2020 Full States EV and EC of One Check

This is the example vignette for function: **snw_evuvw20_jaeemk** from the **PrjOptiSNW Package.** 2020 integrated over VU and VW

Test SNW_EVUVW20_JAEEMK Defaults Dense

Solve for policy functions and obtain distributions.

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=20.6119
```

```
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=16
```

```
[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=51.8429
```

```
% Get Matrixes
cl_st_precompute_list = {'a', ...
    'inc', 'inc_unemp', 'spouse_inc', 'spouse_inc_unemp', 'ref_earn_wageind_grid'};
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=23.0315
```

Solve for 2020 Evuvw With 0 and 2 Checks

```
% Call Function
welf_checks = 0;
[ev20_jaeemk_check0, ec20_jaeemk_check0] = snw_evuvw20_jaeemk(...
    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=0.64818
```

```
% Call Function
welf_checks = 2;
[ev20_jaeemk_check2, ec20_jaeemk_check2] = snw_evuvw20_jaeemk(...
    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=0.44581
```

Differences between Checks in Expected Value and Expected Consumption

```
mn_V_U_gain_check = ev20_jaeemk_check2 - ev20_jaeemk_check0;
mn_MPC_U_gain_share_check = (ec20_jaeemk_check2 - ec20_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 = 18:100;
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 xxxxxxxxxxxxxxxxxxxxxxxxxxxx

group	savings	mean_eta_1	mean_eta_2	mean_eta_3	mean_eta_4	mean_eta_5	mean_eta_6	mean_
1	0	0.27912	0.11616	0.067433	0.053393	0.049158	0.047687	0.
2	0.00085734	0.27485	0.11517	0.066984	0.053023	0.048799	0.047329	0.0
3	0.0068587	0.23632	0.10481	0.062008	0.048905	0.044801	0.04335	0.0
4	0.023148	0.1863	0.08898	0.054024	0.042348	0.038458	0.037046	0.0
5	0.05487	0.14109	0.071528	0.044733	0.034862	0.031271	0.029909	0.0
6	0.10717	0.10829	0.056442	0.036039	0.027915	0.024677	0.023377	0.0
7	0.18519	0.087242	0.046352	0.02987	0.023041	0.020157	0.018927	0.
8	0.29407	0.072208	0.039367	0.025433	0.019424	0.016782	0.015628	0.0
9	0.43896	0.060221	0.033761	0.021874	0.016469	0.014026	0.012929	0.0
10	0.625	0.050442	0.029129	0.018949	0.014075	0.011802	0.010759	0.0
11	0.85734	0.042316	0.025218	0.016484	0.012083	0.0099683	0.0089837	0.00
12	1.1411	0.035502	0.021857	0.014374	0.010409	0.0084519	0.0075155	0.00
13	1.4815	0.029772	0.01894	0.012549	0.0089945	0.0071841	0.0063034	0.00
14	1.8836	0.024956	0.016396	0.010963	0.0077932	0.0061208	0.0052935	0.00
15	2.3525	0.020913	0.014175	0.0095793	0.0067677	0.0052312	0.0044577	0.00
16	2.8935	0.01753	0.012238	0.008369	0.0058896	0.0044819	0.003759	0.00
17	3.5117	0.014702	0.010552	0.0073101	0.0051351	0.0038505	0.0031792	0.00
18	4.2121	0.012342	0.0090878	0.006382	0.0044849	0.0033193	0.0026984	0.00
19	5	0.010373	0.0078203	0.0055676	0.0039231	0.0028711	0.0022965	0.00
20	5.8805	0.0087301	0.0067252	0.0048575	0.0034361	0.002492	0.0019615	0.0
21	6.8587	0.0073585	0.0057817	0.004238	0.0030132	0.0021701	0.0016824	0.00
22	7.9398	0.0062131	0.0049708	0.0036966	0.0026454	0.001896	0.0014485	0.00
23	9.1289	0.0052558	0.0042752	0.0032257	0.0023256	0.0016619	0.0012519	0.00
24	10.431	0.0044552	0.0036794	0.0028167	0.002047	0.0014609	0.0010864	0.000
25	11.852	0.0037847	0.0031694	0.0024609	0.0018039	0.0012875	0.00094651	0.000
26	13.396	0.0032224	0.0027332	0.0021515	0.0015915	0.0011375	0.00082806	0.000
27	15.069	0.0027501	0.00236	0.0018824	0.0014053	0.0010072	0.00072742	0.000

