z,LOWEDU);
    Z = fndCombineMatrices(Z,RACEDUMMIES);
    Z = fndCombineMatrices(Z,NDQINDUS);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract

    y = 100*Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 12:15 19 20 21 22 23 24]);
    y = 100*Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 17 19 20 21 22 23 24 25 26:74]);
elseif modelNum == 3
    Z = fndCombineMatrices(AGE,TURN);
    Z = fndCombineMatrices(Z,EXPER);
    y = Z(:,3); x = Z(:,[2 4]);
elseif modelNum == 4
    Z = fndCombineMatrices(AGE,TURN);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
```

```
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract

    y = Z(:,3); x = Z(:,[2 4 6 8 10 11 5 9 12 13:15]); x(:,1) = log(x(:,1));
elseif modelNum == 5
    Z = fndCombineMatrices(AGE,DY);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract

    y = Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 12 13:15]); %x(:,1) = log(x(:,1));
elseif modelNum == 16
    Z = fndCombineMatrices(AGE,ADE);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract

    y = Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 12 13:15]); %x(:,1) = log(x(:,1));
elseif modelNum == 7
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    y = Z(:,3); x = Z(:,[2 4]);
elseif modelNum == 8 | modelNum == 11
    Z = fndCombineMatrices(AGE,PERF);
```

```
    Z = fndCombineMatrices(Z,EXPER);
    y = Z(:,3); x = Z(:,[2 4]);
elseif modelNum == 9 | modelNum == 10 | modelNum == 12 | modelNum == 210 | modelNum == 909 |
modelNum == 1009 ...
        | modelNum == 69 | modelNum == 2009 | modelNum == 99
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1:3]));
    %Z = fndCombineMatrices(Z,COGABP);
    %Z = fndCombineMatrices(Z,NDQINDUS);
    %Z = fndCombineMatrices(Z,AGEEXP(:,[1 3]));
    %Z = fndCombineMatrices(Z,METRO);
    %Z = fndCombineMatrices(Z,TDADUMMY(:,[1 3]));
    %Z = fndCombineMatrices(Z,VOLDDUMMY);
    %Z = fndCombineMatrices(Z,TDADummy);
    %Z = fndCombineMatrices(Z,AgeDummyInteract);
    %Z = fndCombineMatrices(Z,KAEDUMMIES);
    %Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1 7]));
    %Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1 8]));
    %Z = fndCombineMatrices(Z,SR);
    %Z = fndCombineMatrices(Z,AGERANGE);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract
    % (21)Old*PoorDummy (22)Old*UneducatedDummy (23)MetroDummy
    % (24)VeryOldDummy (25)TDADummy (26)AgeTDAInteract (27)CogAbility

    y = Z(:,3); x = Z(:,[2 4 17 20 21 22 6 8 10 11 5 7 12 9 13:15 19]); %x(:,1) = log(x(:,1));
    %y = Z(:,3); x = Z(:,[23 4 5 7]);
    %y = Z(:,3); x = Z(:,[2 4 17 20 21 22]);
elseif modelNum == 101 | modelNum == 102
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    Z = fndCombineMatrices(Z,RACEAGE2);
    Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1:3]));
    Z = fndCombineMatrices(Z,METRO);
    %Z = fndCombineMatrices(Z,TDADUMMY(:,[1 3]));
    Z = fndCombineMatrices(Z,AgeDummyInteract);
    %Z = fndCombineMatrices(Z,KAEDUMMIES);
```

```
    %Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1 7]));
    %Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1 8]));
    %Z = fndCombineMatrices(Z,SR);
    %Z = fndCombineMatrices(Z,AGERANGE);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Hisp*AgeInteraction (18)Div (19)Mfunds (20)AfrAm*AgeInteract
    % (21)Old*PoorDummy (22)Old*UneducatedDummy (23)MetroDummy
    % (24)AgeExperDummy
    % (24)Old*MetroDummy (25)Old*ProfDummy (26)TradesPerAssetClass
    % (27-30)AgeDecadeDummies

    y = Z(:,3); x = Z(:,[2 6 8 10 11 5 7 12 9 13:15 17 20 21 22 24]); %x(:,1) = log(x(:,1));
    y = Z(:,3); x = Z(:,[2]);
elseif modelNum == 103 | modelNum == 104  | modelNum == 202
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,TURN);
    %Z = fndCombineMatrices(Z,GENDER);
    %Z = fndCombineMatrices(Z,marStatus);
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)PortfTurnover
    % (7)MaleDummy (8)MarriedDummy

    y = Z(:,3); x = Z(:,[2 4 5 6]); %x(:,1) = log(x(:,1));
elseif modelNum == 2002
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,DY);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
    Z = fndCombineMatrices(Z,FACTEXPO);
    Z = fndCombineMatrices(Z,IR);
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,DIV);
    Z = fndCombineMatrices(Z,MFUNDS);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,LB);
    %Z = fndCombineMatrices(Z,GENDERAGE);
    %Z = [Z fndStd(Z(:,2)).*fndStd(Z(:,8))];
    %Z = fndCombineMatrices(Z,[allHH ndqIndusConc]); size(Z)
    %Z = fndCombineMatrices(Z,HHCensus(:,[1 3]));
    % (1)HHNum (2)Age (3)Perf (4)InvExper (5)PortfSize (6)Income
    % (7)PortfTurnover (8)Education (9)PortfDivYield (10)MaleDummy
    % (11)RetirementDummy (12)RMRF (13)SMB (14)HML (15)UMD (16)IR
    % (17)Race*AgeInteraction (18)Div (19)Mfunds

    y = Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9 12:15 19]);
    %y = Z(:,3); x = Z(:,[2 4 6 8 10 11 5 7 9]);
    %y = Z(:,3); x = Z(:,[2 4 17 18]);
    %y = Z(:,19); x = Z(:,[2 4 6 8 10 11 5 7 9 12 13:15 19]);
elseif modelNum == 3001
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,COGABP);
    Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,TURN);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,GENDER);
    Z = fndCombineMatrices(Z,RETDUMMY);
```

```
    %Z = fndCombineMatrices(Z,FACTEXPO);
    %Z = fndCombineMatrices(Z,RACEAGE);
    %Z = fndCombineMatrices(Z,RACEAGE2);

    y = Z(:,3); x = Z(:,[5 4 6 7 8 9 10]);
    y = Z(:,3); x = Z(:,[5 4 6 7 8 9 10]);
elseif modelNum == 9999
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,PSIZE);
    %Z = fndCombineMatrices(Z,FACTEXPO);
    %Z = fndCombineMatrices(Z,COGABP);
    LowExpYoungInteract = fndInteract(Z(:,4),Z(:,2),1,1,20);
    LowExpOldInteract = fndInteract(Z(:,4),Z(: ,2),1,5,20);
    LargePortfOldInteract = fndInteract(Z(:,5),Z(:,2),5,5,20);
    %RiskyPortfLowCogabInteract = fndInteract(Z(:,6),Z(:,10),5,1,20);

    y = 100*Z(:,3); x = [Z(:,[2 4 5]) LowExpYoungInteract LowExpOldInteract
LargePortfOldInteract];
elseif modelNum == 9999.1
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,SOCCAP);
    Z = fndCombineMatrices(Z,VOLDDUMMY);
    %Z = fndCombineMatrices(Z,INDUSCONC); [r,Nz] = size(Z);

    y = Z(:,3); x = [Z(:,[2 4 5]) Z(:,2).*Z(:,5)];
    y = Z(:,3); x = [Z(:,[2 4 5 6])];
elseif modelNum == 9999.2
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,MFWEIGHT);
    Z = fndCombineMatrices(Z,VOLDDUMMY);
    %LowExpYoungInteract = fndInteract(Z(:,4),Z(:,2),1,1,20);

    y = Z(:,3); x = [Z(:,[2 4 5 6]) Z(:,5).*Z(:,6)];
elseif modelNum == 9999.3
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,VOLDDUMMY);
    Z = fndCombineMatrices(Z,INC);
    Z = fndCombineMatrices(Z,LOWINC);
    Z = fndCombineMatrices(Z,EDU);
    Z = fndCombineMatrices(Z,LOWEDU);
    Z = fndCombineMatrices(Z,OLDINC);
    Z = fndCombineMatrices(Z,OLDEDU);
    Z = fndCombineMatrices(Z,GENDER);
    %LowExpYoungInteract = fndInteract(Z(:,4),Z(:,2),1,1,20);

    y = Z(:,3); x = [Z(:,[2 4 5 6 7 8 9 10 11 12])];
elseif modelNum == 9999.4
    KAE = [KAEDUMMIES(:,1) nanmax(KAEDUMMIES(:,2:3)')'];
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    %Z = fndCombineMatrices(Z,GENDER);
    %Z = fndCombineMatrices(Z,HIGHEXP);
    %Z = fndCombineMatrices(Z,OLDINC);
    %Z = fndCombineMatrices(Z,OLDEDU);
    %Z = fndCombineMatrices(Z,AGEANDSQ);
    %Z = fndCombineMatrices(Z,INTERNETHH);
    %Z = fndCombineMatrices(Z,PSIZE);
    Z = fndCombineMatrices(Z,RACEDUMMIES);
    %Z = fndCombineMatrices(Z,OLDHIGHEXP);
    %Z = fndCombineMatrices(Z,INC);
    %Z = fndCombineMatrices(Z,EDU);
    %Z = fndCombineMatrices(Z,LOWINC);
    %Z = fndCombineMatrices(Z,LOWEDU);
    %Z = fndCombineMatrices(Z,AGEDUMMIES(:,[1:3]));
    Z = fndCombineMatrices(Z,RACEAGE);
    Z = fndCombineMatrices(Z,RACEAGE2);
```

