2019 Full States EV and EC of One Check

This is the example vignette for function: **snw_evuvw20_jaeemk** from the **PrjOptiSNW Package.** 2019 integrated over VU and VW, given optimal savings choices, unemployment shocks and various expectations.

Test SNW_EVUVW19_JAEEMK Defaults Dense

VFI and Distribution

Call the function with defaults.

```
clear all;
st_solu_type = 'bisec_vec';

% Solve the VFI Problem and get Value Function
mp_params = snw_mp_param('default_docdense');
mp_controls = snw_mp_control('default_test');

% set Unemployment Related Variables
xi=0.5; % Proportional reduction in income due to unemployment (xi=0 refers to 0 labor income;
b=0; % Unemployment insurance replacement rate (b=0 refers to no UI benefits; b=1 refers to 100
TR=100/58056; % Value of a welfare check (can receive multiple checks). TO DO: Update with alte

mp_params('xi') = xi;
mp_params('b') = b;
mp_params('TR') = TR;

% Solve for Unemployment Values
mp_controls('bl_print_vfi') = false;
mp_controls('bl_print_ds') = false;
mp_controls('bl_print_ds_verbose') = false;
mp_controls('bl_print_precompute') = false;
mp_controls('bl_print_precompute_verbose') = false;
mp_controls('bl_print_a4chk') = false;
mp_controls('bl_print_a4chk_verbose') = false;
mp_controls('bl_print_evuvw20_jaeemk') = false;
mp_controls('bl_print_evuvw20_jaeemk_verbose') = false;

% Solve the Model to get V working and unemployed
[V_ss,ap_ss,cons_ss,mp_valpol_more_ss] = snw_vfi_main_bisec_vec(mp_params, mp_controls);
```

```
Completed SNW_VFI_MAIN_BISEC_VEC;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;time=16.9824
```

```
inc_VFI = mp_valpol_more_ss('inc_VFI');
spouse_inc_VFI = mp_valpol_more_ss('spouse_inc_VFI');
total_inc_VFI = inc_VFI + spouse_inc_VFI;
% Solve unemployment
[V_unemp,~,cons_unemp,~] = snw_vfi_main_bisec_vec(mp_params, mp_controls, V_ss);
```

```
Completed SNW_VFI_MAIN_BISEC_VEC 1 Period Unemp Shock;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;time=18
```

```
[Phi_true] = snw_ds_main(mp_params, mp_controls, ap_ss, cons_ss, mp_valpol_more_ss);
```

```
Completed SNW_DS_MAIN;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;time=61.7368
```

```
% Get Matrixes
cl_st_precompute_list = {'a', ...
    'inc', 'inc_unemp', 'spouse_inc', 'spouse_inc_unemp', 'ref_earn_wageind_grid',...
    'ap_idx_lower_ss', 'ap_idx_higher_ss', 'ap_idx_lower_weight_ss'};
mp_controls('bl_print_precompute_verbose') = false;
[mp_precompute_res] = snw_hh_precompute(mp_params, mp_controls, cl_st_precompute_list, ap_ss, P
```

```
Wage quintile cutoffs=0.49295     0.79302      1.3138      2.1063
Completed SNW_HH_PRECOMPUTE;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;time cost=35.3213
```

Solve for 2019 Evuvw With 0 and 2 Checks

```
% Call Function
welf_checks = 0;
[ev19_jaeemk_check0, ec19_jaeemk_check0, ev20_jaeemk_check0, ec20_jaeemk_check0] = snw_evuvw19_
    welf_checks, st_solu_type, mp_params, mp_controls, ...
    V_ss, cons_ss, V_unemp, cons_unemp, mp_precompute_res);
```

```
Completed SNW_A4CHK_UNEMP_BISEC_VEC;welf_checks=0;TR=0.0017225;xi=0.5;b=0;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=
Completed SNW_A4CHK_WRK_BISEC_VEC;welf_checks=0;TR=0.0017225;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;
Completed SNW_EVUVW20_JAEEMK;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;timeEUEC=0.67039
Completed SNW_EVUVW19_JAEEMK;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;time=8.1727
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_outcomes ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefvari
ec19_jaeemk	1	1	6	1.8942e+06	82	23100	8.2855e+06	4.3742	5.2363	1.1971
ec20_jaeemk	2	2	6	1.9173e+06	83	23100	9.7703e+06	5.0959	8.3753	1.6435
ev19_jaeemk	3	3	6	1.8942e+06	82	23100	-3.7288e+06	-1.9685	22.206	-11.28
ev20_jaeemk	4	4	6	1.9173e+06	83	23100	-4.1377e+06	-2.1581	22.856	-10.591

xxx TABLE:ec19_jaeemk xxxxxxxxxxxxxxxxxx

	c1	c2	c3	c4	c5	c23096	c23097	c23098	c23099	c2310
r1	0.08064	0.080744	0.082559	0.086958	0.092876	10.7	10.935	11.171	11.402	11.62
r2	0.081432	0.081432	0.084644	0.089174	0.095493	10.86	11.091	11.32	11.543	11.7
r3	0.083622	0.083622	0.085501	0.091223	0.097903	11.024	11.249	11.471	11.683	11.86
r4	0.086619	0.086619	0.088508	0.094197	0.10085	11.168	11.388	11.601	11.801	11.96
r5	0.089528	0.089528	0.091431	0.097077	0.1037	11.309	11.521	11.724	11.908	12.04
r78	0.21707	0.21707	0.21707	0.21707	0.22416	26.837	28.052	29.222	30.511	32.06
r79	0.21707	0.21707	0.21707	0.21707	0.22416	28.992	31.165	32.888	34.236	35.60
r80	0.21707	0.21707	0.21707	0.21707	0.22416	32.266	33.961	36.121	38.357	40.63
r81	0.21707	0.21707	0.21707	0.21707	0.22416	38.348	39.931	42.54	46.29	49.27
r82	0.21707	0.21707	0.21707	0.21707	0.22361	51.027	52.913	57.047	61.392	64.15

xxx TABLE:ec20_jaeemk xxxxxxxxxxxxxxxxxx

	c1	c2	c3	c4	c5	c23096	c23097	c23098	c23099	c2310
r1	0.078786	0.079375	0.080372	0.084318	0.090081	10.601	10.841	11.082	11.321	11.55
r2	0.078786	0.079674	0.081595	0.086622	0.09285	10.755	10.992	11.228	11.46	11.68
r3	0.079575	0.080463	0.084561	0.089582	0.096059	10.912	11.145	11.375	11.598	11.80
r4	0.082616	0.083505	0.087581	0.092591	0.099121	11.281	11.504	11.718	11.91	11.91
r5	0.085578	0.086467	0.090513	0.095505	0.10201	11.412	11.627	11.83	12.006	12.01
r79	0.21707	0.21796	0.22416	0.241	0.26692	34.627	36.056	37.896	40.382	42.95
r80	0.21707	0.21796	0.22416	0.241	0.26692	39.694	41.355	43.554	46.378	49.23
r81	0.21707	0.21796	0.22416	0.241	0.26692	47.978	51.293	53.451	55.657	58.48

r82	0.21707	0.21796	0.22416	0.241	0.26745	63.659	68.583	71.594	74.676	79.57
r83	0.21707	0.21796	0.22416	0.241	0.27378	112.67	119.43	126.45	133.75	141.3

xxx TABLE:ev19_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c23096**	**c23097**	**c23098**	**c23099**	**c23100**
r1	-231.16	-231.1	-230.08	-227.9	-224.17	20.989	21.13	21.27	21.409	21.549
r2	-222.14	-222.14	-221.33	-219.39	-215.84	20.973	21.11	21.246	21.382	21.519
r3	-213.6	-213.6	-213.21	-211.33	-207.93	20.953	21.086	21.219	21.352	21.486
r4	-205.67	-205.67	-205.31	-203.55	-200.36	20.925	21.055	21.185	21.316	21.443
r5	-198.45	-198.45	-198.11	-196.45	-193.44	20.891	21.018	21.145	21.273	21.389
r78	-9.9698	-9.9698	-9.9698	-9.9698	-9.824	2.4708	2.4858	2.5001	2.5134	2.526
r79	-8.9313	-8.9313	-8.9313	-8.9313	-8.7855	2.2408	2.2553	2.2677	2.278	2.2877
r80	-7.6669	-7.6669	-7.6669	-7.6669	-7.5211	1.9523	1.9632	1.9722	1.9796	1.9868
r81	-5.9967	-5.9967	-5.9967	-5.9967	-5.8508	1.5534	1.5588	1.5654	1.5715	1.5765
r82	-3.6113	-3.6113	-3.6113	-3.6113	-3.4767	0.9548	0.95648	0.95963	0.96249	0.96406

xxx TABLE:ev20_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c23096**	**c23097**	**c23098**	**c23099**	**c23100**
r1	-235.08	-234.92	-233.84	-231.32	-227.14	21.095	21.236	21.376	21.516	21.65
r2	-226.34	-226.18	-225.14	-222.7	-218.72	21.082	21.219	21.356	21.492	21.62
r3	-217.94	-217.78	-216.77	-214.45	-210.67	21.065	21.198	21.331	21.464	21.59
r4	-209.81	-209.67	-208.73	-206.55	-203.01	21.045	21.174	21.304	21.434	21.56
r5	-202.41	-202.27	-201.4	-199.35	-196.01	21.014	21.14	21.267	21.394	21.52
r79	-9.9634	-9.9447	-9.8178	-9.5061	-8.9993	2.5241	2.5372	2.5494	2.5607	2.571
r80	-8.9252	-8.9064	-8.7795	-8.4679	-7.9637	2.2926	2.3026	2.3118	2.3204	2.328
r81	-7.661	-7.6423	-7.5154	-7.2037	-6.7021	1.9992	2.0061	2.0125	2.0185	2.02
r82	-5.9911	-5.9724	-5.8455	-5.5339	-5.0346	1.5919	1.5958	1.5994	1.6027	1.605
r83	-3.6067	-3.588	-3.4611	-3.1494	-2.6526	0.97826	0.97949	0.98063	0.98169	0.9826