28	16.875	0.0023527	0.0020407	0.0016483	0.001242	0.00089377	0.00064142	0.000
29	18.82	0.0020176	0.0017673	0.0014445	0.0010993	0.0007946	0.00056753	0.000
30	20.91	0.0017343	0.0015329	0.0012672	0.00097404	0.00070765	0.0005038	0.000
31	23.148	0.0014944	0.0013318	0.0011128	0.00086375	0.00063126	0.00044868	0.000
32	25.541	0.0012908	0.0011591	0.00097818	0.00076679	0.00056405	0.00040076	0.000
33	28.093	0.0011176	0.0010105	0.0008609	0.00068159	0.00050477	0.0003589	0.000
34	30.81	0.00096988	0.00088251	0.00075856	0.00060649	0.00045237	0.00032217	0.000
35	33.697	0.0008436	0.00077212	0.00066925	0.00054025	0.00040595	0.00028984	0.000
36	36.758	0.00073542	0.00067675	0.0005912	0.00048173	0.00036464	0.00026127	0.000
37	40	0.00064253	0.00059424	0.00052296	0.00043001	0.00032792	0.00023595	0.000
38	43.427	0.00056261	0.00052272	0.00046322	0.00038423	0.00029534	0.00021344	0.000
39	47.044	0.00049368	0.00046065	0.00041086	0.00034368	0.00026633	0.00019338	0.000
40	50.856	0.00043412	0.00040667	0.00036494	0.00030774	0.0002404	0.00017546	0.000
41	54.87	0.00038251	0.00035966	0.00032459	0.00027584	0.00021718	0.00015944	0.000
42	59.089	0.00033773	0.00031865	0.00028912	0.00024751	0.00019642	0.00014507	0.000
43	63.519	0.00029877	0.00028279	0.00025789	0.00022233	0.00017785	0.00013216	9.401
44	68.164	0.00026481	0.0002514	0.00023035	0.00019993	0.00016119	0.00012053	8.602
45	73.032	0.00023517	0.00022388	0.00020604	0.00017998	0.00014624	0.00011006	7.887
46	78.125	0.00020923	0.00019969	0.00018455	0.0001622	0.00013279	0.00010058	7.245
47	83.45	0.00018648	0.00017841	0.00016553	0.00014633	0.00012068	9.1999e-05	6.669
48	89.011	0.00016649	0.00015965	0.00014868	0.00013216	0.00010978	8.4257e-05	6.152
49	94.815	0.00014891	0.0001431	0.00013373	0.00011949	9.995e-05	7.7256e-05	5.688
50	100.87	0.00013342	0.00012846	0.00012044	0.00010816	9.1084e-05	7.0912e-05	5.275
51	107.17	0.00011973	0.0001155	0.00010862	9.8007e-05	8.308e-05	6.5154e-05	4.909
52	113.73	0.00010762	0.000104	9.8089e-05	8.8914e-05	7.5856e-05	5.9928e-05	4.588
53	120.55	9.6907e-05	9.3809e-05	8.873e-05	8.0788e-05	6.9365e-05	5.5235e-05	4.315
54	127.64	8.7567e-05	8.4918e-05	8.0546e-05	7.3655e-05	6.364e-05	5.1109e-05	4.094
55	135	8.7567e-05	8.4918e-05	8.0546e-05	7.3656e-05	6.3641e-05	5.1109e-05	4.094

```
% 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
```

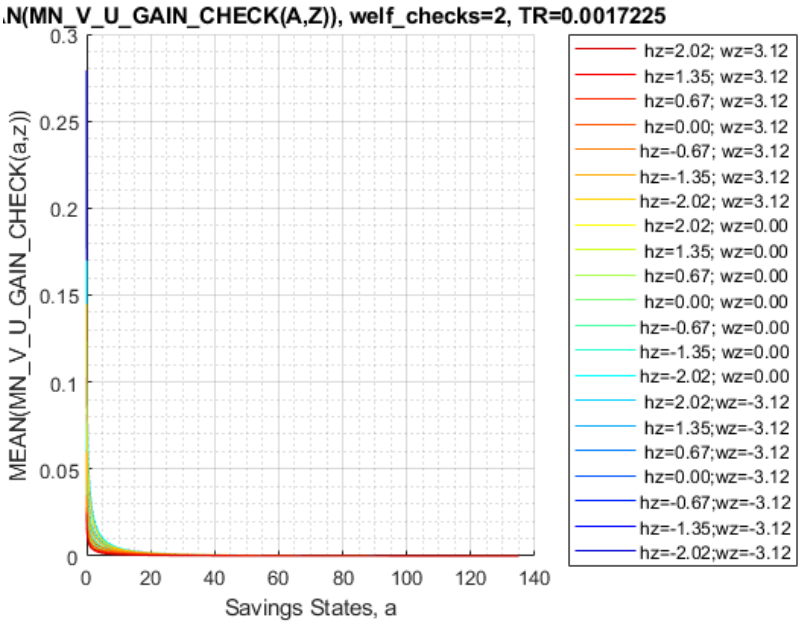
xxx MEAN(MN_MPC_U_GAIN_CHECK(A,Z)), welf_checks=2, TR=0.0017225 xxxxxxxxxxxxxxxxxxxxxxxxxxx