```
    %y = Z(:,3); x = [Z(:,[5 4 7]) Z(:,5).*(1-Z(:,7))];
    %y = Z(:,3); x = [Z(:,[2 4 5 6 7 8]) Z(:,5).*(Z(:,6))];
    y = Z(:,3); x = [Z(:,[2 4 5 6 7 8])];
elseif modelNum == 9999.9
    Z = fndCombineMatrices(AGE,PERF);
    Z = fndCombineMatrices(Z,EXPER);
    Z = fndCombineMatrices(Z,AGEANDSQ);

    y = Z(:,3); x = [Z(:,[5 6 4])];
end;

% OVER-WRITE EVERYTHING
%[r,c] = find(MFUNDS(:,2)==0); MFUNDS(r,:) = [];
%    Z = fndCombineMatrices(AGE,PERF);
%    Z = fndCombineMatrices(Z,EXPER);
%    Z = fndCombineMatrices(Z,MFUNDS);
%    y = Z(:,5); x = Z(:,[2 4]);

if trimDistrib == 1
    y = fndTrimDistrib(y,trimPct); x = fndTrimDistrib(x,trimPct);
else
    y = fndTruncateDistrib(y,trimPct); x = fndTruncateDistrib(x,trimPct);
end;
x = fndStd(x); y = 100*y;
%y = fndStd(y);
[b,r2,rbar2,tval,fval,dw,Ncases,pval] = fndRegress(y,x);
fndPrintRegRes(b,r2,rbar2,tval,fval,dw,Ncases,pval);
vif = fndMultiCollTests(x); vif
return;

%if regNum == 1
%    A = [[b tval]; [NaN Ncases]; [NaN r2]];
%else
%    AA = [[b tval]; [NaN Ncases]; [NaN r2]];
%    A = [A AA];
%end;
%A

%a = Z(:,23:71); a(isnan(a))=1; A = nansum(a); [r,c] = find(A>0); c
y = Z(:,3); x = Z(:,[2 4 17 20 21 22 6 8 10 11 5 7 12 9 13:15 19 23:42]);
if trimDistrib == 1
    y = fndTrimDistrib(y,trimPct); x = fndTrimDistrib(x,trimPct);
else
    y = fndTruncateDistrib(y,trimPct); x = fndTruncateDistrib(x,trimPct);
end;
x = fndStd(x); y = fndStd(y);
[b,r2,rbar2,tval,fval,dw,Ncases,pval] = fndRegress(y,x);
fndPrintRegRes(b,r2,rbar2,tval,fval,dw,Ncases,pval);
z = [y x]; [r,c] = find(isnan(z)); r = unique(r); z(r,:) = [];
save c:\1ut\Stata.txt z -ascii;


%
% OldAgeAndPerfUnivar.m
%
% AK, 6/21/07

UseRawRet = 0;
TruncPctl = 0.5;
minMonths = 6;
AgePctl = 12.5; Nagegrps = floor(100/AgePctl);
MinPortfSize = 10000; MaxPortfSize = 100000000;
%MinPortfSize = 25000; MaxPortfSize = 100000000;

AgeBkPts2 = [
    10    30
    30    35
    35    40
    40    45
    45    50
    50    55
```

```
    55    60
    60    65
    65    70
    70    75
    75    100
    ];

AgeBkPts2 = [
    10    32
    32    36
    36    40
    40    44
    44    48
    48    52
    52    56
    56    64
    64    72
    72    100
    ];
Nagegrps2 = max(size(AgeBkPts2));

% PERF
if UseRawRet == 0
    % Char adjusted returns
    load invMonCharAdjPortfRetBIG R allHH;   % Nhh x 72
else
    % Raw Returns
    load invMonPortfRetBIG R allHH;   % 55277 x 72
end;
load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
[r,c] = find(alph(:,2)>=minMonths); selHH = alph(r,1);
idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(r,1);

% AGE
load hhChars hhc;
% hhc(9): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive
[r,c] = find(hhc(:,2)>=MinPortfSize & hhc(:,2)<MaxPortfSize); selHH = hhc(r,1);
idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(idx,1);
hhc(hhc==-999) = NaN; AGE = hhc(:,[1 8]);
[r,c] = find(isnan(AGE(:,2))); AGE(r,:) = [];
PERF = [allHH 12*100*nanmean(R')'];
PERF(:,2) = fndTruncateDistrib(PERF(:,2),TruncPctl);

z = fndCombineMatrices(hhc(:,[1 8]),PERF); size(z)
% HHNum, Age, Perf

plot(z(:,2),z(:,3),'.'); grid on;

% Age based categories
[AgeGrps,AgeBkPts] = fndAssignToDeciles(AGE,AgePctl);
AgePerf = []; N = [];
for i = 1:Nagegrps
    idx = ismember(PERF(:,1),AgeGrps{i}); N = [N; sum(idx)];
    AgePerf = [AgePerf; nanmean(PERF(idx,2))];
end;
N'
x = nanmean(AgeBkPts,2); y = AgePerf;
clf; bar(y,'g'); grid on;

% Age based categories
AgePerf = []; N = [];
for i = 1:Nagegrps2
    [r,c] = find(AGE(:,2)>=AgeBkPts2(i,1) & AGE(:,2)<AgeBkPts2(i,2)); SelHH = AGE(r,1);
    idx = ismember(PERF(:,1),SelHH); N = [N; sum(idx)];
    AgePerf = [AgePerf; nanmedian(PERF(idx,2))];
end;
N'
x = nanmean(AgeBkPts2,2); y = AgePerf;
clf; bar(y,'g'); grid on;
```

```
% LABEL THE PLOT
xlabel('AGE','FontName','Times','FontSize',14);
set(gca, 'XTick', 0:1:10);
xlabs = '|Below 32|32-36|36-40|40-44|44-48|48-52|52-56|56-64|64-72|Above 72';
set(gca, 'XTickLabel', xlabs, 'FontName','Times','FontSize',10);
ylabel('Annual Characterisitc-Adjusted Performance', 'FontName','Times','FontSize',14);
axis([0 12 -2 2]);

%print -depsc2 e:\1cornell\2research\1behfin\27OldInv\3wpaper\AgeAndPerfMean.eps;


%
% oldPortfSizeAndPerfCogAbPortf.m
%

minMonths = 1;
strtMonPerf = 1; lstMonPerf = 72; Nmonths = lstMonPerf - strtMonPerf + 1;
cogAbPctl = 10; Ncogabgrps = floor(100/cogAbPctl); MinPortfSize =  000; MaxPortfSize = 10000000;
% For (3)
%cogAbPctl = 10; Ncogabgrps = floor(100/cogAbPctl); MinPortfSize =  1000; MaxPortfSize =
10000000;   % For (2)
cogAbPctl = 10; Ncogabgrps = floor(100/cogAbPctl); MinPortfSize =  1000; MaxPortfSize = 10000000;
% For (6)
cogAbPctl = 20; Ncogabgrps = floor(100/cogAbPctl); MinPortfSize =  000; MaxPortfSize = 10000000;
% For (6)
UseRawRet = 0;
InvExpPctl = 50; Ninvexpgrps = floor(100/InvExpPctl); SelInvExpGrp = 0;
SavePlot = 0;
UseResidualCogAb = 0;   % remove the effect of experience
UseVW = 0;
FilterType = 3;   % (1)PortfSize (2)LocBias (3)Concentration (6)Turnover
excludeBottomSIR = 0;

PortfSize = [
    1 5000;
    5001 10000;
    10001 25000;
    25000 75000;
    75001 9999999
    ];

PortfSize = [
    1 9999999;
    1 10000;
    10001 50000;
    50001 9999999
    ];

PortfSize = [
    1 9999999;
    ];

[Nportf,c] = size(PortfSize);

if FilterType > 1
    PortfPctl = 20; Nportf = floor(100/PortfPctl);
end;

if FilterType == 5
    Nportf = 2;
end;

z = []; N = [];
for i = 1:Nportf
    if UseRawRet == 0
        % Char adjusted returns
        load invMonCharAdjPortfRetBIG R allHH;   % Nhh x 72
    else
        % Raw Returns
        load invMonPortfRetBIG R allHH;   % 55277 x 72
```

```
    end;
    load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
    [r,c] = find(alph(:,2)>=minMonths); selHH = alph(r,1); BETA = alph(r,[1 5]);
    idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(r,1);


    % Risk-adjusted returns
    % Jan 1983 to Dec 2005
    load FFData fffmSm;   % 72 x 6 (1991 to 1996)
    Rm = fffmSm(:,2:5); Rf = fffmSm(:,6);