```
% Call Function
welf_checks = 2;
[ev19_jaeemk_check2, ec19_jaeemk_check2, ev20_jaeemk_check2, ec20_jaeemk_check2] = snw_evuvw19_
    welf_checks, st_solu_type, mp_params, mp_controls, ...
    V_ss, cons_ss, V_unemp, cons_unemp, mp_precompute_res);
```

```
Completed SNW_A4CHK_UNEMP_BISEC_VEC;welf_checks=2;TR=0.0017225;xi=0.5;b=0;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=
Completed SNW_A4CHK_WRK_BISEC_VEC;welf_checks=2;TR=0.0017225;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;
Completed SNW_EVUVW20_JAEEMK;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;timeEUEC=0.4338
Completed SNW_EVUVW19_JAEEMK;SNW_MP_PARAM=default_dense;SNW_MP_CONTROL=default_test;time=9.4255
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_outcomes ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

	i	**idx**	**ndim**	**numel**	**rowN**	**colN**	**sum**	**mean**	**std**	**coefvari**
ec19_jaeemk	1	1	6	1.8942e+06	82	23100	8.2866e+06	4.3747	5.2364	1.197
ec20_jaeemk	2	2	6	1.9173e+06	83	23100	9.7714e+06	5.0964	8.3755	1.6434
ev19_jaeemk	3	3	6	1.8942e+06	82	23100	-3.7109e+06	-1.9591	22.18	-11.321
ev20_jaeemk	4	4	6	1.9173e+06	83	23100	-4.1188e+06	-2.1482	22.829	-10.627

xxx TABLE:ec19_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c23096**	**c23097**	**c23098**	**c23099**	**c2310**
r1	0.081703	0.081806	0.0836	0.087675	0.09332	10.7	10.935	11.171	11.402	11.62
r2	0.083719	0.083719	0.085961	0.089966	0.095955	10.86	11.091	11.32	11.543	11.7
r3	0.086201	0.086201	0.087481	0.092085	0.098395	11.024	11.25	11.471	11.683	11.86
r4	0.089189	0.089189	0.090472	0.095056	0.10134	11.168	11.388	11.601	11.801	11.96
r5	0.092085	0.092085	0.093374	0.097931	0.10419	11.309	11.521	11.724	11.908	12.04

r78	0.22052	0.22052	0.22052	0.22052	0.22761	26.838	28.053	29.223	30.512	32.06
r79	0.22052	0.22052	0.22052	0.22052	0.22761	28.993	31.166	32.889	34.237	35.60
r80	0.22052	0.22052	0.22052	0.22052	0.22761	32.267	33.963	36.122	38.358	40.64
r81	0.22052	0.22052	0.22052	0.22052	0.22761	38.349	39.933	42.542	46.292	49.27
r82	0.22052	0.22052	0.22052	0.22052	0.22706	51.03	52.916	57.05	61.396	64.15

xxx TABLE:ec20_jaeemk xxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c23096**	**c23097**	**c23098**	**c23099**	**c2310**
r1	0.079785	0.079927	0.081177	0.084922	0.09046	10.601	10.841	11.082	11.321	11.55
r2	0.080464	0.080738	0.082621	0.087274	0.093237	10.755	10.992	11.228	11.46	11.68
r3	0.082148	0.082733	0.085586	0.090261	0.096428	10.913	11.145	11.375	11.598	11.80
r4	0.085181	0.085763	0.088604	0.093275	0.099485	11.281	11.504	11.718	11.91	11.91
r5	0.088131	0.088709	0.091532	0.096187	0.10238	11.412	11.627	11.83	12.006	12.01
r79	0.22052	0.2214	0.22761	0.24372	0.26885	34.627	36.057	37.898	40.384	42.95
r80	0.22052	0.2214	0.22761	0.24372	0.26914	39.694	41.356	43.555	46.38	49.2
r81	0.22052	0.2214	0.22761	0.24372	0.26933	47.98	51.294	53.452	55.659	58.4
r82	0.22052	0.2214	0.22761	0.24378	0.26982	63.662	68.584	71.596	74.678	79.57
r83	0.22052	0.2214	0.22761	0.24444	0.27722	112.67	119.43	126.45	133.75	141.3

xxx TABLE:ev19_jaeemk xxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c23096**	**c23097**	**c23098**	**c23099**	**c23100**
r1	-230.56	-230.51	-229.56	-227.46	-223.78	20.989	21.13	21.27	21.409	21.549
r2	-221.56	-221.56	-220.82	-218.96	-215.47	20.973	21.11	21.246	21.382	21.519
r3	-213.05	-213.05	-212.7	-210.91	-207.58	20.953	21.086	21.219	21.352	21.486
r4	-205.16	-205.16	-204.83	-203.16	-200.03	20.926	21.055	21.185	21.316	21.443
r5	-197.97	-197.97	-197.66	-196.08	-193.13	20.891	21.018	21.145	21.273	21.389
r78	-9.8986	-9.8986	-9.8986	-9.8986	-9.7602	2.4708	2.4859	2.5001	2.5134	2.526
r79	-8.8601	-8.8601	-8.8601	-8.8601	-8.7216	2.2408	2.2553	2.2677	2.278	2.2877
r80	-7.5957	-7.5957	-7.5957	-7.5957	-7.4572	1.9523	1.9632	1.9722	1.9796	1.9868
r81	-5.9255	-5.9255	-5.9255	-5.9255	-5.787	1.5534	1.5588	1.5654	1.5715	1.5765
r82	-3.5401	-3.5401	-3.5401	-3.5401	-3.4123	0.9548	0.95648	0.95963	0.96249	0.96406

xxx TABLE:ev20_jaeemk xxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c23096**	**c23097**	**c23098**	**c23099**	**c23100**
r1	-234.48	-234.32	-233.33	-230.88	-226.76	21.095	21.236	21.377	21.516	21.65
r2	-225.75	-225.6	-224.64	-222.29	-218.36	21.082	21.219	21.356	21.492	21.62
r3	-217.37	-217.22	-216.3	-214.05	-210.33	21.065	21.198	21.331	21.464	21.59
r4	-209.28	-209.15	-208.29	-206.18	-202.69	21.045	21.174	21.304	21.434	21.56
r5	-201.91	-201.79	-200.98	-199	-195.71	21.014	21.14	21.267	21.394	21.52
r79	-9.8923	-9.8742	-9.754	-9.4529	-8.9555	2.5241	2.5372	2.5494	2.5607	2.571
r80	-8.8541	-8.8359	-8.7157	-8.4149	-7.9203	2.2926	2.3026	2.3118	2.3204	2.328
r81	-7.5899	-7.5718	-7.4516	-7.151	-6.6591	1.9992	2.0061	2.0125	2.0185	2.02
r82	-5.9201	-5.9019	-5.7817	-5.4814	-4.9918	1.5919	1.5958	1.5994	1.6027	1.605
r83	-3.5356	-3.5175	-3.3973	-3.0972	-2.6142	0.97826	0.97949	0.98063	0.98169	0.9826

Differences between Checks in Expected Value and Expected Consumption

```
mn_V_U_gain_check = ev19_jaeemk_check2 - ev19_jaeemk_check0;
mn_MPC_U_gain_share_check = (ec19_jaeemk_check2 - ec19_jaeemk_check0)./(welf_checks*mp_params('
```

Dense Param Results Define Frames

Define the matrix dimensions names and dimension vector values. Policy and Value Functions share the same ND dimensional structure.

```
% Grids:
age_grid = 18:99;
agrid = mp_params('agrid')';
eta_H_grid = mp_params('eta_H_grid')';
eta_S_grid = mp_params('eta_S_grid')';
ar_st_eta_HS_grid = string(cellstr([num2str(eta_H_grid', 'hz=%3.2f;'), num2str(eta_S_grid', 'wz
edu_grid = [0,1];
marry_grid = [0,1];
kids_grid = (1:1:mp_params('n_kidsgrid'))';
% NaN(n_jgrid,n_agrid,n_etagrid,n_educgrid,n_marriedgrid,n_kidsgrid);
cl_mp_datasetdesc = {};
cl_mp_datasetdesc{1} = containers.Map({'name', 'labval'}, {'age', age_grid});
cl_mp_datasetdesc{2} = containers.Map({'name', 'labval'}, {'savings', agrid});
cl_mp_datasetdesc{3} = containers.Map({'name', 'labval'}, {'eta', 1:length(eta_H_grid)});
cl_mp_datasetdesc{4} = containers.Map({'name', 'labval'}, {'edu', edu_grid});
cl_mp_datasetdesc{5} = containers.Map({'name', 'labval'}, {'marry', marry_grid});
cl_mp_datasetdesc{6} = containers.Map({'name', 'labval'}, {'kids', kids_grid});
```

Analyze Difference in V and C with Check

The difference between V and V with Check, marginal utility gain given the check.

```
% Generate some Data
mp_support_graph = containers.Map('KeyType', 'char', 'ValueType', 'any');
mp_support_graph('cl_st_xtitle') = {'Savings States, a'};
mp_support_graph('st_legend_loc') = 'eastoutside';
mp_support_graph('bl_graph_logy') = true; % do not log
mp_support_graph('it_legend_select') = 21; % how many shock legends to show
mp_support_graph('cl_colors') = 'jet';
```

MEAN(MN_V_GAIN_CHECK(A,Z))