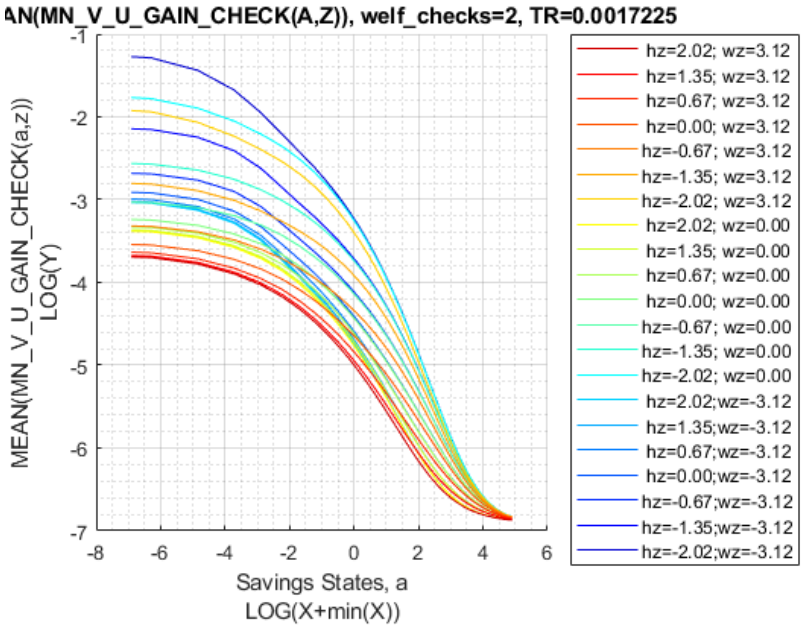
group	savings	mean_eta_1	mean_eta_2	mean_eta_3	mean_eta_4	mean_eta_5	mean_eta_6	mean_
1	0	0.9469	0.94276	0.8717	0.79605	0.7222	0.6457	0.5
2	0.00085734	0.93411	0.9349	0.86947	0.79402	0.72155	0.64512	0.5
3	0.0068587	0.81862	0.82623	0.78945	0.74561	0.6858	0.6149	0.5
4	0.023148	0.71821	0.72464	0.70291	0.65987	0.61284	0.55283	0.4
5	0.05487	0.65702	0.66511	0.64788	0.60732	0.55769	0.51161	0.4
6	0.10717	0.54356	0.56685	0.56196	0.52654	0.48344	0.43245	0.3
7	0.18519	0.34762	0.37134	0.3935	0.38764	0.35935	0.3156	0.2
8	0.29407	0.27014	0.27012	0.26968	0.2781	0.27577	0.25618	0.2
9	0.43896	0.22635	0.22379	0.21808	0.21552	0.21153	0.20821	0.2
10	0.625	0.18643	0.18323	0.18257	0.18047	0.17918	0.17475	0.1
11	0.85734	0.16247	0.15989	0.15903	0.15917	0.1587	0.16115	0.1
12	1.1411	0.14602	0.14447	0.14433	0.1439	0.14242	0.14773	0.
13	1.4815	0.13936	0.13684	0.13479	0.13517	0.13698	0.13519	0.1
14	1.8836	0.13686	0.13554	0.13207	0.13035	0.13296	0.13246	0.1
15	2.3525	0.13507	0.13247	0.12893	0.12723	0.12724	0.1284	0.1
16	2.8935	0.12673	0.12656	0.12382	0.12237	0.12168	0.12377	0.1
17	3.5117	0.12256	0.12043	0.11953	0.11889	0.11806	0.11895	0.1
18	4.2121	0.11793	0.11791	0.11793	0.11601	0.11689	0.11664	0.1
19	5	0.11739	0.11678	0.11684	0.11637	0.11577	0.11666	0.1
20	5.8805	0.11528	0.11487	0.11456	0.11528	0.11413	0.11533	0.1
21	6.8587	0.11285	0.11244	0.11208	0.11157	0.11191	0.11282	0.
22	7.9398	0.11153	0.11118	0.11088	0.1103	0.11091	0.11118	0.1
23	9.1289	0.1122	0.11192	0.11174	0.11112	0.11192	0.11192	0.1
24	10.431	0.11098	0.11077	0.11058	0.11009	0.11068	0.11081	0.1
25	11.852	0.10854	0.10833	0.10817	0.10779	0.10805	0.10841	0.
26	13.396	0.10801	0.10783	0.10765	0.10745	0.10738	0.108	0.1
27	15.069	0.11027	0.11014	0.10997	0.10974	0.1096	0.1103	0.1