    % Portf Size Filter
    load hhChars hhc;
    % hhc(9): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
    %         (9)NmonthsActive

    if MinPortfSize > 0
        [r,c] = find(hhc(:,2)>=MinPortfSize & hhc(:,2)<MaxPortfSize); selHH = hhc(r,1);
        idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(idx,1);
    end;

    if FilterType == 1
        [r,c] = find(hhc(:,2)>=PortfSize(i,1) & hhc(:,2)<PortfSize(i,2)); selHH = hhc(r,1);
        idx = ismember(selHH,allHH);  PSIZE = hhc(idx,[1 2]);
    elseif FilterType == 2
        % LOCAL BIAS
        load LocalBias LB; % HHNum, LBOld, LBNew
        LB = LB(:,[1 3]);
        %load hhChars EXPER; LB = EXPER;
        [LocBiasGrps,LocBiasBkPts] = fndAssignToDeciles(LB,PortfPctl);
        selHH = LocBiasGrps{i};
    elseif FilterType == 3
        load hhChars hdivcAllP;   % 41039 x 6
        DIV = [hdivcAllP(:,1) -hdivcAllP(:,3)];
        %DIV = [hdivcAllP(:,1) hdivcAllP(:,5)];
        % hdivcAllP contains: HHNum, Nobs, Nstks, Sqw, NormVar, AvgCorrel
        [DivGrps,DivBkPts] = fndAssignToDeciles(DIV,PortfPctl); selHH = DivGrps{i}; DivBkPts
    elseif FilterType == 5
        load infoBaseDivStudy.txt; hhc = infoBaseDivStudy;
        % contains: (1)HHNum, (2)IncCode, (3)JobCode, (4)SpouseJobCode, (5)HHAge, (6)SpouseAge,
        %           (7)ChildrenBelow18, (8)ChildrenAbove18,
        %           (9)MaritalStatus, (10)RentOrOwn, (11)NumAdults
        marStatus = hhc(:,9); marStatus(marStatus==-1)=NaN;
        [r,c] = find(marStatus==i-1); selHH = hhc(r,1);

        % LOCATION
        % Metro-Non-Metro status
        load MetroNonMetroStatusEachHH GP;   % HHNum, Metro-Non-Metro Status
        % MetroNonMetroStatusEachHH2: with 20 metros
        % load FinCenterStatusEachHH GP;
        %load NewYorkStatusEachHH GP;

        Nmetrohh = max(size(GP));
        METRO = zeros(Nmetrohh,2); METRO(:,1) = GP(:,1);
        met = GP(:,2); met(met>1)=0;  % Metro dummy
        remote = GP(:,2); remote(remote==1)=0; remote(remote==2)=0; remote(remote==3)=1;
        METRO(:,2) = met; %METRO(:,3) = remote;
        [r,c] = find(METRO(:,2)==i-1); selHH = METRO(r,1);

        % GENDER
        load infobaseWithGender.txt; gender = infobaseWithGender(:,[1 12]);
        [r,c] = find(gender(:,1)==2869090); gender(r(1),:)=[];
        [r,c] = find(gender(:,1)==3916722); gender(r(1),:)=[];
        load hhChars ahmap;
        [r,c] = find(ahmap(:,1)==2869090); ahmap(r(1),:)=[];
        [r,c] = find(ahmap(:,1)==3916722); ahmap(r(1),:)=[];
        [r,c] = find(ahmap(:,1)==3844354); ahmap(r(1),:)=[];
        [r,c] = find(ahmap(:,1)==8827950); ahmap(r(1),:)=[];
        gender = fndCombineMatrices(gender,ahmap); gender = gender(:,[3 2]);
        [r,c] = find(gender(:,1)==8862950); gender(r(1),:)=[];
```

```
        % gender contains: HHNum, Gender (0: Female, 1: Male, -1: Unknown)
        gender(gender==-1)=NaN; GENDER = gender;
        [r,c] = find(GENDER(:,2)==i-1); selHH = GENDER(r,1);
    elseif FilterType == 6
        load hhChars htc hhc5min; hhc = hhc5min; clear hhc5min;
        hhc = sortrows(hhc,1); htc = sortrows(htc,1); hhc = [hhc htc(:,2:6)];
        turn = nanmean(hhc(:,[10 11])')';  TURN = [hhc(:,1) turn];
        %TURN = [hhc(:,1) hhc(:,11)];
        [InvGrps,InvBkPts] = fndAssignToDeciles(TURN,PortfPctl); selHH = InvGrps{i};
    elseif FilterType == 9    % Test cases
        load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
        [r,c] = find(alph(:,2)>=minMonths); selHH = alph(r,1); ALPHA = alph(r,[1 5]);
        [InvGrps,InvBkPts] = fndAssignToDeciles(ALPHA,PortfPctl); selHH = InvGrps{i};
    elseif FilterType == 10
        oldAllHH = allHH;
        load TradeClustering ATC; ATC(:,2) = -ATC(:,2);
        load AdjustedDispEffect ADE;
        [InvGrps,InvBkPts] = fndAssignToDeciles(ATC,PortfPctl); selHH = InvGrps{i};
        allHH = oldAllHH;
    end;

    % Returns for selected HH
    idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(idx,1); Nhh = sum(idx); N = [N;
Nhh];

    if SelInvExpGrp > 0
        % INVESTMENT EXPER Filter
        load hhChars EXPER;   % HHNum, InvExperience

        [invexpgrps,invexpbkpts] = fndAssignToDeciles(EXPER,InvExpPctl);

        selHH = invexpgrps{SelInvExpGrp};
        idx = ismember(allHH,selHH); R = R(idx,:); allHH = allHH(idx,1); Nhh = sum(idx); N = [N;
Nhh];
    end;

    if excludeBottomSIR == 1   % exclude play money accounts
        load hhChars hhc;
        icd = hhc(:,6);
        icd(icd == 1) = 7.5; icd(icd == 2) = 17.5; icd(icd == 3) = 25; icd(icd == 4) = 35;
        icd(icd == 5) = 45; icd(icd == 6) = 62.5; icd(icd == 7) = 87.5; icd(icd == 8) = 112.5;
        icd(icd == 9) = 250;    % ref: Ivkovich:Weisbenner
        SIR = [hhc(:,1) hhc(:,2)./(1000*hhc(:,6))];

        [sirhh,sirbkpts] = fndAssignToDeciles(SIR,50);  sirbkpts
        bottomSIR = sirhh{1};
        idx = ismember(allHH,bottomSIR); R(idx,:) = []; allHH(idx,:) = []; Nhh =
max(size(allHH)); N = [N; Nhh]; size(R), fndPause;
    end;

    % Predicted cognitive ability
    load C:\1cornell\2research\1behfin\PredictedCogAbility.mat COGABP;
    load PredictedCogAbilityNEW.mat COGABP;
    load PredictedCogAbilityWithImpute COGABP;
    % HHNum, CogAb
    %load hhChars hhc; hhc(hhc==-99) = NaN; hhc(hhc==-999) = NaN; COGABP = hhc(:,[1 8]);

    if UseResidualCogAb == 1
        % INVESTMENT EXPER
        load hhChars EXPER;   % HHNum, InvExperience

        Z = fndCombineMatrices(COGABP,EXPER);
        y = Z(:,2); x = Z(:,3);
        [b,r2,rbar2,tval,fval,dw,Ncases,pval] = fndRegress(y,x);
%fndPrintRegRes(b,r2,rbar2,tval,fval,dw,Ncases,pval);
        yhat = b(1) + b(2)*x(:,1);
        COGABP = [Z(:,1) (y - yhat)];
    end;

    % Low and high cog ab investors
    [cogabgrps,cogabbkpts] = fndAssignToDeciles(COGABP,cogAbPctl);
```

```
    Rp = NaN*zeros(Nmonths,Ncogabgrps);
    for j = 1:Ncogabgrps
        idx = ismember(allHH,cogabgrps{j});
        if UseVW == 1
            w = PSIZE(idx,2); w = w./nansum(w); w = w*ones(1,72);
            Rp(:,j) = 100*nansum(R(idx,:).*w)';
        else
            Rp(:,j) = 100*nanmean(R(idx,:))';
        end;
    end;

    [nanmean(Rp); nanmedian(Rp); nanstd(Rp)]