Tabulate value and policies along savings and shocks:

```
% Set
ar_permute = [1,4,5,6,3,2];
% Value Function
st_title = ['MEAN(MN_V_U_GAIN_CHECK(A,Z)), welf_checks=' num2str(welf_checks) ', TR=' num2str(m
tb_az_v = ff_summ_nd_array(st_title, mn_V_U_gain_check, true, ["mean"], 4, 1, cl_mp_datasetdesc
```

```
xxx  MEAN(MN_V_U_GAIN_CHECK(A,Z)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	savings	mean_eta_1	mean_eta_2	mean_eta_3	mean_eta_4	mean_eta_5	mean_eta_6	mean_
1	0	0.16422	0.072711	0.044261	0.035185	0.031921	0.030628	0.0
2	0.00085734	0.16411	0.072685	0.044254	0.035183	0.031921	0.030628	0.0
3	0.0068587	0.16089	0.072203	0.044174	0.03516	0.031906	0.030615	0.0
4	0.023148	0.14347	0.068722	0.043233	0.034539	0.031319	0.030033	0.0
5	0.05487	0.12489	0.062983	0.040587	0.032437	0.029301	0.028027	0.0
6	0.10717	0.10956	0.056872	0.037152	0.029572	0.026538	0.025288	0.0
7	0.18519	0.093145	0.050122	0.032855	0.025873	0.022971	0.02175	0.0
8	0.29407	0.076208	0.042336	0.027832	0.021566	0.018844	0.017666	0.0
9	0.43896	0.063079	0.035973	0.023726	0.018152	0.015638	0.014514	0.0
10	0.625	0.052632	0.030859	0.020414	0.01541	0.013081	0.012018	0.0
11	0.85734	0.044007	0.026566	0.017638	0.013131	0.010971	0.0099637	0.00
12	1.1411	0.036806	0.022912	0.015283	0.011232	0.0092306	0.0082784	0.00

13	1.4815	0.030786	0.019772	0.013268	0.0096429	0.0077925	0.0068951	0.00
14	1.8836	0.025765	0.017067	0.011541	0.0083128	0.006606	0.005763	0.00
15	2.3525	0.021575	0.014731	0.010054	0.0071926	0.0056241	0.0048349	0.00
16	2.8935	0.018077	0.012701	0.0087613	0.006238	0.0048017	0.0040654	0.00
17	3.5117	0.015151	0.010936	0.0076339	0.0054202	0.00411	0.0034259	0.00
18	4.2121	0.012711	0.0094052	0.0066501	0.0047191	0.0035306	0.0028969	0.00
19	5	0.010676	0.008083	0.0057903	0.0041163	0.0030444	0.0024584	0.0
20	5.8805	0.008979	0.0069435	0.0050432	0.003597	0.0026352	0.0020946	0.00
21	6.8587	0.0075642	0.0059636	0.0043935	0.0031477	0.002289	0.0017916	0.00
22	7.9398	0.0063835	0.0051226	0.003827	0.0027577	0.001995	0.0015384	0.00
23	9.1289	0.0053974	0.0044023	0.0033352	0.0024197	0.0017445	0.0013264	0.00
24	10.431	0.0045736	0.0037863	0.0029093	0.0021264	0.0015304	0.0011486	0.000
25	11.852	0.0038839	0.0032595	0.0025394	0.0018711	0.0013461	0.00099862	0.000
26	13.396	0.0033057	0.0028093	0.0022181	0.0016485	0.001187	0.00087181	0.000
27	15.069	0.0028204	0.0024245	0.0019392	0.0014539	0.0010493	0.00076433	0.000
28	16.875	0.0024123	0.0020957	0.0016969	0.0012837	0.00092975	0.00067284	0.000
29	18.82	0.0020683	0.0018143	0.0014864	0.0011352	0.00082552	0.00059445	0.000
30	20.91	0.0017776	0.0015732	0.0013032	0.0010051	0.00073426	0.00052692	0.000
31	23.148	0.0015314	0.0013663	0.0011438	0.00089056	0.00065417	0.00046852	0.000
32	25.541	0.0013225	0.0011887	0.001005	0.00079002	0.00058386	0.00041788	0.000
33	28.093	0.0011448	0.0010361	0.00088415	0.00070181	0.00052199	0.00037374	0.000
34	30.81	0.00099337	0.00090469	0.00077879	0.00062417	0.0004674	0.00033511	0.000
35	33.697	0.00086393	0.00079137	0.00068687	0.00055572	0.00041911	0.00030114	0.000
36	36.758	0.00075305	0.0006935	0.0006066	0.00049531	0.0003762	0.00027117	0.000
37	40	0.00065787	0.00060885	0.00053645	0.00044195	0.00033809	0.00024466	0.000
38	43.427	0.00057598	0.0005355	0.00047505	0.00039475	0.00030432	0.00022111	0.000
39	47.044	0.00050536	0.00047182	0.00042126	0.00035297	0.00027428	0.00020014	0.000
40	50.856	0.00044434	0.00041648	0.00037408	0.00031594	0.00024745	0.00018144	0.000
41	54.87	0.00039149	0.00036829	0.00033267	0.00028312	0.00022344	0.00016475	0.000
42	59.089	0.00034563	0.00032626	0.00029627	0.00025398	0.00020201	0.0001498	0.000
43	63.519	0.00030574	0.00028952	0.00026422	0.00022808	0.00018284	0.00013638	9.761
44	68.164	0.00027097	0.00025735	0.00023597	0.00020506	0.00016566	0.00012431	8.924
45	73.032	0.00024062	0.00022915	0.00021104	0.00018456	0.00015024	0.00011344	8.175
46	78.125	0.00021406	0.00020439	0.00018901	0.00016629	0.00013638	0.00010362	7.504
47	83.45	0.00019078	0.00018259	0.00016951	0.00015	0.00012392	9.4736e-05	6.902
48	89.011	0.00017032	0.00016338	0.00015223	0.00013545	0.00011269	8.6725e-05	6.36
49	94.815	0.00015233	0.00014643	0.00013691	0.00012244	0.00010258	7.9488e-05	5.879
50	100.87	0.00013647	0.00013144	0.00012329	0.00011081	9.3454e-05	7.2935e-05	5.448
51	107.17	0.00012246	0.00011817	0.00011118	0.0001004	8.5225e-05	6.6992e-05	5.067
52	113.73	0.00011007	0.0001064	0.0001004	9.1075e-05	7.78e-05	6.1598e-05	4.733
53	120.55	9.9117e-05	9.5971e-05	9.0813e-05	8.2743e-05	7.113e-05	5.6758e-05	4.448
54	127.64	8.9565e-05	8.6875e-05	8.2434e-05	7.5433e-05	6.5251e-05	5.2505e-05	4.217
55	135	8.5718e-05	8.3153e-05	7.8927e-05	7.2281e-05	6.2631e-05	5.0567e-05	4.091

```
% Consumption
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(A,Z)), welf_checks=' num2str(welf_checks) ', TR=' num2str
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check, true, ["mean"], 4, 1, cl_mp_dat
```

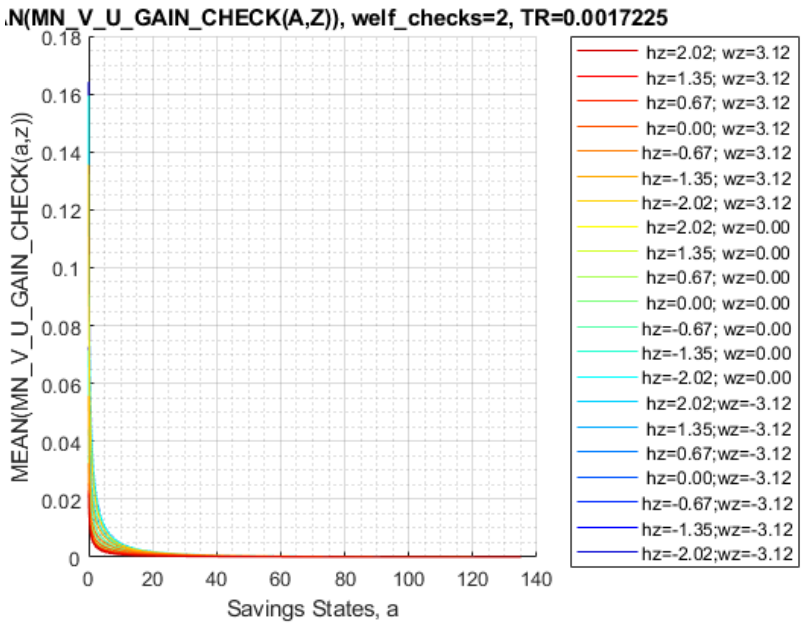
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(A,Z)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
```