28	16.875	0.11158	0.11146	0.11134	0.11129	0.11095	0.11166	0.1
29	18.82	0.11001	0.1099	0.1098	0.10996	0.1093	0.11003	0.1
30	20.91	0.1067	0.10661	0.10651	0.10645	0.10603	0.10679	0.
31	23.148	0.10684	0.10677	0.10669	0.10658	0.10631	0.10702	0.1
32	25.541	0.10847	0.10842	0.10835	0.1084	0.108	0.10868	0.1
33	28.093	0.1086	0.10855	0.10851	0.10857	0.10826	0.10885	0.1
34	30.81	0.10826	0.10822	0.10817	0.10821	0.10802	0.10841	0.1
35	33.697	0.10867	0.10864	0.1086	0.10861	0.10855	0.10875	0.
36	36.758	0.10953	0.10951	0.10947	0.10947	0.10941	0.10949	0.1
37	40	0.10783	0.1078	0.10777	0.10776	0.10793	0.10764	0.1
38	43.427	0.10628	0.10627	0.10624	0.10623	0.1065	0.10593	0.1
39	47.044	0.10618	0.10618	0.10615	0.10614	0.1063	0.10584	0.1
40	50.856	0.10875	0.10873	0.10871	0.1087	0.10863	0.10854	0.1
41	54.87	0.1092	0.10918	0.10917	0.10916	0.10914	0.10898	0.1
42	59.089	0.10782	0.1078	0.10779	0.10778	0.10797	0.10758	0.1
43	63.519	0.10676	0.10675	0.10673	0.10673	0.10684	0.10646	0.1
44	68.164	0.10765	0.10764	0.10763	0.10762	0.1077	0.10743	0.1
45	73.032	0.10869	0.10868	0.10867	0.10866	0.10871	0.10864	0.1
46	78.125	0.10779	0.10778	0.10777	0.10776	0.1078	0.10768	0.1
47	83.45	0.10629	0.10629	0.10628	0.10627	0.1063	0.10635	0.1
48	89.011	0.10682	0.10682	0.10681	0.1068	0.10681	0.10702	0.
49	94.815	0.10832	0.10832	0.10831	0.1083	0.1083	0.10837	0.1
50	100.87	0.10817	0.10817	0.10816	0.10815	0.10814	0.10815	0.1
51	107.17	0.10742	0.10742	0.10741	0.10739	0.10736	0.10704	0.1
52	113.73	0.10763	0.10763	0.10761	0.10757	0.1075	0.10702	0.09
53	120.55	0.10824	0.10821	0.10815	0.10805	0.10784	0.1071	0.09
54	127.64	0.10659	0.10646	0.10629	0.10605	0.10564	0.10453	0.0
55	135	0.10659	0.10646	0.10629	0.10605	0.10564	0.10453	0.0

Graph Mean Values:

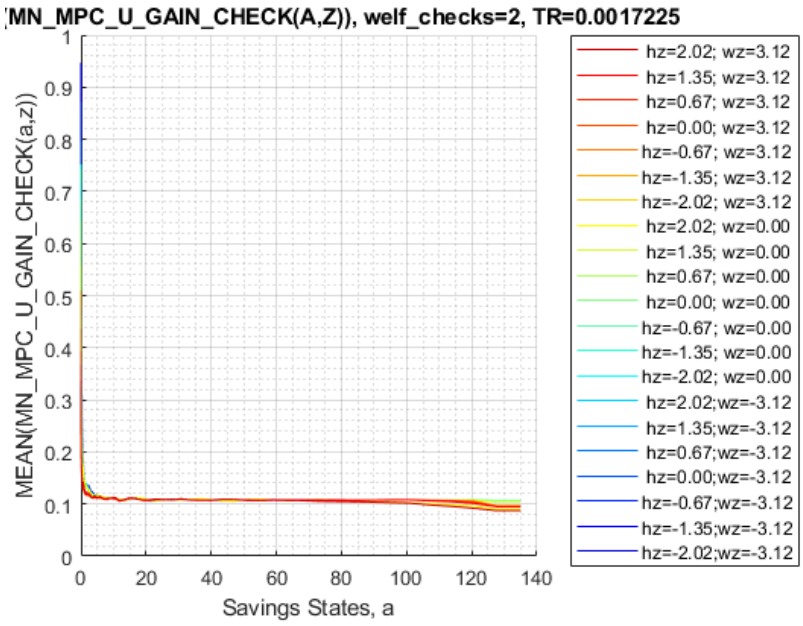
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(A,Z)), welf\_checks=' num2str(welf_checks) ', TR=' num2
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(a,z))'};
ff_graph_grid((tb_az_v{1:end, 3:end})', ar_st_eta_HS_grid, agrid, mp_support_graph);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

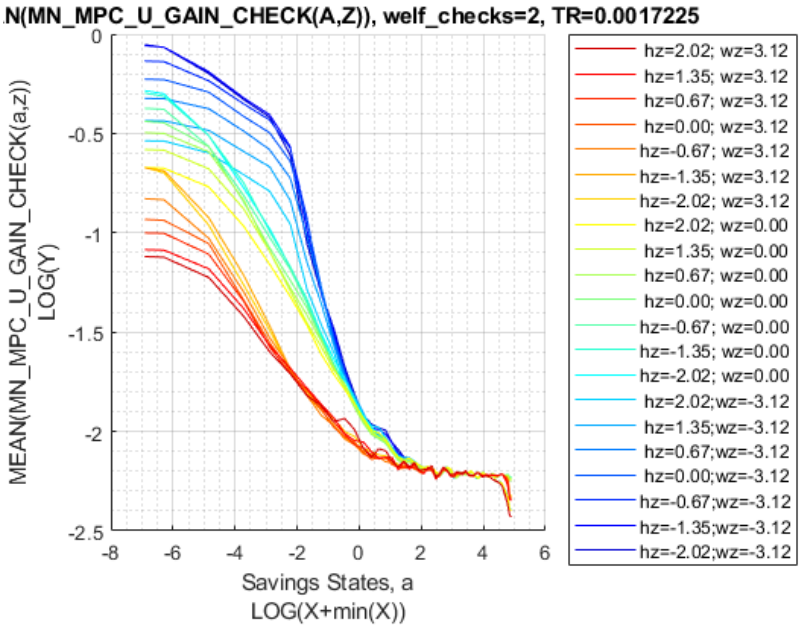




Graph Mean Consumption (MPC: Share of Check Consumed):

```
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 38, 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(20,:,1:mp_params('n_eta_H_grid'),1,1,1);
mn_V_W_gain_check_use = ev20_jaeemk_check2 - ev20_jaeemk_check0;
mn_C_W_gain_check_use = ec20_jaeemk_check2 - ec20_jaeemk_check0;
```