    Rpd = Rp(:,Ncogabgrps)-Rp(:,1); MnDiff = 12*nanmean(Rpd);
    t = (71^0.5)*nanmean(Rpd)/nanstd(Rpd);
    y = Rpd; x = Rm;
    [b,r2,rbar2,tval,fval,dw,n,pval] = fndRegress(y,x);

    z = [z; [12*nanmean(Rp(:,[1 Ncogabgrps])) MnDiff t 12*b(1) tval(1) b(2:5)' tval(2:5)']];
end;

z, N
clf; bar(z(:,1:3));
grid on; colormap hot;

if FilterType == 1
    % LABEL THE PLOT
    xlabel('Portfolio Size','FontName','Times','FontSize',14);
    set(gca, 'XTick', 0:1:4);
    xlabs = '|All|Small (Below $10,000)|Medium ($10,000-$50,000)|Large (Above $50,000)';
    set(gca, 'XTickLabel', xlabs, 'FontName','Times','FontSize',11);
    ylabel('Annual Risk-Adjusted Performance', 'FontName','Times','FontSize',14);
    legend('Low Cognitive Ability','High Cognitive Ability','High Minus
Low','Location','NorthWest');
    %axis([0 5 -2 4]);
elseif FilterType == 2
    % LABEL THE PLOT
    xlabel('Local Bias Quintiles','FontName','Times','FontSize',14);
    set(gca, 'XTick', 1:1:6);
    xlabs = 'Q1 (Low)|Q2|Q3|Q4|Q5 (High)|';
    set(gca, 'XTickLabel', xlabs, 'FontName','Times','FontSize',8);
    ylabel('Annual Char-Adjusted Performance', 'FontName','Times','FontSize',14);
    legend('Low Cognitive Ability','High Cognitive Ability','High Ability - Low
Ability','Location','NorthWest');
    axis([0 6 -2 7]);
elseif FilterType == 3
    % LABEL THE PLOT
    xlabel('Portfolio Concentration Quintiles','FontName','Times','FontSize',14);
    set(gca, 'XTick', 1:1:6);
    xlabs = 'Q1 (Low)|Q2|Q3|Q4|Q5 (High)|';
    set(gca, 'XTickLabel', xlabs, 'FontName','Times','FontSize',8);
    ylabel('Annual Char-Adjusted Performance', 'FontName','Times','FontSize',14);
    legend('Low Cognitive Ability','High Cognitive Ability','High Ability - Low
Ability','Location','NorthWest');
    axis([0 6 -4 7]);
elseif FilterType == 6
    % LABEL THE PLOT
    xlabel('Portfolio Turnover Quintiles','FontName','Times','FontSize',14);
    set(gca, 'XTick', 1:1:6);
    xlabs = 'Q1 (Low)|Q2|Q3|Q4|Q5 (High)|';
    set(gca, 'XTickLabel', xlabs, 'FontName','Times','FontSize',8);
    ylabel('Annual Char-Adjusted Performance', 'FontName','Times','FontSize',14);
    legend('Low Cognitive Ability','High Cognitive Ability','High Ability - Low
Ability','Location','NorthWest');
    axis([0 6 -3 7]);

end;

if FilterType == 1
```

```
    %print -depsc2 e:\1cornell\2research\1behfin\27OldInv\3wpaper\CogAbAndPerf.eps;
elseif FilterType == 2
    %print -depsc2 e:\1cornell\2research\1behfin\27OldInv\3wpaper\CogAbAndLocalBias.eps;
elseif FilterType == 3
    %print -depsc2 e:\1cornell\2research\1behfin\27OldInv\3wpaper\CogAbAndPortfConc.eps;
elseif FilterType == 6
    %print -depsc2 e:\1cornell\2research\1behfin\27OldInv\3wpaper\CogAbAndTrading.eps;
end;
%
% oldStockPrefsYoungAndOldInv.m
% Compare the stock preferences of younger and older investors
%
% AK, 7/6/05

runNum = 1; useDiff = 0;   % useDiff is used only when runNum = 2.
agePctl = 33; Nagegrps = floor(100/agePctl);
oldAgeCutoff = 60; considerOldOnly = 0;
trimDistrib = 0; trimPct = 1;
remZeros = 1;
loadData = 1;
excludeCAInvestors = 0;
minPriceFilter = 0;

inoutFile = 'C:\1cornell\2research\1behfin\64amtosh\2data\StockPrefIPOAndNonIPOHH.mat';

load hhChars hhc;
% hhc(9): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive
hhc(hhc==-999)=NaN; AGE = hhc(:,[1 8]);

% Old and young investors
[r,c] = find(AGE(:,2)>=oldAgeCutoff); Nold = max(size(r));
if considerOldOnly == 1
    AGE = AGE(r,:); hhc = hhc(r,:);
end;
[agegrps,agebkpts] = fndAssignToDeciles(AGE,agePctl); agebkpts
oldHH = agegrps{Nagegrps-1}; Noldhh = max(size(oldHH)); fprintf(1, '%d older investors.\n',
Noldhh);
youngHH = agegrps{1};
Nyounghh = max(size(youngHH)); fprintf(1, '%d younger investors.\n', Nyounghh);


% Get the aggregate retail portfolio
load invAggregateMonPortf aip allStks;   % aip contains: 9673 x 72 matrix (last month empty)
% Contains the end of month positions
posStks = allStks;

if runNum == 1
    selHH = oldHH;
else
    selHH = youngHH;
end;

if excludeCAInvestors == 1
    % STATE filter
    % Get zipcode-state map
    % CA: 6, NY: 36.
    load ZipcodeStateMap zsMap;   % Zipcode, StateCode
    Nzip = max(size(zsMap)); fprintf(1, '%d zipcodes.\n', Nzip);
    load HHNumAndZipcode hhNumZip;   % HHNum, Zipcode (2 households have duplicate entries:
2869090 and 3916722)
    [r,c] = find(hhNumZip(:,1)==2869090); hhNumZip(r(1),:)=[]; [r,c] =
find(hhNumZip(:,1)==3916722); hhNumZip(r(1),:)=[];
    % => 55431 valid HH

    [r,c] = find(zsMap(:,2)==6); stateZip = zsMap(r,1); Nzip = max(size(r));
    idx = ismember(hhNumZip(:,2),stateZip); caHH = hhNumZip(idx,1);
    idx = ismember(selHH,caHH);   selHH(idx,:) = [];
end;
```

```
Nhh = max(size(selHH)); fprintf(1, '%d households chosen.\n', Nhh);

% Get the portfolio weights for this subset of investors
aip = dyGetAggInvPortf(posStks,selHH);   % aip is Nstks x 71

if loadData == 1
    % Get the three return moments for all stocks
    load crspMon3RetMomBIG Mu Sd Skew allStks;   % 276 x 21706 matrix (1980 to 2002)
    % Get 1991-96 data (actually the data is for Dec 1990 - Oct 1996)
    Mu = Mu(132:202,:); Sd = Sd(132:202,:); Skew = Skew(132:202,:);

    % Control variables: MktBeta, size, BM, Momentum
    load crspMonMomentumBIG M1 M3 M12 allStks; % 276 x 21706 matrix (1/1980 to 12/2002)
    rowNum1 = 132; rowNum2 = 132+70;
    MOM1 = M1(rowNum1:rowNum2,:); MOM2 = M3(rowNum1:rowNum2,:); MOM3 = M12(rowNum1:rowNum2,:);

    load crspStkMultVarsBIG P SO BM;
    SZ = P .* SO; [Nr,Ncols] = size(SZ);
    SZ = SZ(rowNum1:rowNum2,:); BM = BM(rowNum1:rowNum2,:); %R = R(rowNum1:rowNum2,:);
    VW = SZ./(nansum(SZ')'*ones(1,Ncols));
    oldAllStks = allStks;

    % Monthly turnover
    load crspStkMultVarsBIG V; TURN = V; clear V SO;    % 276 x Nstks matrix

    % Market Beta
    load crspStkBetas80To01BIG B allStks; % 264 x 22896
    idx = ismember(allStks,oldAllStks); B = B(rowNum1:rowNum2,idx);
    allStks = oldAllStks;

    % Idiosyncratic Risk
    load crspStkIdioSyncRisk80To01BIG IR allStks; % 276 x 22896 matrix (1/1980 to 12/2002)
    idx = ismember(allStks,oldAllStks); IR = IR(rowNum1:rowNum2,idx);
    allStks = oldAllStks;

    % Kurtosis
    oldAllStks = allStks;
    load crspStkKurtosis80To02BIG KURT allStks; % 264 x 22896 matrix (1/1980 to 12/2001)
    idx = ismember(allStks,oldAllStks); KURT = KURT(rowNum1:rowNum2,idx);
    allStks = oldAllStks;

    idx = ismember(allStks,posStks); idx2 = ismember(posStks,allStks); matchStks =
allStks(idx,1);
    Mu = Mu(:,idx); Sd = Sd(:,idx); Skew = Skew(:,idx); aip = aip(idx2,:);
    MOM1 = MOM1(:,idx); MOM2 = MOM2(:,idx); MOM3 = MOM3(:,idx); SZ = SZ(:,idx); TURN =
TURN(:,idx); P = P(:,idx);
    BM = BM(:,idx); B = B(:,idx); IR = IR(:,idx);
    KURT = KURT(:,idx);
    Nmatches = max(size(matchStks));

    % Earnings momentum
    %load crspEarnMoments91To96 EMu ESd ESkew allStksEps;   % Nstks x 72
    %idx = ismember(matchStks,allStksEps); idx2 = ismember(allStksEps,matchStks); matchStks =
matchStks(idx,1);
    %Mu = Mu(:,idx); Sd = Sd(:,idx); Skew = Skew(:,idx); aip = aip(idx,:);
    %MOM1 = MOM1(:,idx); MOM2 = MOM2(:,idx); MOM3 = MOM3(:,idx); SZ = SZ(:,idx); TURN =
TURN(:,idx);
    %P = P(:,idx); BM = BM(:,idx); B = B(:,idx); IR = IR(:,idx);
    %KURT = KURT(:,idx); SDP = SDP(:,idx); SDN = SDN(:,idx);
    %EMu = EMu(idx2,:)'; ESd = ESd(idx2,:)'; ESkew = ESkew(idx2,:)';