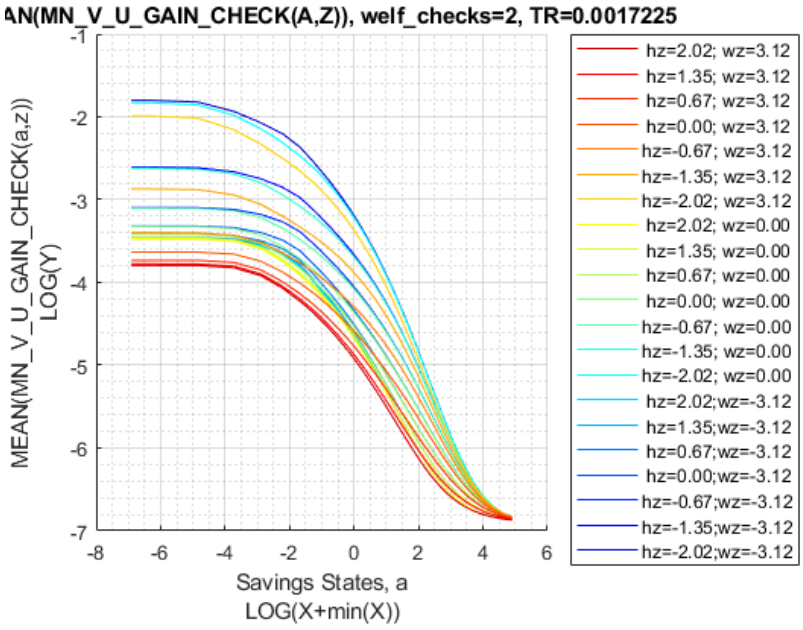
group	savings	mean_eta_1	mean_eta_2	mean_eta_3	mean_eta_4	mean_eta_5	mean_eta_6	mean_
1	0	0.74098	0.72618	0.66567	0.62344	0.58432	0.53574	0.4
2	0.00085734	0.7405	0.72615	0.66568	0.62348	0.58437	0.53579	0.4
3	0.0068587	0.72669	0.71733	0.66211	0.62125	0.58261	0.53437	0.4
4	0.023148	0.61478	0.61585	0.59388	0.58469	0.55844	0.51216	0.
5	0.05487	0.52118	0.52002	0.51447	0.51652	0.50302	0.46639	0.4
6	0.10717	0.4677	0.46139	0.46307	0.46985	0.45963	0.43084	0.3
7	0.18519	0.3973	0.37778	0.38896	0.39697	0.38775	0.36486	0.3
8	0.29407	0.32167	0.31637	0.29863	0.31313	0.31844	0.30063	0.2
9	0.43896	0.25268	0.24569	0.2474	0.24208	0.24978	0.24274	0.2
10	0.625	0.20528	0.20521	0.20529	0.20426	0.20225	0.20456	0.1
11	0.85734	0.1787	0.17548	0.175	0.17606	0.17597	0.17577	0.1
12	1.1411	0.16031	0.1574	0.15581	0.15574	0.15674	0.15888	0.1

13	1.4815	0.14645	0.14407	0.14238	0.14223	0.14334	0.146	0.1
14	1.8836	0.13586	0.1339	0.13226	0.13233	0.13318	0.13504	0.1
15	2.3525	0.12905	0.12748	0.12583	0.12631	0.12672	0.12812	0.
16	2.8935	0.1248	0.12332	0.12186	0.12224	0.12237	0.12327	0.1
17	3.5117	0.12316	0.12217	0.12089	0.121	0.12082	0.12193	0.1
18	4.2121	0.1215	0.12046	0.11937	0.11949	0.11943	0.12071	0.1
19	5	0.11643	0.11587	0.1151	0.1147	0.11496	0.11585	0.1
20	5.8805	0.11287	0.11225	0.11191	0.11124	0.11141	0.11229	0.1
21	6.8587	0.112	0.11157	0.11084	0.11056	0.11096	0.11147	0.1
22	7.9398	0.11185	0.11153	0.11102	0.11039	0.11115	0.11162	0.
23	9.1289	0.11312	0.1128	0.11262	0.11186	0.11264	0.11291	0.1
24	10.431	0.1119	0.11162	0.11143	0.11076	0.11149	0.11153	0.1
25	11.852	0.10944	0.10926	0.10903	0.10841	0.10905	0.10904	0.1
26	13.396	0.10862	0.10843	0.10826	0.10771	0.10818	0.10845	0.1
27	15.069	0.10932	0.10916	0.109	0.1085	0.10883	0.10922	0.1
28	16.875	0.1099	0.10978	0.10963	0.10944	0.10935	0.10984	0.1
29	18.82	0.10798	0.10788	0.10777	0.10777	0.10737	0.10794	0.
30	20.91	0.10705	0.10696	0.10687	0.10661	0.10637	0.10718	0.1
31	23.148	0.10761	0.10755	0.10746	0.10724	0.10705	0.10788	0.1
32	25.541	0.1096	0.10954	0.10947	0.1095	0.10909	0.10997	0.1
33	28.093	0.11004	0.10999	0.10994	0.11	0.10956	0.1104	0.1
34	30.81	0.10764	0.1076	0.10756	0.10759	0.10721	0.10793	0.1
35	33.697	0.10612	0.10609	0.10605	0.10605	0.10566	0.10632	0.1
36	36.758	0.10599	0.10597	0.10593	0.10592	0.10557	0.10611	0.
37	40	0.10721	0.1072	0.10716	0.10717	0.10715	0.10722	0.1
38	43.427	0.10865	0.10862	0.10859	0.10859	0.10875	0.10846	0.1
39	47.044	0.10816	0.10813	0.1081	0.10808	0.1081	0.10787	0.1
40	50.856	0.10782	0.10782	0.10778	0.10777	0.10749	0.10758	0.1
41	54.87	0.10726	0.10727	0.10724	0.10722	0.10708	0.10705	0.
42	59.089	0.10707	0.10706	0.10706	0.10703	0.10722	0.10686	0.1
43	63.519	0.10759	0.10756	0.10757	0.10755	0.10766	0.10735	0.1
44	68.164	0.1073	0.10729	0.10727	0.10728	0.10734	0.10703	0.1
45	73.032	0.10665	0.10665	0.10662	0.10663	0.10666	0.10632	0.1
46	78.125	0.10663	0.10662	0.10661	0.1066	0.10665	0.10617	0.1
47	83.45	0.10789	0.10788	0.10787	0.10785	0.10789	0.10772	0.1
48	89.011	0.10873	0.10873	0.10872	0.10871	0.10873	0.10887	0.1
49	94.815	0.10779	0.10779	0.10779	0.10778	0.10777	0.10783	0.1
50	100.87	0.10679	0.10679	0.10678	0.10677	0.10675	0.10663	0.1
51	107.17	0.10631	0.1063	0.10629	0.10627	0.10623	0.10561	0.09
52	113.73	0.10697	0.10696	0.10694	0.10689	0.10681	0.10606	0.09
53	120.55	0.1072	0.10716	0.10708	0.10697	0.10673	0.10588	0.09
54	127.64	0.10698	0.10688	0.10673	0.10652	0.10612	0.10498	0.09
55	135	0.10607	0.10591	0.10572	0.10546	0.10504	0.10393	0.08

Graph Mean Values:

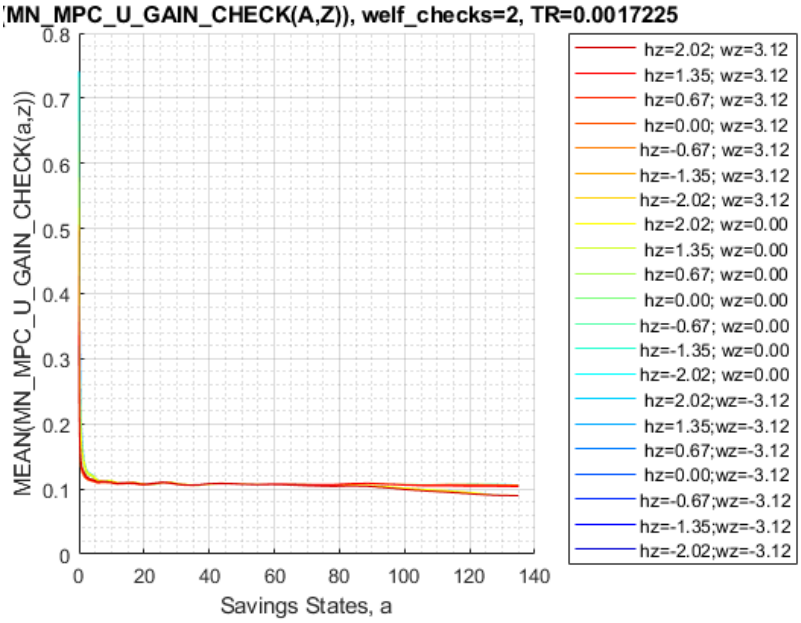
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(A,Z)), welf\_checks=' num2str(welf_checks) ', TR=' num2
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(a,z))'};
ff_graph_grid((tb_az_v{1:end, 3:end})', ar_st_eta_HS_grid, agrid, mp_support_graph);
```

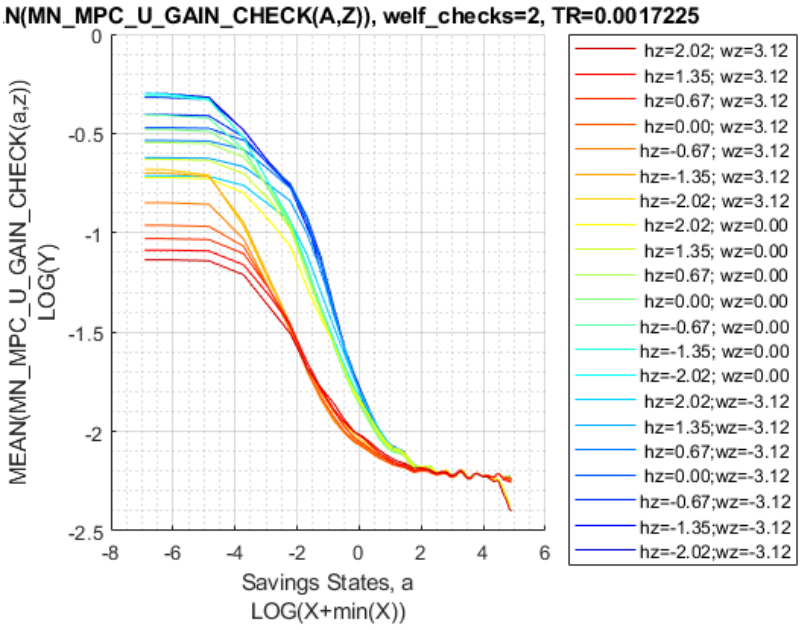




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
st_title = ['MEAN(MN\_MPC\_U\_GAIN\_CHECK(A,Z)), welf\_checks=' num2str(welf_checks) ', TR=' nu
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_MPC\_U\_GAIN\_CHECK(a,z))'};
ff_graph_grid((tb_az_c{1:end, 3:end})', ar_st_eta_HS_grid, agrid, mp_support_graph);
```





Analyze Marginal Value and MPC over Y(a,eta), Conditional On Kids, Marry, Age, Education

Income is generated by savings and shocks, what are the income levels generated by all the shock and savings points conditional on kids, marital status, age and educational levels. Plot on the Y axis MPC, and plot on the X axis income levels, use colors to first distinguish between different a levels, then use colors to distinguish between different eta levles.