Select Age, Education, Marital, Kids Count:s

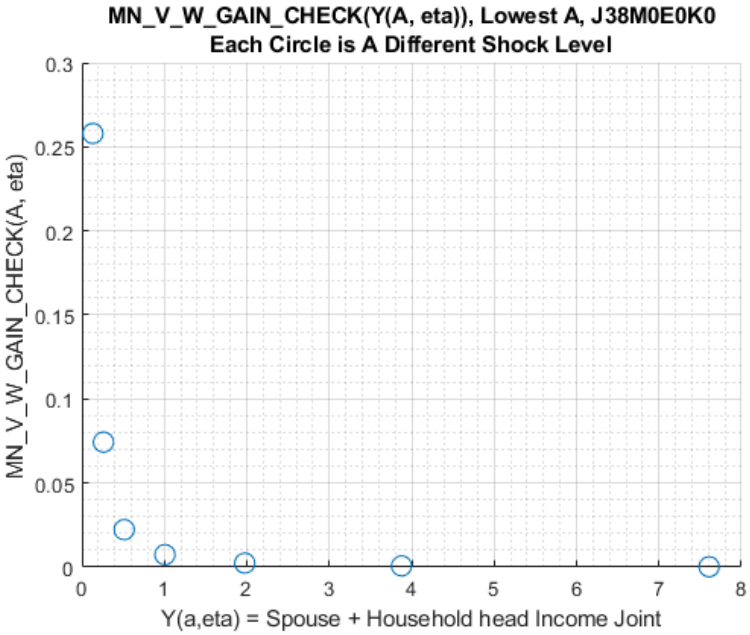
```
% Selections
it_age = 21; % +18
it_marital = 1; % 1 = unmarried
it_kids = 1; % 1 = kids is zero
it_educ = 1; % 1 = lower education
% Select: NaN(n_jgrid,n_agrid,n_etagrid,n_educgrid,n_marriedgrid,n_kidsgrid);
mn_C_W_gain_check_jemk = mn_C_W_gain_check_use(it_age, :, 1:mp_params('n_eta_H_grid'), it_educ,
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);
```

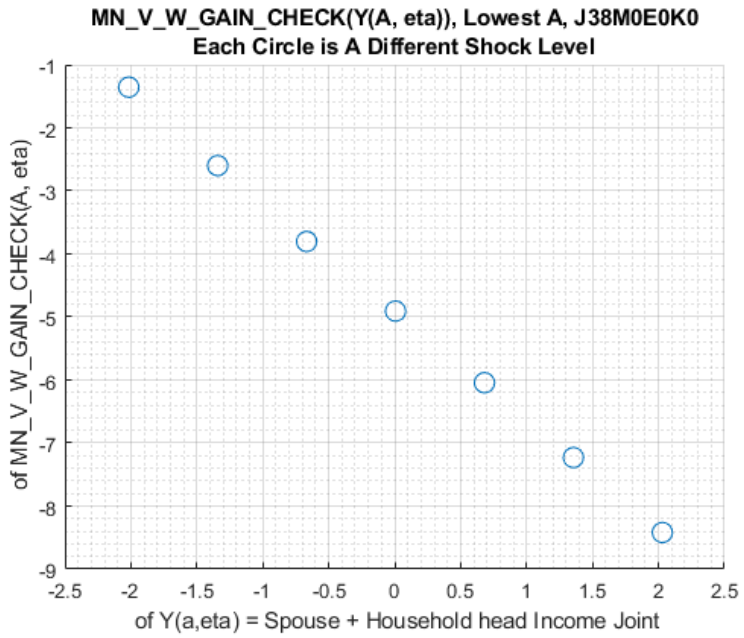
Marginal Value Gains, Color as Shock, Conditional on Age, Marital, Kids, and Education

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;
```

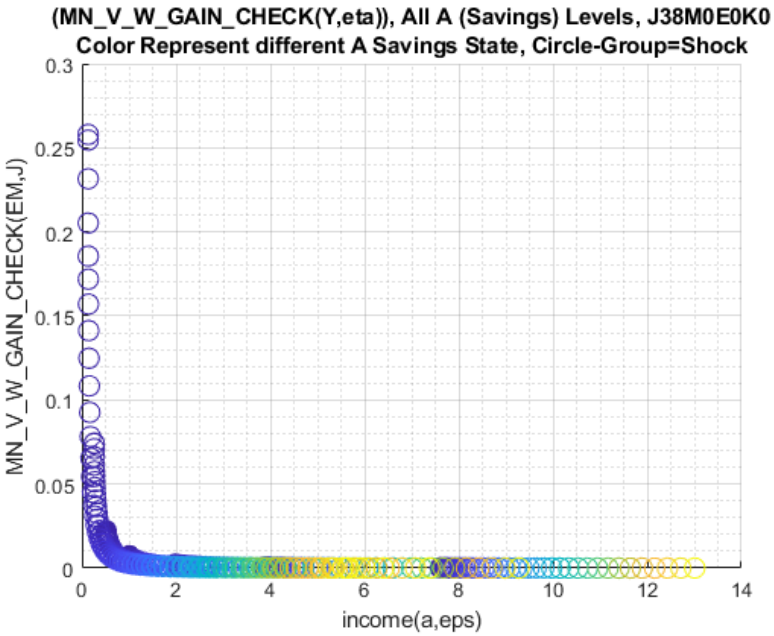