    % SP500 Dummy
    SP500DUMMY = zeros(Nmatches,1);
    load SP500Comp1990To2003 sp500comp;   % 500 x 14 (1990 to 2003)
    sp500 = sp500comp(:,2:7); sp500 = unique(sp500);
    idx = ismember(matchStks,sp500); SP500DUMMY(idx,1) = 1;

    % Dividend yields
    % Div Paying Stock Dummy
    load DividendYld divpr; divPayStks=unique(divpr(:,1));
    DYDUMMY = zeros(Nmatches,1);
```

```
    idx = ismember(matchStks,divPayStks); DYDUMMY(idx,1) = 1;

    DYY = zeros(6,Nmatches);
    % Dividend yield
    load crspDivPaymentsBIG DY allDivStks;   % 25 x Nstks (1979 to 2003)
    idx = ismember(matchStks,allDivStks); idx2 = ismember(allDivStks,matchStks);
    DYY(:,idx) = DY(12:17,idx2);   % 1990-1995
    DYY = nanmean(DYY)';
end;



if runNum == 1
    W1 = [];
else
    load(inoutFile,'W1');
end;

% Generate a matrix with the stock numbers
NumStksVector = [];
for i = 1:71
    NumStksVector = [NumStksVector; [1:Nmatches]'];
end;

Y = []; X = []; N = []; %FE = zeros(626149,70); strtRow = 1;
yrNum = 1;
for i = 1:71
    % Actual weight
    pos = aip(:,i); w = pos/nansum(pos);

    % Expected weight
    ew = [SZ(i,:)/nansum(SZ(i,:))]';

    if runNum == 1
        y = 100*(w-ew)./ew;
        %y = w;
        W1 = [W1 y];
    else
        if useDiff == 1
            y = W1(:,i) - (w-ew)./ew;
        else
            y = (w-ew)./ew;
        end;
    end;

    if mod(i,12) == 0
        yrNum = yrNum + 1; % div yld (matrix contains info for the prev year)
    end;

    % Get the moments and risk measures
    %x = [Mu(i,:)' Sd(i,:)' IR(i,:)' Skew(i,:)' KURT(i,:)'];
    %x = [Mu(i,:)' IR(i,:)' Skew(i,:)' KURT(i,:)' P(i,:)' EMu(i,:)' ESd(i,:)' ESkew(i,:)'];
    x = [Mu(i,:)' IR(i,:)' Skew(i,:)' KURT(i,:)' P(i,:)'];

    % Control vars
    %cv = [B(i,:)' log(SZ(i,:)') BM(i,:)' MOM1(i,:)' MOM2(i,:)' MOM3(i,:)' LB(i,:)'];
    %cv = [B(i,:)' log(SZ(i,:)') TURN(i,:)' BM(i,:)' MOM1(i,:)' MOM3(i,:)' SP500DUMMY];
    cv = [B(i,:)' log(SZ(i,:)') BM(i,:)' MOM1(i,:)' MOM3(i,:)' SP500DUMMY TURN(i,:)' DYDUMMY];
    x = [x cv];
    %x = [x DYY];

    X = [X; x]; Y = [Y; y]; Nx = max(size(x)); %[Nfe,c] = size(FE);

    N = [N; Nx];

end;
if runNum == 1
    save(inoutFile,'W1');
end;

clear B DIV IR LB M1 M12 M3 MOM1 MOM2 MOM3 W1 aip allSkts divHH SZ Sd Skew KURT BM DY SDP SDN;
```

```
clear z w x y selHH selHH2 invStks idx idx2 P Mu IPOs9196 hdivcAllP EMu ESd ESkew DYDUMMY VW
TURN;

if minPriceFilter > 0
    [r,c] = find(X(:,5)<minPriceFilter); X(r,:) = []; Y(r,:) = [];
end;
% Generate the time period column
Tp = zeros(sum(N),1); strtRow = 1;
for i = 1:71
        lstRow = strtRow + N(i) - 1;
        Tp(strtRow:lstRow,1) = i;
        strtRow = lstRow + 1;
end;
Tp = [Tp NumStksVector]; clear NumStksVector;
%X = [X FE]; clear FE;

% X contains: (1)ExpRet (2)StdDev (3)IdioRisk (4)Skewness (5)Kurtosis
%             (6)MarketBeta (7)Log(Size) (8)B/M (9)1-month Momentum (10)12-month Momentum
%             (11)Strength Local Clientele (12)Price (13)Dividend Yield

[Nr,Nx] = size(X);
regModels = {[1:Nx]}; [tmp,Nmod] = size(regModels);

%X(:,[13 18 20 21 22 24]) = [];
%X(:,[13 14 19 21 22 23]) = [];
% Panel regression
%x = fndTrimDistrib(X,1); y = fndTrimDistrib(Y,1);
x = fndTruncateDistrib(X,0.5); y = fndTruncateDistrib(Y,0.5);
%wtPctl = prctile(Y,10); [r,c] = find(Y<=wtPctl);
%X(r,:) = []; Y(r,:) = [];
%x = X; y = Y;
%x = X; y = Y;
xx = fndStd(x); yy = fndStd(y);
[b,r2,rbar2,tval,fval,dw,n,pval] = fndRegressPanel(yy,xx,Tp);
fndPrintRegRes(b,r2,rbar2,tval,fval,dw,n,pval);
return;

RES = zeros(2*11+3,5);
for i = 1:Nmod
    [b,r2,rbar2,tval,fval,dw,n,pval] = fndRegress(yy,xx(:,regModels{i}),0);
fndPrintRegRes(b,r2,rbar2,tval,fval,dw,n,pval);
    %vif = fndMultiCollTests(xx); vif

    if i == 1
        rowNum = 1;
        for j = 1:2:8
            RES(j,3) = b(rowNum); RES(j+1,3) = tval(rowNum);
            rowNum = rowNum + 1;
        end;
        RES(24:25,3) = [n; r2*100];
    elseif i == 2
        rowNum = 1;
        for j = 9:2:22
            RES(j,4) = b(rowNum); RES(j+1,4) = tval(rowNum);
            rowNum = rowNum + 1;
        end;
        RES(24:25,4) = [n; r2*100];
    else
        rowNum = 1;
        for j = 1:2:22
            RES(j,5) = b(rowNum); RES(j+1,5) = tval(rowNum);
            rowNum = rowNum + 1;
        end;
        RES(24:25,5) = [n; r2*100];
    end;
end;
RES
CreateLatexTable3dec(RES);


function [cs, ct, as, tot] = fndDgtwPerfDecomp(w,ListStocks);
```

```
%
% fndDgtwPerfDecomp.m
% Performance decomposition using the DGTW (1997) methodology
%
% Assume: w is Nstocks x Nmonths (one COLUMN for each month)
%
% AK, 7/1/07

LagMonth = 12;
StartYear = 1991; LastYear = 1996;
Nlist = max(size(ListStocks));

% Get raw returns data
load PriceRetSharesOut62To04 R retStks;
% R: 25764 x 510 (retStks: PermNum, ShareCode, Exchange, 510 Months: July
% 1962 to Dec 2004.
R = R(:,7+12*12:510);   % Jan 1975 to Dec 2004 (360 months)

% Char Benchmark Returns
load C:\1ut\3database\2processed\DGTW75To05 DGTW allStks;
dgtwStocks = allStks; DGTW = DGTW*100; DGTW = DGTW(1:360,:)';
Nstks = max(size(dgtwStocks));

% MATCH
idx = ismember(retStks,dgtwStocks); idx2 = ismember(dgtwStocks,retStks); AllStocks =
retStks(idx,1);
R = R(idx,:); DGTW = DGTW(idx2,:);
% 15578 x 360

% Match with input list
idx = ismember(AllStocks,ListStocks); idx2 = ismember(ListStocks,AllStocks);
AllStocks = AllStocks(idx,1); Nmatch = sum(idx);
R = R(idx,:); DGTW = DGTW(idx,:); w = w(idx2,:);

% Renormalize weights
sumw = nansum(w); sumw = ones(Nmatch,1)*sumw; w = w./sumw; clear sumw;

% Choose appropriate time period
% t-1 is current
StartMonth = (StartYear - 1975)*12 + 1; LastMonth = (LastYear - 1975 + 1)*12-1;
Rcur = R(:,StartMonth:LastMonth); Rfwd = R(:,StartMonth+1:LastMonth+1);  %Rlag12 =
R(:,StartMonth-12:LastMonth-12);
DGTWcur = DGTW(:,StartMonth:LastMonth); DGTWlag12 = DGTW(:,StartMonth-LagMonth:LastMonth-
LagMonth);
wlag12 = [NaN*ones(Nmatch,12) w(:,1:59)];

% CS Measure
cs = w .* (Rfwd - DGTWcur); cs = nansum(cs); cs = nanmean(cs);

% CT measure
ct = w.*Rcur - wlag12.*DGTWlag12; ct = nansum(ct); ct = nanmean(ct);

% AS measure
as = wlag12.*DGTWlag12; as = nansum(as); as = nanmean(as);

tot = cs + ct + as;

%R(1:10,1:10), DGTW(1:10,1:10), w(1:10,1:10)

%
% OldDgtwPerfDecompEachHH.m
% Perf decomposition for each household
%
% AK, 7/14/07

clear all;