Set Up date, Select Age 37vn

, unmarried, no kids, lower education:

```
% NaN(n_jgrid,n_agrid,n_etagrid,n_educgrid,n_marriedgrid,n_kidsgrid);
% 38 year old, unmarried, no kids, lower educated
```

```
% Only Household Head Shock Matters so select up to 'n_eta_H_grid'
mn_total_inc_jemk = total_inc_VFI(19,:,1:mp_params('n_eta_H_grid'),1,1,1);
mn_V_W_gain_check_use = ev19_jaeemk_check2 - ev19_jaeemk_check0;
mn_C_W_gain_check_use = ec19_jaeemk_check2 - ec19_jaeemk_check0;
```

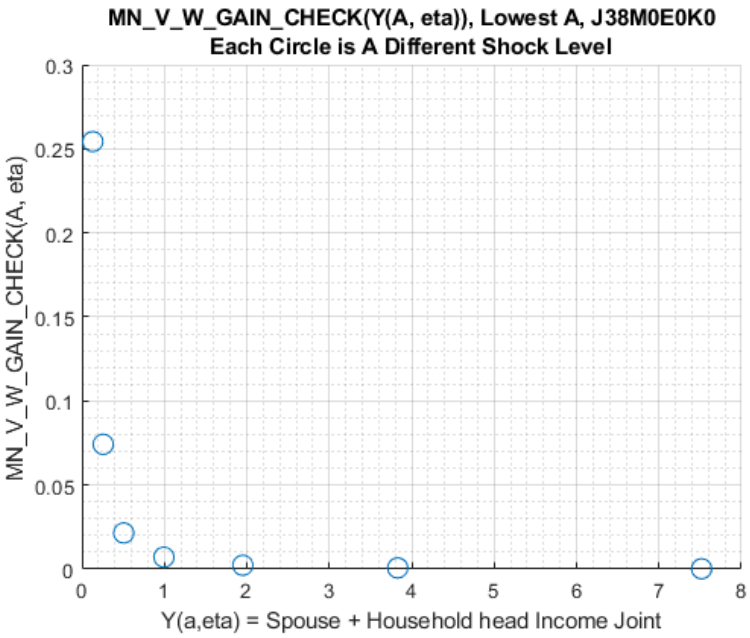
Select Age, Education, Marital, Kids Count:s

```
% Selections
it_age = 21; % +18
it_marital = 1; % 1 = unmarried
it_kids = 1; % 1 = kids is zero
it_educ = 1; % 1 = lower education
% Select: NaN(n_jgrid,n_agrid,n_etagrid,n_educgrid,n_marriedgrid,n_kidsgrid);
mn_C_W_gain_check_jemk = mn_C_W_gain_check_use(it_age, :, 1:mp_params('n_eta_H_grid'), it_educ,
mn_V_W_gain_check_jemk = mn_V_W_gain_check_use(it_age, :, 1:mp_params('n_eta_H_grid'), it_educ,
% Reshape, so shock is the first dim, a is the second
mt_total_inc_jemk = permute(mn_total_inc_jemk,[3,2,1]);
mt_C_W_gain_check_jemk = permute(mn_C_W_gain_check_jemk,[3,2,1]);
mt_C_W_gain_check_jemk(mt_C_W_gain_check_jemk<=1e-10) = 1e-10;
mt_V_W_gain_check_jemk = permute(mn_V_W_gain_check_jemk,[3,2,1]);
mt_V_W_gain_check_jemk(mt_V_W_gain_check_jemk<=1e-10) = 1e-10;
% Generate meshed a and shock grid
[mt_eta_H, mt_a] = ndgrid(eta_H_grid(1:mp_params('n_eta_H_grid')), agrid);
```

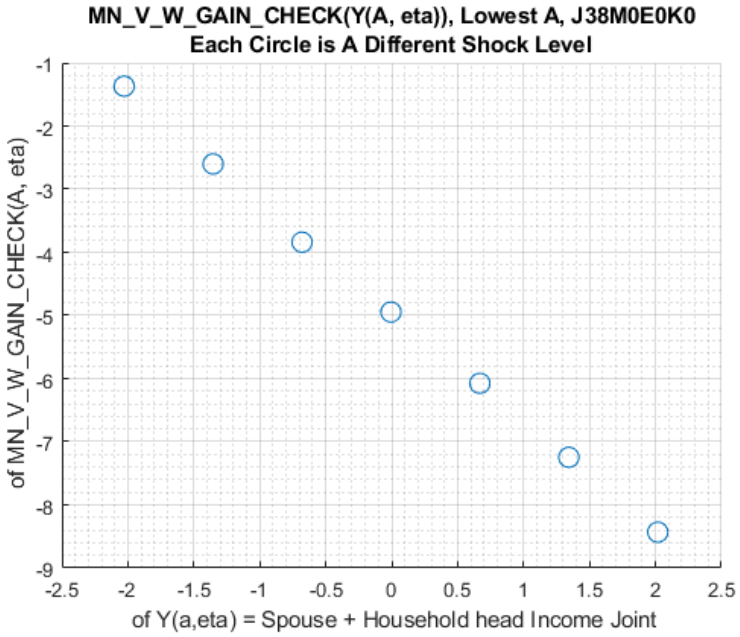
Marginal Value Gains, Color as Shock, Conditional on Age, Marital, Kids, and Education

How do shocks and a impact marginal value. First plot one asset level, variation comes only from increasingly higher shocks:

```
figure();
it_a = 1;
scatter((mt_total_inc_jemk(:,it_a)), (mt_V_W_gain_check_jemk(:,it_a)), 100);
title({'MN\_V\_W\_GAIN\_CHECK(Y(A, eta)), Lowest A, J38M0E0K0', ...
    'Each Circle is A Different Shock Level'});
xlabel('Y(a,eta) = Spouse + Household head Income Joint');
ylabel('MN\_V\_W\_GAIN\_CHECK(A, eta)');
grid on;
grid minor;
```



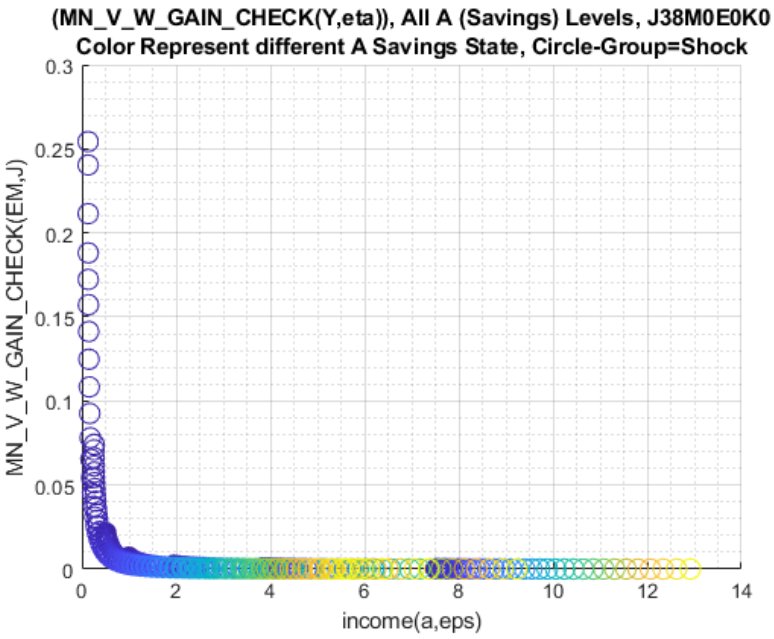

```
figure();
it_shock = 1;
scatter(log(mt_total_inc_jemk(:,it_a)), log(mt_V_W_gain_check_jemk(:,it_a)), 100);
title({'MN\_V\_W\_GAIN\_CHECK(Y(A, eta)), Lowest A, J38M0E0K0', ...
    'Each Circle is A Different Shock Level'});
xlabel(' of Y(a,eta) = Spouse + Household head Income Joint');
ylabel(' of MN\_V\_W\_GAIN\_CHECK(A, eta)');
grid on;
grid minor;
```



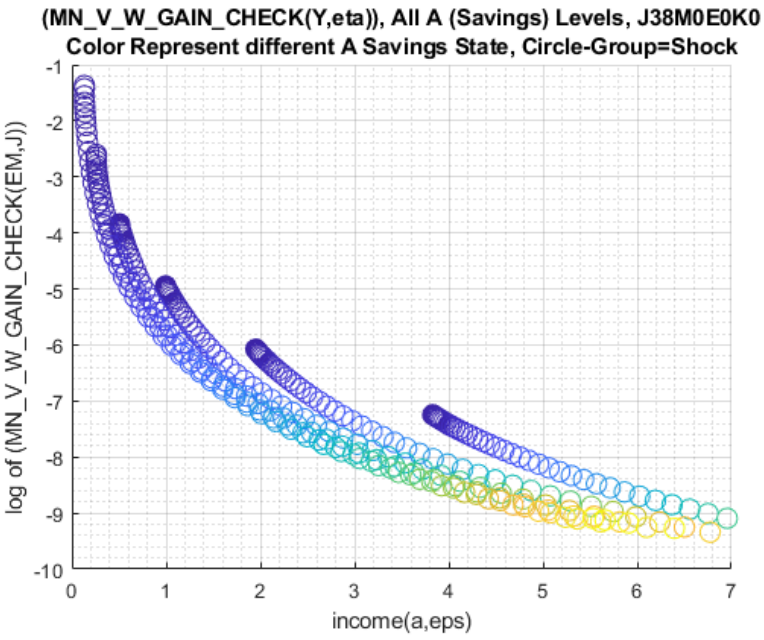
Plot all asset levels:

```
figure();
scatter((mt_total_inc_jemk(:)), (mt_V_W_gain_check_jemk(:)), 100, mt_a(:));
```

```
title({'(MN\_V\_W\_GAIN\_CHECK(Y,eta)), All A (Savings) Levels, J38M0E0K0', ...
    'Color Represent different A Savings State, Circle-Group=Shock'});
xlabel('income(a,eps)');
ylabel('MN\_V\_W\_GAIN\_CHECK(EM,J)');
grid on;
grid minor;
```