```
figure();
it_shock = 1;
scatter(log(mt_total_inc_jemk(:,it_a)), log(mt_V_W_gain_check_jemk(:,it_a)), 100);
title({'MN\_V\_W\_GAIN\_CHECK(Y(A, eta)), Lowest A, J38M0E0K0', ...
    'Each Circle is A Different Shock Level'});
xlabel(' of Y(a,eta) = Spouse + Household head Income Joint');
ylabel(' of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



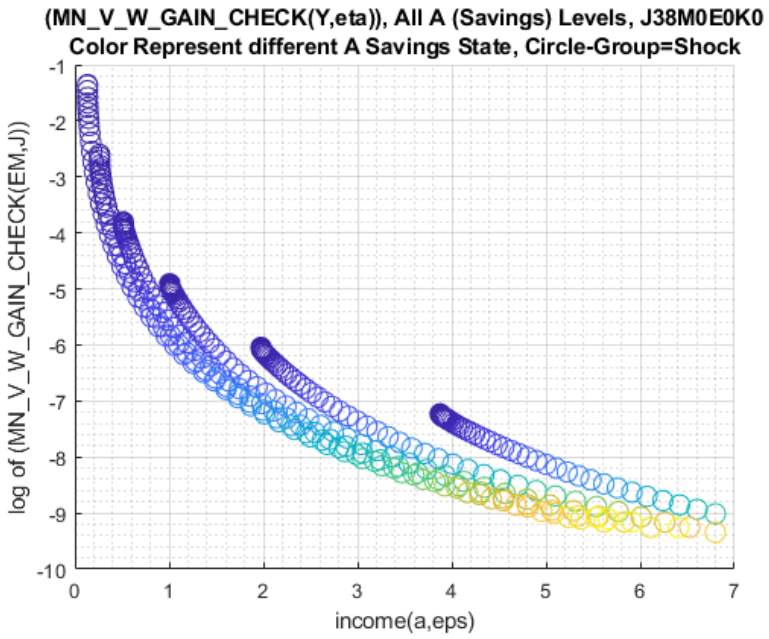
Plot all asset levels:

```
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', ...
```

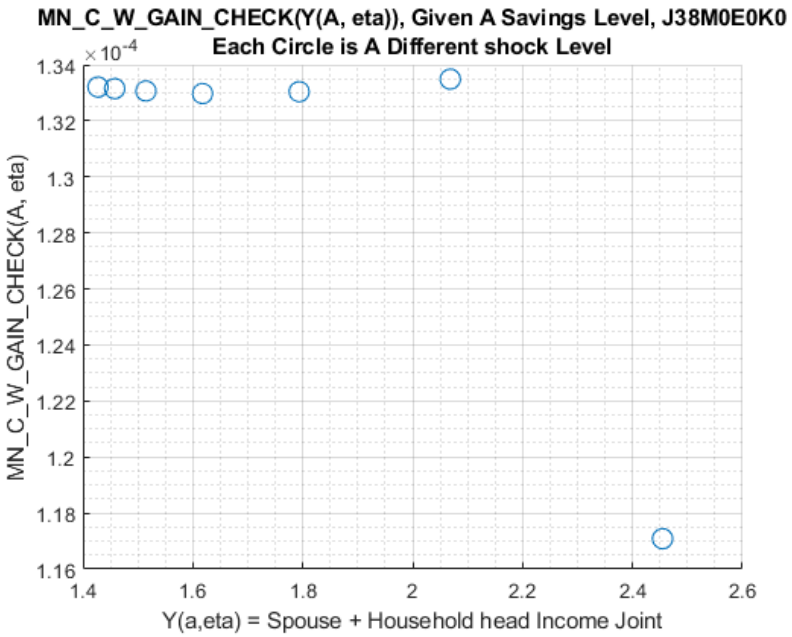
```
    'Color Represent different A Savings State, Circle-Group=Shock'});
xlabel('income(a,eps)');
ylabel('log of (MN\_V\_W\_GAIN\_CHECK(EM,J))');
xlim([0,7]);
grid on;
grid minor;
```