LagMonth = 12;
StartYear = 1991; LastYear = 1996;

OutFile = 'E:\1cornell\2research\1behfin\27OldInv\2data\DgtwPerfDecompEachHH.mat';
```

```
% Raw Returns
load invMonPortfRetBIG allHH;   % 55277 x 72
AllHHWithPerf = allHH; Nhh = max(size(AllHHWithPerf));

load invPortfWtsHHBig;   % returns allHH, allStks, w1, ..., w71: 77995 x 9673 matrix
Nstocks = max(size(allStks));
idx = ismember(allHH,AllHHWithPerf); [RowsValidHH,c] = find(idx==1);

% Get raw returns data
load PriceRetSharesOut62To04 R retStks;
% R: 25764 x 510 (retStks: PermNum, ShareCode, Exchange, 510 Months: July
% 1962 to Dec 2004.
R = R(:,7+12*12:510);   % Jan 1975 to Dec 2004 (360 months)

% Char Benchmark Returns
load e:\1ut\3database\2processed\DGTW75To05 DGTW allStks;
dgtwStocks = allStks; DGTW = DGTW*100; DGTW = DGTW(1:360,:)';
Nstks = max(size(dgtwStocks));

% MATCH
idx = ismember(retStks,dgtwStocks); idx2 = ismember(dgtwStocks,retStks); AllStocks =
retStks(idx,1);
R = R(idx,:); DGTW = DGTW(idx2,:);
% 15578 x 360

DGTWHH = NaN*zeros(Nhh,4);
for i = 1:Nhh
    try
        whh = zeros(Nstocks,71);
        for j = 1:71
            evalStr = sprintf('w = w%d;',j); eval(evalStr);
            % w contains: Nhh x Nstks
            whh(:,j) = w(RowsValidHH(i,1),:)';
        end;
        %whh(isnan(whh)) = 0;
        whh2 = whh; whh2(whh2>0) = 1; StksInHH = nansum(whh2,2);
        [r,c] = find(StksInHH>0); whh = whh(r,:); ListStocks = allStks(r,1);
        if ~isempty(ListStocks)
            %[cs, ct, as, tot] = fndDgtwPerfDecomp(whh,ListStocks); [cs ct as tot]

            % Match with input list
            [Nlist,c] = size(ListStocks);
            idx = ismember(AllStocks,ListStocks); idx2 = ismember(ListStocks,AllStocks);
            %AllStocks2 = AllStocks(idx,1);
            Nmatch = sum(idx);
            R2 = R(idx,:); DGTW2 = DGTW(idx,:); whh = whh(idx2,:);

            % Renormalize weights
            sumw = nansum(whh); sumw = ones(Nmatch,1)*sumw; whh = whh./sumw;

            % Choose appropriate time period
            % t-1 is current
            StartMonth = (StartYear - 1975)*12 + 1; LastMonth = (LastYear - 1975 + 1)*12-1;
            Rcur = R2(:,StartMonth:LastMonth); Rfwd = R2(:,StartMonth+1:LastMonth+1);  %Rlag12 =
R(:,StartMonth-12:LastMonth-12);
            DGTWcur = DGTW2(:,StartMonth:LastMonth); DGTWlag12 = DGTW2(:,StartMonth-
LagMonth:LastMonth-LagMonth);
            wlag12 = [NaN*ones(Nmatch,12) whh(:,1:59)];

            % CS Measure
            cs = whh .* (Rfwd - DGTWcur); cs = nansum(cs); cs = nanmean(cs);

            % CT measure
            ct = whh.*Rcur - wlag12.*DGTWlag12; ct = nansum(ct); ct = nanmean(ct);

            % AS measure
            as = wlag12.*DGTWlag12; as = nansum(as); as = nanmean(as);

            tot = cs + ct + as; %[cs ct as tot]
```

```
            if as ~= 0 & ct ~= 0 & cs ~= 0
                DGTWHH(i,:) = [cs ct as tot]; %[cs ct as tot], DGTWHH(i,:), fndPause;
            end;
        end;
    end;
    if mod(i,100) == 0
        i
        save(OutFile,'DGTWHH','AllHHWithPerf');
    end;
end;

save(OutFile,'DGTWHH','AllHHWithPerf');



%
% OldCogAbAndAggPortfPerf.m
%
% AK, 7/1/07

clear all;

cogAbPctl = 20; Ncogabgrps = floor(100/cogAbPctl);
UseResidualCogAb = 1;   % remove the effect of experience
MinPortfSize = 1000;


% Predicted cognitive ability
load e:\1cornell\2research\1behfin\PredictedCogAbility.mat COGABP;
% load PredictedCogAbilityNEW.mat COGABP;
%load PredictedCogAbilityWithImpute COGABP;
% HHNum, CogAb

% HHNum, CogAb
    %load hhChars hhc; hhc(hhc==-99) = NaN; hhc(hhc==-999) = NaN; COGABP = hhc(:,[1 8]);

if UseResidualCogAb == 1
    % INVESTMENT EXPER
    load hhChars EXPER;   % HHNum, InvExperience

    Z = fndCombineMatrices(COGABP,EXPER);
    y = Z(:,2); x = Z(:,3);
    [b,r2,rbar2,tval,fval,dw,Ncases,pval] = fndRegress(y,x);
%fndPrintRegRes(b,r2,rbar2,tval,fval,dw,Ncases,pval); fndPause;
    yhat = b(1) + b(2)*x(:,1);
    COGABP = [Z(:,1) (y - yhat)];
end;

% Low and high cog ab investors
[cogabgrps,cogabbkpts] = fndAssignToDeciles(COGABP,cogAbPctl);
load invPortfWtsHHBig allStks;

% Portfolio size filter
if MinPortfSize > 0
    load hhChars hhc; [r,c] = find(hhc(:,2)>=MinPortfSize); PortfSizeHH = hhc(r,1);
end;

CS = []; CT = []; AS = []; TOTAL = [];
for i = 1:Ncogabgrps
    SelHH = cogabgrps{i};

    % Portfolio size filter
    if MinPortfSize > 0
        idx = ismember(SelHH,PortfSizeHH); SelHH = SelHH(idx,1);
    end;
    w = invGetTotalPortfWt([],SelHH);   % 9673 x 71
    [cs, ct, as, tot] = fndDgtwPerfDecomp(w,allStks);
    CS = [CS; cs]; CT = [CT; ct]; AS = [AS; as]; TOTAL = [TOTAL; tot];
    %[CS CT AS TOTAL]
end;
```

```
PERF = 12*[CS CT AS TOTAL]; PERF
DiffPerf = (PERF(Ncogabgrps,:) - PERF(1,:)); DiffPerf

clf; bar(PERF(:,1:2)); grid on;


%
% oldAgeAndPortfChars.m
% To examine the life-cycle hypothesis.
% AK, 6/23/05

minMonths = 24;
oldAgeCutoff = 60;
agePctl = 20; Nagegrps = floor(100/agePctl);
considerOldOnly = 1;
categ = 9; Ngrps = 3;
% (1)Size (2)BM (3)Price (4)IO (5)OneMonthMom (6)3MonMom (7)SixMonMom (8)9MonMom (9)12MonthMom
% (10)DivNonDiv (11)DY (12)TotalRisk (13)SystematicRisk (Mkt Beta) (14)Idiosyncratic Risk
% (15)ADIV

% DOMESTIC FF Estimates
% Fama-French estimates (portfolio risk measures)
load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
[r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
FACTORS = alph(r,[1 3:7]);   % HHNum, AlphaDom, 4FactorExposures


% Demographic vars
% Portfolio chars + Trading chars: combine them
load hhChars hhc; %hhc = hhc5min; [r,Nhhc] = size(hhc);
% hhc(16): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive, (10)PurchTurn, (11)SalesTurn, (12) MFOwnerProp
hhc(hhc==-999) = NaN;
AGE = [hhc(:,1) hhc(:,8)];

% Old investors
[r,c] = find(AGE(:,2)>=oldAgeCutoff); Nold = max(size(r));
if considerOldOnly == 1
    AGE = AGE(r,:);
end;


[agegrps,agebkpts] = fndAssignToDeciles(AGE,agePctl); agebkpts

z = []; N = []; STY = [];
for i = 1:Nagegrps
    idx = ismember(FACTORS(:,1),agegrps{i}); Nmatch = sum(idx);

    z = [z; nanmean(FACTORS(idx,2:6))]; N = [N; Nmatch];

    [sty,A] = styStylePrefGroupHH(agegrps{i},Ngrps,categ); STY = [STY; sty];
end;
z,N, STY
[styExp,A] = styExpectedStylePref(Ngrps,categ); styExp


%
% oldAgeAndRiskExposures.m
%
% AK, 6/30/05

minMonths = 24;
agePctl = 20; Nagegrps = floor(100/agePctl);
oldAgeCutoff = 60; considerOldOnly = 1;


load hhChars hhc;
% hhc(9): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive
```

```
hhc(hhc==-999)=NaN; AGE = hhc(:,[1 8]);

% Old investors
[r,c] = find(AGE(:,2)>=oldAgeCutoff); Nold = max(size(r));
if considerOldOnly == 1
    AGE = AGE(r,:); hhc = hhc(r,:);
end;