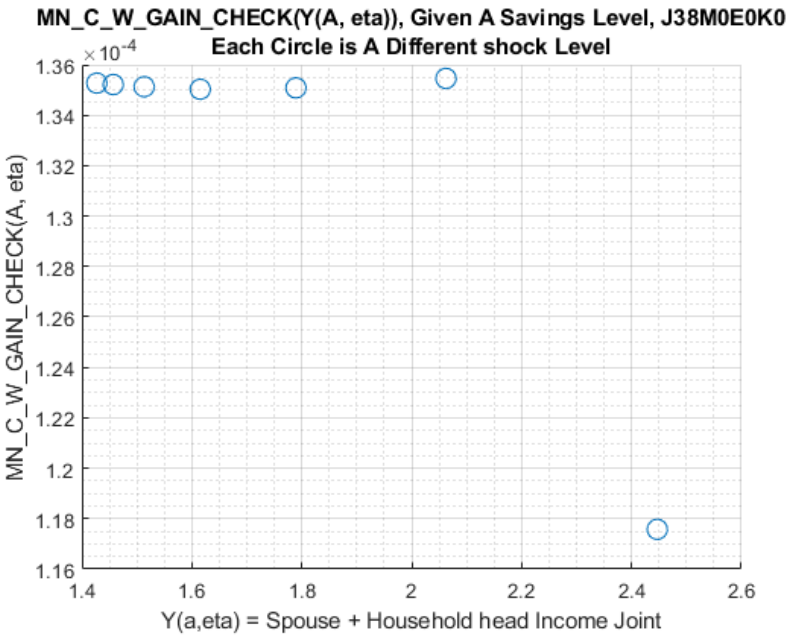

```
figure();
scatter((mt_total_inc_jemk(:)), log(mt_V_W_gain_check_jemk(:)), 100, mt_a(:));
title({'(MN\_V\_W\_GAIN\_CHECK(Y,eta)), All A (Savings) Levels, J38M0E0K0', ...
    'Color Represent different A Savings State, Circle-Group=Shock'});
xlabel('income(a,eps)');
ylabel('log of (MN\_V\_W\_GAIN\_CHECK(EM,J))');
xlim([0,7]);
grid on;
grid minor;
```



Marginal Consumption Gains, Color as Shock, Conditional on Age, Marital, Kids, and Education

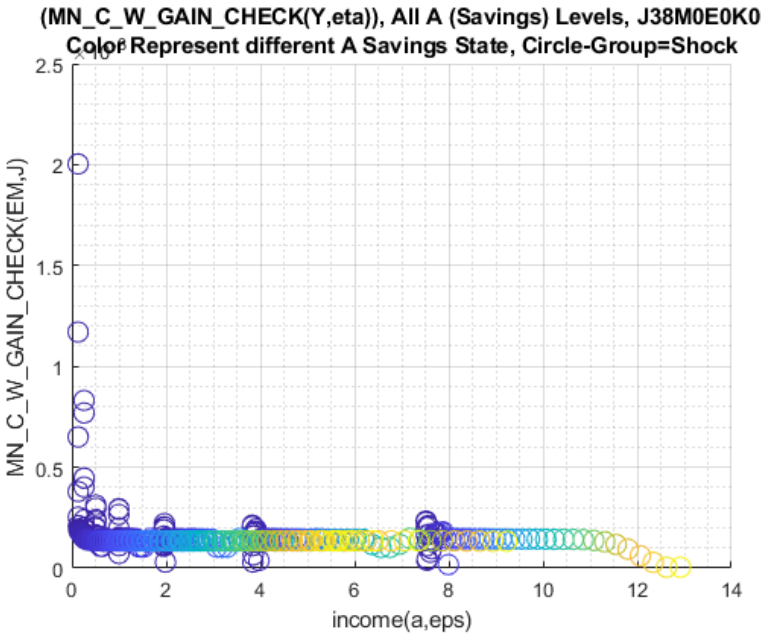
How do shocks and a impact marginal value. First plot one asset level, variation comes only from increasingly higher shocks:

```
figure();
it_a = 50;
scatter(log(mt_total_inc_jemk(:,it_a)), mt_C_W_gain_check_jemk(:,it_a), 100);
title({'MN\_C\_W\_GAIN\_CHECK(Y(A, eta)), Given A Savings Level, J38M0E0K0', ...
    'Each Circle is A Different shock Level'});
xlabel('Y(a,eta) = Spouse + Household head Income Joint');
ylabel('MN\_C\_W\_GAIN\_CHECK(A, eta)');
grid on;
grid minor;
```



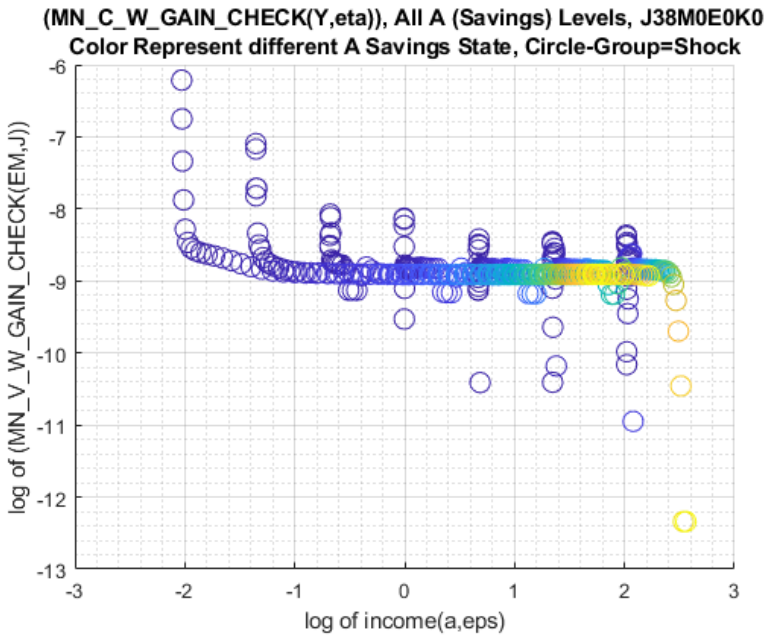
Plot all asset levels:

```
figure();
scatter((mt_total_inc_jemk(:)), (mt_C_W_gain_check_jemk(:)), 100, mt_a(:));
title({'(MN\_C\_W\_GAIN\_CHECK(Y,eta)), All A (Savings) Levels, J38M0E0K0', ...
    'Color Represent different A Savings State, Circle-Group=Shock'});
xlabel('income(a,eps)');
ylabel('MN\_C\_W\_GAIN\_CHECK(EM,J)');
grid on;
grid minor;
```




```
figure();
scatter(log(mt_total_inc_jemk(:)), log(mt_C_W_gain_check_jemk(:)), 100, mt_a(:));
title({'(MN\_C\_W\_GAIN\_CHECK(Y,eta)), All A (Savings) Levels, J38M0E0K0', ...
```

```
    'Color Represent different A Savings State, Circle-Group=Shock'});
xlabel('log of income(a,eps)');
ylabel('log of (MN\_V\_W\_GAIN\_CHECK(EM,J))');
grid on;
grid minor;
```



Analyze Kids and Marriage and Age

Aggregating over education, savings, and shocks, what are the differential effects of Marriage and Age.

```
% Generate some Data
mp_support_graph = containers.Map('KeyType', 'char', 'ValueType', 'any');
ar_row_grid = [...
    "k0M0", "K1M0", "K2M0", "K3M0", "K4M0", ...
    "k0M1", "K1M1", "K2M1", "K3M1", "K4M1"];
mp_support_graph('cl_st_xtitle') = {'Age'};
mp_support_graph('st_legend_loc') = 'best';
mp_support_graph('bl_graph_logy') = true; % do not log
mp_support_graph('st_rounding') = '6.2f'; % format shock legend
mp_support_graph('cl_scatter_shapes') = {...
    'o', 'd' ,'s', 'x', '*', ...
    'o', 'd', 's', 'x', '*'};
mp_support_graph('cl_colors') = {...
    'red', 'red', 'red', 'red', 'red'...
    'blue', 'blue', 'blue', 'blue', 'blue'};
```

MEAN(VAL(KM,J)), MEAN(AP(KM,J)), MEAN(C(KM,J))