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

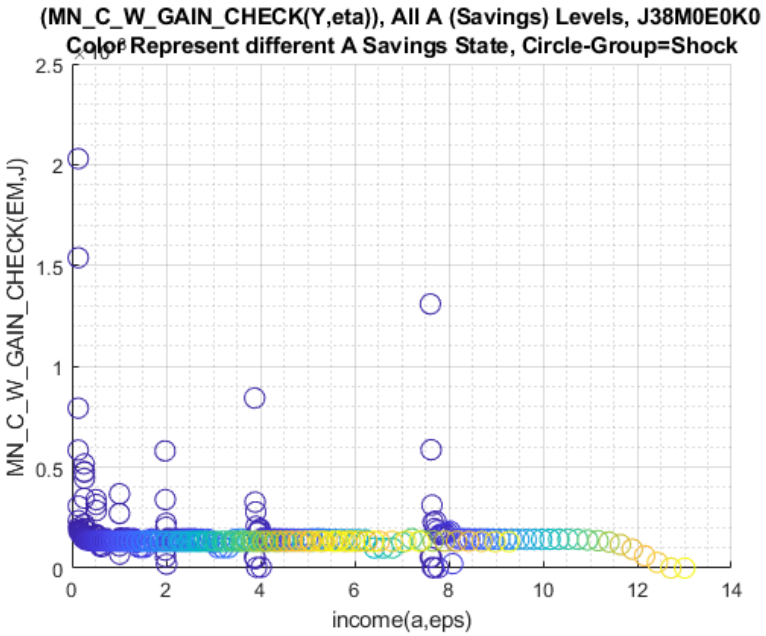
How do shocks and a impact marginal value. First plot one asset level, variation comes only from increasingly higher shocks:

```
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;
```