% DOMESTIC FF Estimates
% Fama-French estimates (portfolio risk measures)
load invFamaFrenchEachHH alph;  % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
[r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
FACTEXPO = alph(r,[1 4:7]);   % HHNum, AlphaDom


% Idiosyncratic risk
load invIdioSyncRiskEachHH.mat IR;
% IR contains: HHNum, Nmonths, IRusingCAPM, IRusingFF, TotalRisk
IR = [IR(:,[1 5]) 100*(IR(:,[4])./IR(:,[5]) 100*(IR(:,[5])-IR(:,[4])./IR(:,[5])];


[agegrps,agebkpts] = fndAssignToDeciles(AGE,agePctl); agebkpts

z = []; N = [];
for i = 1:Nagegrps
    idx1 = ismember(FACTEXPO(:,1),agegrps{i}); Nmatch1 = sum(idx1);
    idx2 = ismember(IR(:,1),agegrps{i}); Nmatch2 = sum(idx2);

    z = [z; [nanmean(IR(idx2,2:4)) nanmean(FACTEXPO(idx1,2:5))]]; N = [N; [Nmatch1 Nmatch2]];
end;
z

if considerOldOnly == 0
    A = [zeros(Nagegrps,2) z];
else
    [r,Na] = size(A); A = [A; zeros(2,Na); [zeros(Nagegrps,2) z]];
    CreateLatexTable3dec(A);
end;



%
% oldPropLosersRealizedDec.m
% Measure tax loss selling in Dec
%
% AK, 7/21/05

% FILES
datadir = 'E:\1Cornell\3Database\2Processed\';
% Files for winners/losers data
inFile =  [datadir 'invBSPair_ActSell_ALL.mat'];
inFile2 = [datadir 'invBSPair_PotSell_ALL.mat'];

% Get December dates
DecDates = [];
dtNum1 = datenum(1991,11,1); dtNum2 = datenum(1991,12,31); DecDates = [DecDates;
[dtNum1:dtNum2]];
dtNum1 = datenum(1992,11,1); dtNum2 = datenum(1992,12,31); DecDates = [DecDates;
[dtNum1:dtNum2]];
dtNum1 = datenum(1993,11,1); dtNum2 = datenum(1993,12,31); DecDates = [DecDates;
[dtNum1:dtNum2]];
dtNum1 = datenum(1994,11,1); dtNum2 = datenum(1994,12,31); DecDates = [DecDates;
[dtNum1:dtNum2]];
dtNum1 = datenum(1995,11,1); dtNum2 = datenum(1995,12,31); DecDates = [DecDates;
[dtNum1:dtNum2]];

% LOAD DATA
% Actual winners/losers
% First trade date: 3-Jan-91, Last trade date: 28-Nov-96 => 71 months
load(inFile,'bspr');
```

```
% bspr contains:
%     (1)HHNum, (2)Accnum, (3)PermNum, (4)Qty, (5)SellDatenum, (6)SellPrice, (7)BuyDatenum,
(8)BuyQty, (9)BuyPrice,
%     (10)SizeInfo (11) DysHeld: TradingDays
bsprA = bspr; clear bspr;
idx = ismember(bsprA(:,5),DecDates); bsprA = bsprA(idx,:); Nactdectrds = sum(idx);
fprintf(1, '%d actual Dec trades.\n', Nactdectrds);
% Identify losers
PrDiff = bsprA(:,6) - bsprA(:,9);
[r,c] = find(PrDiff<0); bsprA = bsprA(r,:); Nactdectrds = max(size(bsprA));
fprintf(1, '%d losses realized in Dec.\n', Nactdectrds);

% Paper winners/losers
load(inFile2,'bspr'); [tmp,Ncols] = size(bspr);
% returns bspr with 8 cols: (1)HHNum (2)PermNum (3)Qty (4)SellDatenum (5)BuyDatenum (6)SellPrice
(7)BuyPrice
%                           (8)SizeInfo  (9) DysHeld: TradingDays
idx = ismember(bspr(:,5),DecDates); bspr = bspr(idx,:); Npapdectrds = sum(idx);
fprintf(1, '%d paper Dec trades.\n', Npapdectrds);
% Identify losers
PrDiff = bspr(:,6) - bspr(:,9);
[r,c] = find(PrDiff<0); bspr = bspr(r,:); Npapdectrds = max(size(bspr));
fprintf(1, '%d paper losses in Dec.\n', Npapdectrds);

allHH = unique(bsprA(:,1)); Nhh = max(size(allHH));
fprintf(1, '%d HH trade in Dec.\n', Nhh);

TaxSkill = zeros(Nhh,3); TaxSkill(:,1) = allHH;
for i = 1:Nhh
    [r,c] = find(bsprA(:,1)==allHH(i,1)); [Nact,c] = size(r);
    [r,c] = find(bspr(:,1)==allHH(i,1)); [Npap,c] = size(r);

    TaxSkill(i,2:3) = [Nact Npap];
    if mod(i,250) == 0
        i
    end;
end;

save e:\1cornell\3database\2processed\invNovDecTaxLossSelling.mat TaxSkill;


%
% oldCorrelSkillMeasures.m
%
% AK, 7/21/05

alphaCol = 3; minMonths = 12; useCAPMAlpha = 0;

% PORTFOLIO PERFORMANCE
% Get performance data
load invPortfPerfAllHH ipc;   % returns ipc
% ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
%               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep (ex
post Sharpe ratio);
%               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta, (23)
Tval: FFAlpha,
%               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28) Rbarsq:
CAPM Reg,
%               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
PERF1 = ipc(:,[1 16]);


% DOMESTIC FF Estimates
% Fama-French estimates (portfolio risk measures)
load invFamaFrenchEachHH alph;   % Nhh x 12: % HHNum, Ncases, b0,... b4, t0, ..., t4
[r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
PERF2 = alph(r,[1 3]);   % HHNum, AlphaDom

load invFamaFrenchEachHH capmAlph; alph = capmAlph;

[r,c] = find(alph(:,2)>=minMonths); Nr = max(size(r));
```

```
PERF3 = alph(r,[1 3]);   % HHNum, AlphaDom



% Informed in domestic setting?
minTrades = 3; locCutoff = 250;
winSize = 10;   % (1)5 (2)10 (3)21 (4)42 (5)63 (6)84 (7)105 (8)126 (9)252
infoFile = ['E:\1cornell\2research\1behfin\51local\2data\OverConfiEachHH_Win' num2str(winSize)
'_Dist' num2str(locCutoff) '.mat'];
load(infoFile,'OC','NTR','allHH');
%OC = 100*OC(:,:,ocCol)); % OC = NaN*zeros(Nhh,6,9); NTR = NaN*zeros(Nhh,6);
% OC: ExPostBuyLoc, ExPostSellLoc, ExPostBuyNonLoc, ExPostSellNonLoc, ExPostBuyAll, ExPostSellAll
for i = 1:6
    [r,c] = find(NTR(:,i)<minTrades); OC(r,i)=NaN;
end;

% Get info measure
info = [allHH OC(:,1)-OC(:,2) OC(:,3)-OC(:,4) OC(:,5)-OC(:,6)];
% HHNum, InfoMeasureLoc, InfoMeasureNonLoc, InfoMeasureAll
PERF4 = info(:,[1 4]);


% PORTFOLIO PERFORMANCE
% Get performance data
load invPortfPerfAllHH ipc;   % returns ipc
% ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
%               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep (ex
post Sharpe ratio);
%               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta, (23)
Tval: FFAlpha,
%               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28) Rbarsq:
CAPM Reg,
%               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
PERF5 = ipc(:,[1 8]);


% Market timing
load invMktTimingEachHH mtm allHH;
% (1)rsr (2)esr (3)jenalph (4)tJenAlph (5)gh(1) (6)Tgh1 (7)gh2 (8)Tgh2 (9)beta2HM (10)tBeta2HM
% (11)beta2TM (12)tBeta2TM (13)beta (14)ffalph (15)tFFAlph
PERF6 = [allHH mtm(:,[5 7 9 11])];


x = fndCombineMatrices(PERF1,PERF2);
x = fndCombineMatrices(x,PERF3);
x = fndCombineMatrices(x,PERF4);
x = fndCombineMatrices(x,PERF5);
x = fndCombineMatrices(x,PERF6);
[r,Nx] = size(x);
% (1)SR (2)Alpha (3)CAPMAlpha (4)PTBSD (5)MeanRet (6)MktTiming1 (7)MktTiming2

x = x(:,2:Nx);
fndCorrCoef(x)


%
% oldSummaryStats2.m
%
% AK, 4/22/05

oldAgeCutoff = 60; considerOldOnly = 0; considerRetiredOnly = 0;
agePctl = 20; Nagegrps = floor(100/agePctl);

% Demographic vars
% Portfolio chars + Trading chars: combine them
load hhChars hhc5min; hhc = hhc5min; [r,Nhhc] = size(hhc);
% hhc(16): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive, (10)PurchTurn, (11)SalesTurn, (12) MFOwnerProp
hhc(hhc==-999) = NaN;
Nhh = max(size(hhc)); Nhh
```

```
turn = nanmean(hhc(:,10:11)')'; hhc(:,11) = turn;