Tabulate value and policies:

```
% Set
% NaN(n_jgrid,n_agrid,n_etagrid,n_educgrid,n_marriedgrid,n_kidsgrid);
ar_permute = [2,3,4,1,6,5];
% Value Function
```

```
st_title = ['MEAN(MN_V_U_GAIN_CHECK(KM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2str(
tb_az_v = ff_summ_nd_array(st_title, mn_V_U_gain_check, true, ["mean"], 3, 1, cl_mp_datasetdesc
```

```
xxx  MEAN(MN_V_U_GAIN_CHECK(KM,J)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	kids	marry	mean_age_18	mean_age_19	mean_age_20	mean_age_21	mean_age_22	mean_age_23
1	1	0	0.018081	0.017399	0.016444	0.015175	0.01411	0.013211
2	2	0	0.024198	0.023312	0.022022	0.020238	0.018732	0.017454
3	3	0	0.027972	0.027119	0.025477	0.023434	0.02171	0.020247
4	4	0	0.031555	0.030668	0.028763	0.026469	0.024532	0.022889
5	5	0	0.03448	0.033614	0.031423	0.028959	0.026879	0.025115
6	1	1	0.0041769	0.0038579	0.0035234	0.0032138	0.0029477	0.0027235
7	2	1	0.0057601	0.0053292	0.004867	0.0044266	0.0040531	0.003739
8	3	1	0.0068486	0.0063558	0.0058113	0.0052918	0.0048508	0.0044744
9	4	1	0.0083574	0.007782	0.0071369	0.006497	0.0059602	0.0054988
10	5	1	0.010213	0.0095931	0.0088366	0.0080803	0.0074291	0.0068739

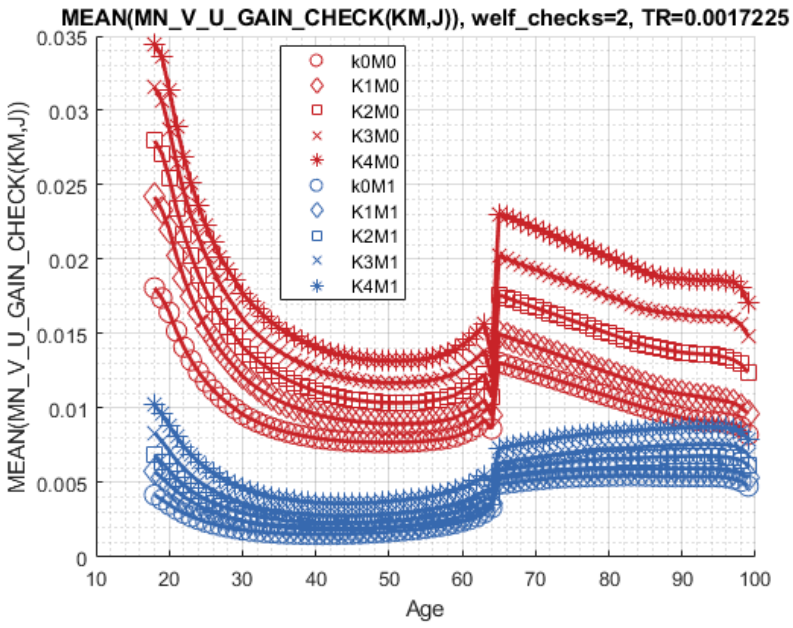
```
% Consumption Function
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(KM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2st
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check, true, ["mean"], 3, 1, cl_mp_dat
```

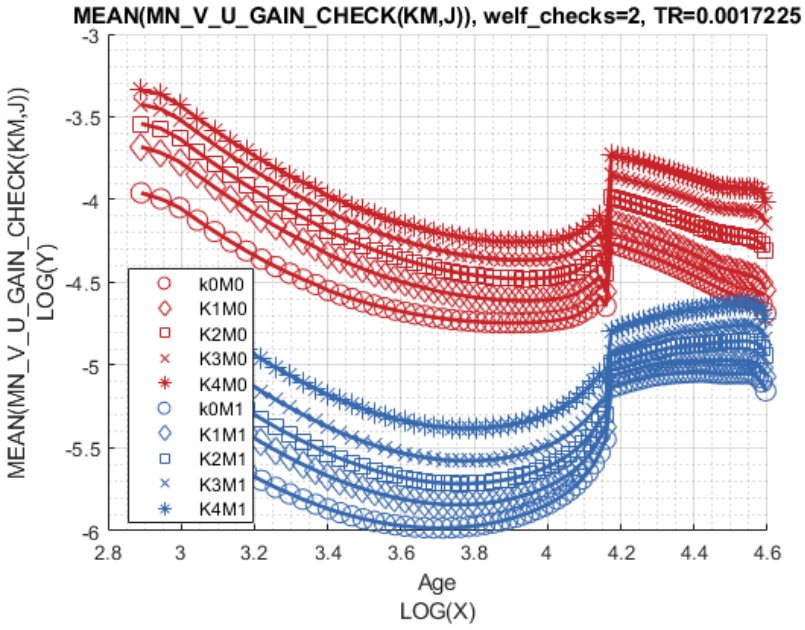
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(KM,J)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	kids	marry	mean_age_18	mean_age_19	mean_age_20	mean_age_21	mean_age_22	mean_age_23
1	1	0	0.074031	0.10225	0.11599	0.11147	0.10577	0.095081
2	2	0	0.089094	0.11803	0.13543	0.132	0.12978	0.12523
3	3	0	0.10426	0.14094	0.15528	0.1534	0.15067	0.1437
4	4	0	0.1102	0.15019	0.16182	0.16053	0.1588	0.15606
5	5	0	0.12088	0.15838	0.16724	0.16598	0.16431	0.16225
6	1	1	0.088885	0.10829	0.11259	0.10847	0.10521	0.10685
7	2	1	0.093837	0.11495	0.1194	0.11493	0.11003	0.1082
8	3	1	0.10258	0.126	0.1301	0.12812	0.12491	0.1238
9	4	1	0.10896	0.1303	0.1351	0.13564	0.13402	0.1313
10	5	1	0.12962	0.14112	0.14504	0.14418	0.14327	0.14158

Graph Mean Values:

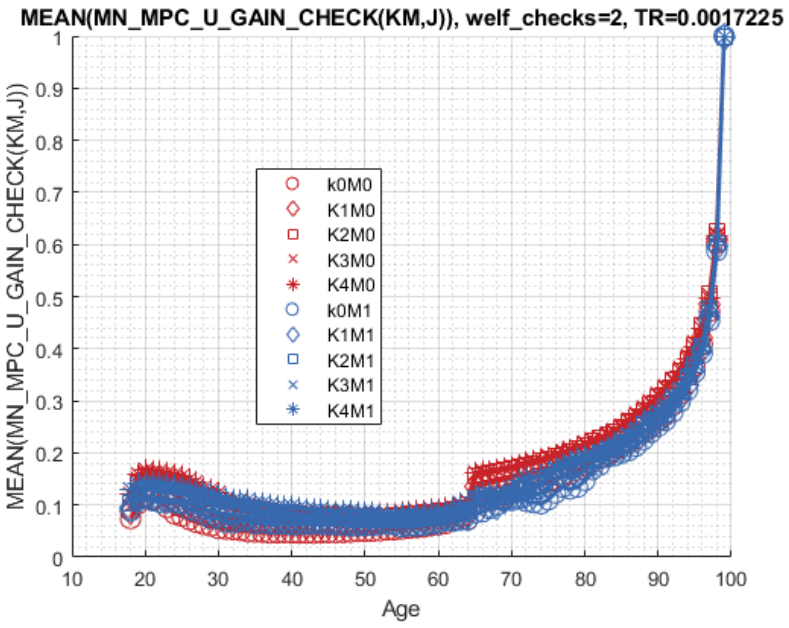
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(KM,J)), welf\_checks=' num2str(welf_checks) ', TR=' num
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(KM,J))'};
ff_graph_grid((tb_az_v{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);
```

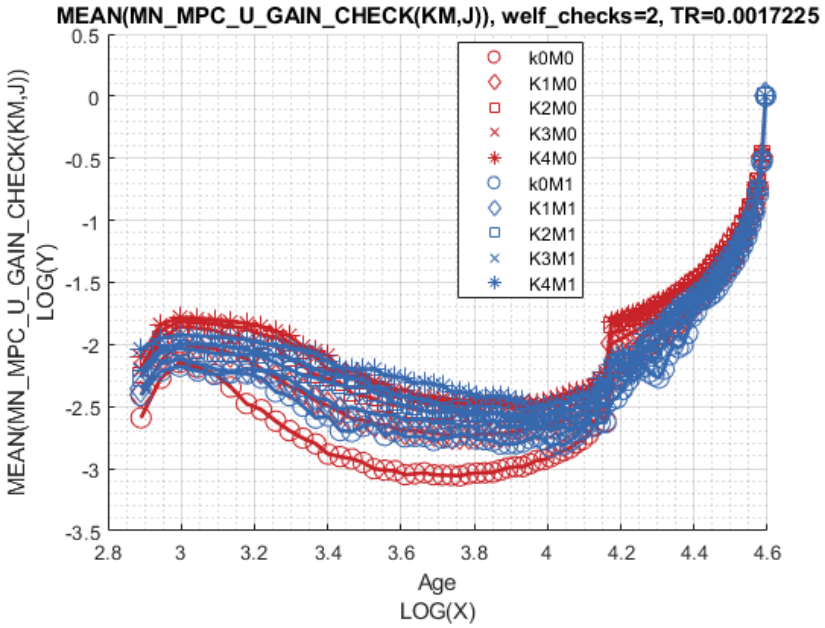




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
st_title = ['MEAN(MN\_MPC\_U\_GAIN\_CHECK(KM,J)), welf\_checks=' num2str(welf_checks) ', TR=' r
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_MPC\_U\_GAIN\_CHECK(KM,J))'};
ff_graph_grid((tb_az_c{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);
```





Analyze Education and Marriage

Aggregating over education, savings, and shocks, what are the differential effects of Marriage and Age.