% Retired investors
[r,c] = find(hhc(:,6)==8); Nretired = max(size(r));
if considerRetiredOnly == 1
    hhc = hhc(r,:);
end;
[r,c] = find(isnan(hhc(:,6))==0); Nvalid = max(size(r)); Nvalid
fprintf(1, '%d investors (%.2f percent OR %.2f percent) are retired.\n', Nretired,
100*Nretired/Nhh, 100*Nretired/Nvalid);

% Old investors
[r,c] = find(hhc(:,8)>=oldAgeCutoff); Nold = max(size(r));
if considerOldOnly == 1
    hhc = hhc(r,:);
    [r,c] = find(isnan(hhc(:,8))==0); Nvalid = max(size(r)); Nvalid
end;
fprintf(1, '%d investors (%.2f percent OR %.2f percent) are above %d years.\n', Nold,
100*Nold/Nhh, 100*Nold/Nvalid, oldAgeCutoff);


% AGE and INCOME
% Convert income codes into mid-points of income range
origIncCode = hhc(:,6);
icd = hhc(:,6);
icd(icd == 1) = 7.5;
icd(icd == 2) = 17.5;
icd(icd == 3) = 25;
icd(icd == 4) = 35;
icd(icd == 5) = 45;
icd(icd == 6) = 62.5;
icd(icd == 7) = 87.5;
icd(icd == 8) = 112.5;
icd(icd == 9) = 250;    % ref: Ivkovich:Weisbenner
hhc(:,6) = icd*1000;
hhc = [hhc origIncCode];

% Diversification
load hhChars hdivcAllP;   % 41039 x 6
DIV = hdivcAllP(:,3);
% hdivcAllP contains: HHNum, Nobs, Nstks, Sqw, NormVar, AvgCorrel
divHH = hdivcAllP(:,1); Ndivhh = max(size(hdivcAllP));
fprintf(1, '%d HH with DIV data.\n', Ndivhh);

% INVESTMENT EXPER
load hhChars EXPER;   % HHNum, InvExperience


load TaxRateIncomeNetWorth2.txt; x = TaxRateIncomeNetWorth2; clear TaxRateIncomeNetWorth2;
% x contains: HHNum, TaxRate, Income, NetWorth
% Clean data
[r,c] = find(x(:,4)>0); netWorth = x(r,[1 4]);
netWorth(:,2)=fndTruncateDistrib(netWorth(:,2),0.5);

hhc = fndCombineMatrices(hhc,netWorth);

[agegrps,agebkpts] = fndAssignToDeciles(hhc(:,[1 8]),agePctl); agebkpts

agebkpts = [
    20 34;
    35 44;
    45 54;
    55 64;
    65 74;
    75 100;
];

agebkpts = [[20 24]; [25 29]; [30 34]; [35 39]; [40 44]; [45 49]; [50 54]; [55 59]; [60 64]; [65
69]; [70 74]; [75 100];];

Nagegrps = max(size(agebkpts));
```

```
% Portf size to income/wealth ratio
z = []; Nstks = []; Exper = [];
for i = 1:Nagegrps
    [r,c] = find(hhc(:,8) >= agebkpts(i,1) & hhc(:,8) <= agebkpts(i,2)); max(size(r))
    selHH = hhc(r,1);
    %idx = ismember(hhc(:,1),agegrps{i}); [r,c] = find(idx==1);
    z = [z; [nanmean(hhc(r,2)) nanmean(hhc(r,6))  nanmean(hhc(r,2)./hhc(r,6))  ...
                nanmean(hhc(r,11)) nanmean(hhc(r,12))]];

    idx = ismember(divHH,selHH); Nstks = [Nstks; nanmean(DIV(idx,1))];

    idx = ismember(EXPER(:,1),selHH); Exper = [Exper; nanmedian(EXPER(idx,2))/365];
end;
bar(z(:,5)); grid on;
Nstks, Exper


%
% oldAgeAndBehBiases.m
%
% AK, 6/25/05

agePctl = 10; Nagegrps = floor(100/agePctl);
oldAgeCutoff = 60; considerOldOnly = 0;

load hhChars hhc;
% hhc(9): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive
hhc(hhc==-999)=NaN; AGE = hhc(:,[1 8]); AGE(:,2) = AGE(:,2);

% Old investors
[r,c] = find(AGE(:,2)>=oldAgeCutoff); Nold = max(size(r));
if considerOldOnly == 1
    AGE = AGE(r,:);
end;


% LOCAL BIAS
load LocalBias LB; % HHNum, LBOld, LBNew
LB = LB(:,[1 3]);

% DE
%load DispEffectEachHHNew Z; % z = [allHH(i,1) Nxx Nyy (4)propAct (5)propExp];
%DE = [Z(:,1) 100*(Z(:,5)-Z(:,4))./Z(:,5)]; DE(isinf(DE)) = NaN;
load AdjustedDispEffect ADE;

load TradeClustering ATC;   % HHNum, ATC

% Diversification
load hhChars hdivcAllP;   % 41039 x 6
DIV = hdivcAllP(:,[1 3 5 6]);
% hdivcAllP contains: (1)HHNum (2)Nobs (3)Nstks (4)Sqw (5)NormVar
% (6)AvgCorrel

% PORTFOLIO DIVIDEND YIELD
load invMonPortfDivYldBIG DY allHH;
dy = DY(:,1:72); DY = [allHH 4*nanmean(100*dy')'];   % Annual dividend yield


% OVERCONFIDENCE
% Dummy = 1 if in the top turnover quintile and bottom performance quintile
% Get performance data
load invPortfPerfAllHH ipc;   % returns ipc
% ipc contains: (1) HHNum (2)Nvalid (3)annRet (4)annRetM (5)annRetRF (6)alph (7)beta (8)mnRet
%               (9)sd (10)mnNetRet (11) sdNet (12)mnRetM (13)sdM (14)mnRetRF (15)sdRF (16)Sep (ex
post Sharpe ratio);
%               (17) FFAlpha (18) RMRF (19) SMB (20) HML (21) Tval: Alpha, (22) Tval: Beta, (23)
Tval: FFAlpha,
%               (24) Tval: RMRF, (25) Tval: SMB, (26) Tval: HML, (27) Rsq: CAPM Reg, (28) Rbarsq:
CAPM Reg,
```

```
%               (29) Rsq: FF Reg, (30) Rbarsq: FF Reg
% Demographics
% Cross-sectional regression
% Portfolio chars + Trading chars: combine them
load hhChars htc hhgrps10 hhc5min; hhc = hhc5min; hhgrps = hhgrps10; clear hhc5min hhgrps10;
hhc = sortrows(hhc,1); htc = sortrows(htc,1);
hhc = [hhc htc(:,2:6)];
% htc contains: (1)HHNum (2)Nstks (3)NbuyTrades (4)NsellTrades (5)NdaysActive (6)dysBetTrades
% hhgrps(7): HHNum, Ntr(buys), Ntr(sells), Groups for BUYS (2: 21d, 63d), Groups for SELLS (2:
21d, 63d)
% hhc(16): HHNum, (2)AvgPortfVal, (3)MaxPortfVal, AvgTradeSize, (5)NumTrades, (6)IncomeCode,
(7)JobCode, (8)Age,
%         (9)NmonthsActive, (10)PurchTurn, (11)SalesTurn, (12) MFOwnerProp
%         (13)Nstks (14)NbuyTrades (15)NsellTrades (16)NdaysActive
%         (17)dysBetTrades (18) PortfSize/Income ratio
hhc(hhc==-99)=NaN; hhc(hhc==-999)=NaN;
turn = nanmean(hhc(:,[10 11])')';  TURN = [hhc(:,1) turn];
Nhh = max(size(ipc)); OCProxy = zeros(Nhh,2); OCProxy(:,1) = ipc(:,1);
[perfgrps,perfbkpts] = fndAssignToDeciles(ipc(:,[1 16]),33); lowperfhh = perfgrps{1}; highperfhh
= perfgrps{3};
[turngrps,perfbkpts] = fndAssignToDeciles(TURN,33); lowturnhh = turngrps{1}; highturnhh =
turngrps{3};
idx = ismember(lowperfhh,highturnhh); selhh = lowperfhh(idx,1); Nochh = max(size(selhh));
idx = ismember(OCProxy(:,1),selhh); OCProxy(idx,2) = 1;   % Low Perf + High Turn
OVERCONFI = OCProxy;   % HHNum, OC


[agegrps,agebkpts] = fndAssignToDeciles(AGE,agePctl); agebkpts

z = []; N = [];
for i = 1:Nagegrps
    idx = ismember(LB(:,1),agegrps{i}); idx2 = ismember(ADE(:,1),agegrps{i});
    idx3 = ismember(DY(:,1),agegrps{i});
    idx4 = ismember(OVERCONFI(:,1),agegrps{i}); Noc = sum(OVERCONFI(idx4,2)); Nhh =
max(size(agegrps{i})); propOC = Noc/Nhh;
    idx5 = ismember(DIV(:,1),agegrps{i});

    z = [z; [nanmean(LB(idx,2)) nanmean(ADE(idx2,2)) nanmean(DY(idx3,2)) propOC
nanmedian(DIV(idx5,2:4))]];
end;
z
```