```
% Generate some Data
mp_support_graph = containers.Map('KeyType', 'char', 'ValueType', 'any');
ar_row_grid = ["E0M0", "E1M0", "E0M1", "E1M1"];
mp_support_graph('cl_st_xtitle') = {'Age'};
mp_support_graph('st_legend_loc') = 'best';
mp_support_graph('bl_graph_logy') = true; % do not log
mp_support_graph('st_rounding') = '6.2f'; % format shock legend
mp_support_graph('cl_scatter_shapes') = {'*', 'p', '*','p' };
mp_support_graph('cl_colors') = {'red', 'red', 'blue', 'blue'};
```

MEAN(VAL(EM,J)), MEAN(AP(EM,J)), MEAN(C(EM,J))

Tabulate value and policies:

```
% Set
% NaN(n_jgrid,n_agrid,n_etagrid,n_educgrid,n_marriedgrid,n_kidsgrid);
ar_permute = [2,3,6,1,4,5];
% Value Function
st_title = ['MEAN(MN_V_U_GAIN_CHECK(EM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2str(
tb_az_v = ff_summ_nd_array(st_title, mn_V_U_gain_check, true, ["mean"], 3, 1, cl_mp_datasetdesc
```

```
xxx  MEAN(MN_V_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
    group    edu    marry    mean_age_18    mean_age_19    mean_age_20    mean_age_21    mean_age_22    mean_age_23
    _____    ___    _____    ___________    ___________    ___________    ___________    ___________    ___________

      1       0       0         0.028598       0.027879       0.026816        0.02537       0.024081       0.022927
      2       1       0         0.025917       0.024966       0.022835        0.02034       0.018305        0.01664
      3       0       1        0.0077546      0.0072639      0.0067503      0.0062497      0.0058142      0.0054304
      4       1       1        0.0063878      0.0059033      0.0053197      0.0047542      0.0042822      0.0038935
```

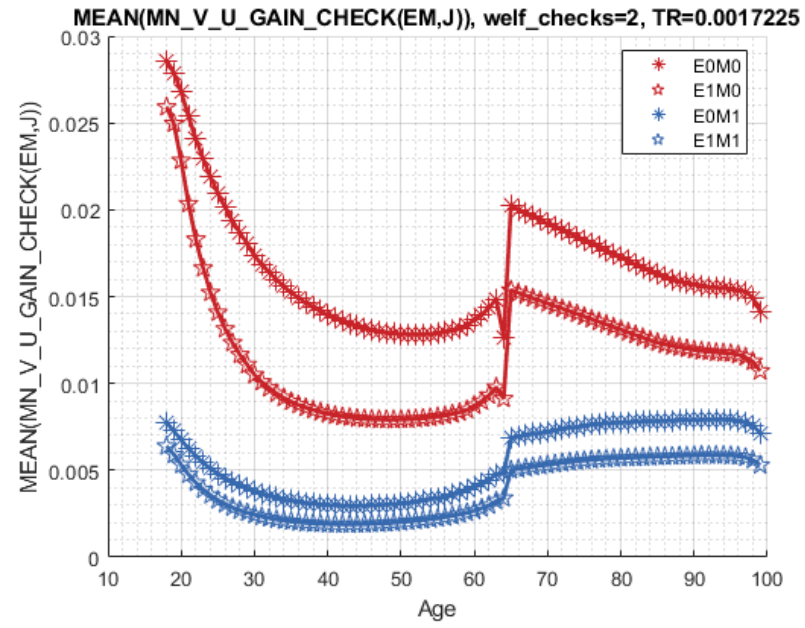
```
% Consumption
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2st
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check, true, ["mean"], 3, 1, cl_mp_dat
```

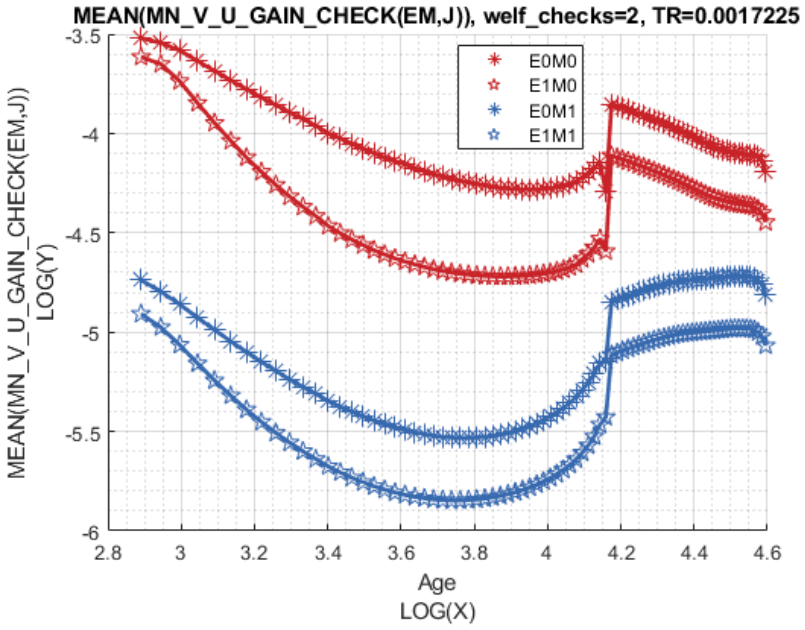
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
    group    edu    marry    mean_age_18    mean_age_19    mean_age_20    mean_age_21    mean_age_22    mean_age_23
    _____    ___    _____    ___________    ___________    ___________    ___________    ___________    ___________

      1       0       0         0.083839        0.10872        0.12214        0.11934        0.11723        0.11285
      2       1       0         0.11555         0.1592        0.17216        0.17002         0.1665        0.16008
      3       0       1        0.096421        0.11218        0.11688        0.11343        0.11043        0.11038
      4       1       1         0.11313        0.13609        0.14001        0.13911        0.13655        0.13431
```

Graph Mean Values:

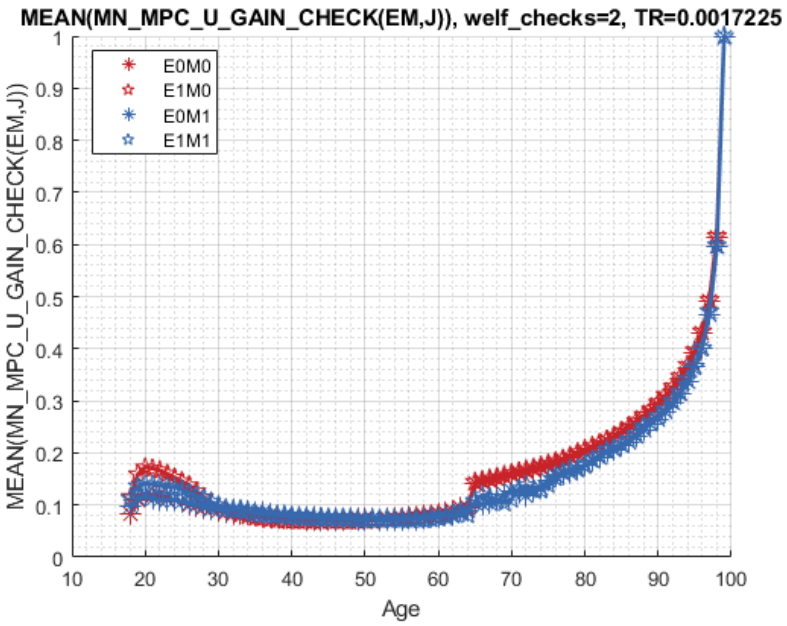
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(EM,J)), welf\_checks=' num2str(welf_checks) ', TR=' num
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(EM,J))'};
ff_graph_grid((tb_az_v{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);
```

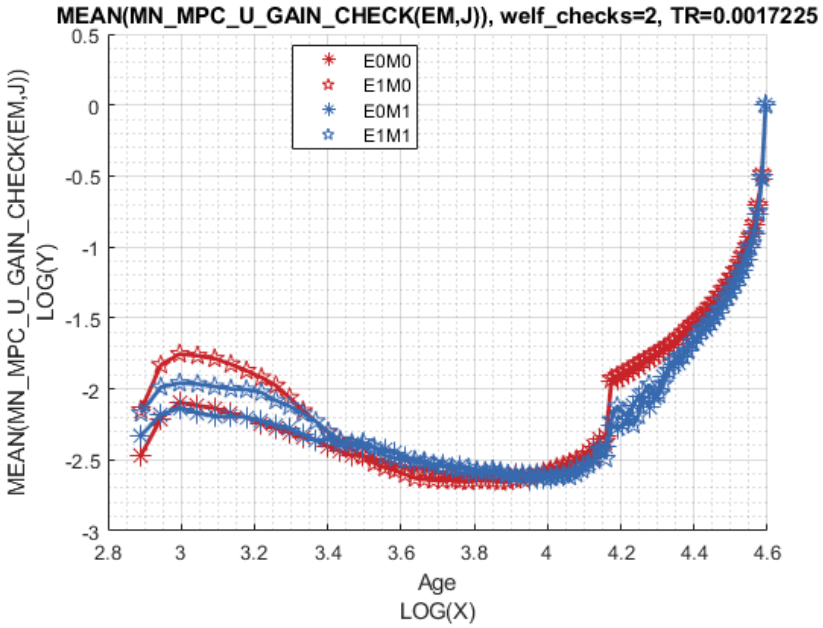




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
st_title = ['MEAN(MN\_MPC\_U\_GAIN\_CHECK(EM,J)), welf\_checks=' num2str(welf_checks) ', TR=' r
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_MPC\_U\_GAIN\_CHECK(EM,J))'};
ff_graph_grid((tb_az_c{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);
```


MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0017225
E0M0
E1M0
E0M1
E1M1
MEAN(MN_MPC_U_GAIN_CHECK(EM,J))
Age


MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0017225
E0M0
E1M0
E0M1
E1M1
MEAN(MN_MPC_U_GAIN_CHECK(EM,J))
LOG(Y)
Age
LOG(X)