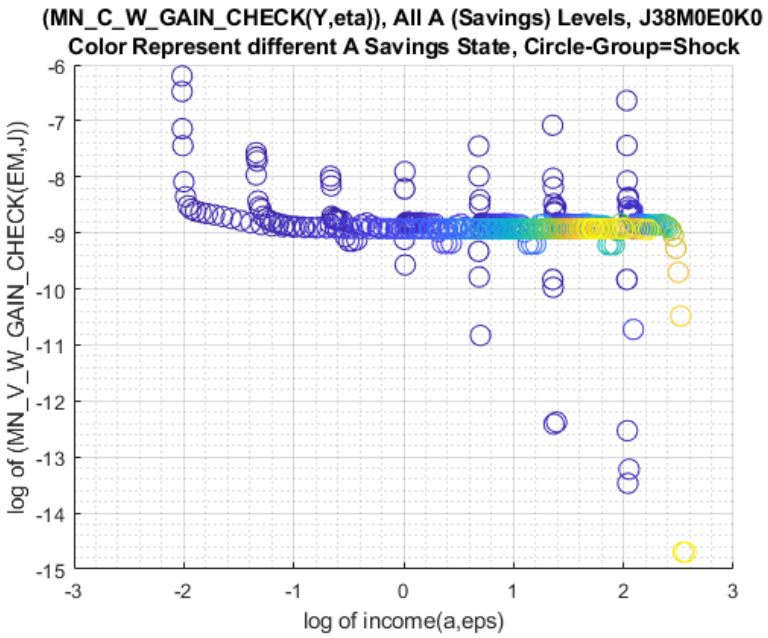
Plot all asset levels:

```
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	0.018089	0.017252	0.016367	0.015089	0.014021	0.01312
2	2	0	0.024167	0.023077	0.021867	0.020077	0.018573	0.017296
3	3	0	0.027732	0.026669	0.025441	0.023362	0.021616	0.020134
4	4	0	0.031193	0.03009	0.028767	0.026421	0.024452	0.022781
5	5	0	0.033947	0.032875	0.031534	0.02899	0.026856	0.025048
6	1	1	0.0061943	0.0059191	0.0056503	0.0051336	0.0047002	0.0043309
7	2	1	0.0081386	0.0077695	0.007405	0.0067275	0.0061552	0.0056679
8	3	1	0.0095966	0.0091734	0.0087634	0.0079591	0.0072803	0.006712
9	4	1	0.011304	0.010815	0.010337	0.0093981	0.0086038	0.0079341
10	5	1	0.013294	0.012771	0.012229	0.011142	0.010228	0.009448

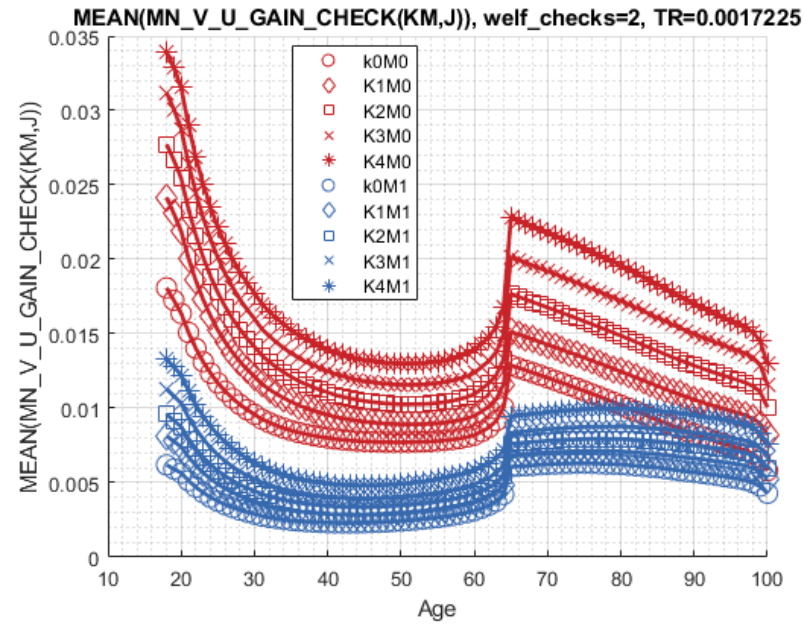
```
% Consumption Function
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(KM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2st
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check, true, ["mean"], 3, 1, cl_mp_dat
```

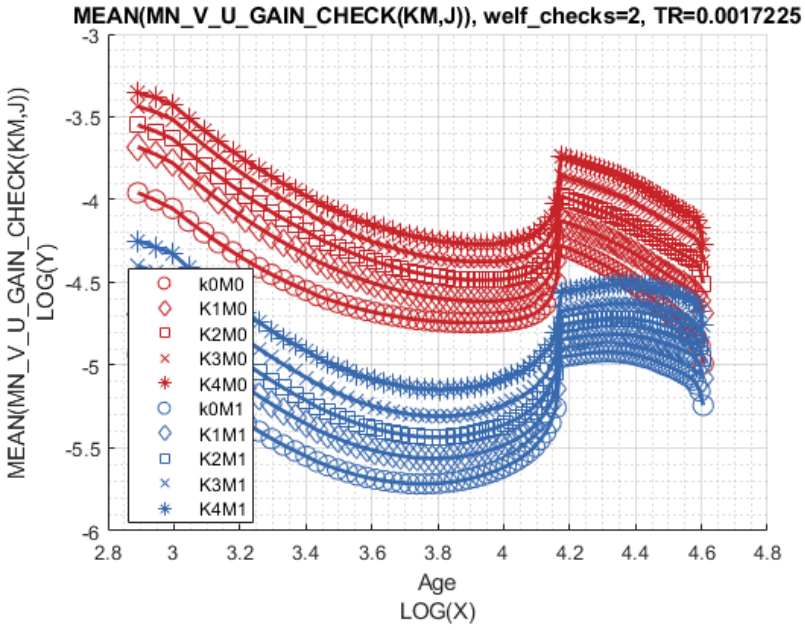
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(KM,J)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	kids	marry	mean_age_18	mean_age_19	mean_age_20	mean_age_21	mean_age_22	mean_age_23
1	1	0	0.064794	0.073581	0.10118	0.097925	0.093502	0.08852
2	2	0	0.072702	0.085068	0.11591	0.11391	0.11124	0.10979
3	3	0	0.087366	0.10442	0.13386	0.13266	0.1309	0.12783
4	4	0	0.088926	0.10344	0.13895	0.138	0.13683	0.13527
5	5	0	0.10276	0.11528	0.14404	0.14305	0.14187	0.14028
6	1	1	0.098235	0.10626	0.11555	0.11505	0.11337	0.11228
7	2	1	0.10346	0.10657	0.11743	0.11639	0.11571	0.1135
8	3	1	0.10975	0.11642	0.12686	0.12437	0.12447	0.12303
9	4	1	0.11043	0.11635	0.12843	0.12825	0.13154	0.1285
10	5	1	0.12035	0.12608	0.13427	0.136	0.13418	0.13489

Graph Mean Values:

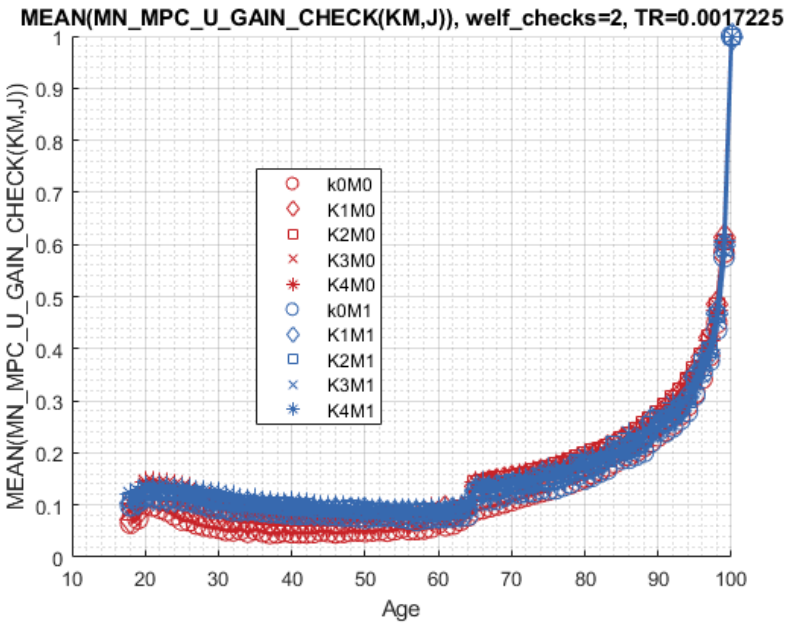
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(KM,J)), welf\_checks=' num2str(welf_checks) ', TR=' num
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(KM,J))'};
ff_graph_grid((tb_az_v{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);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

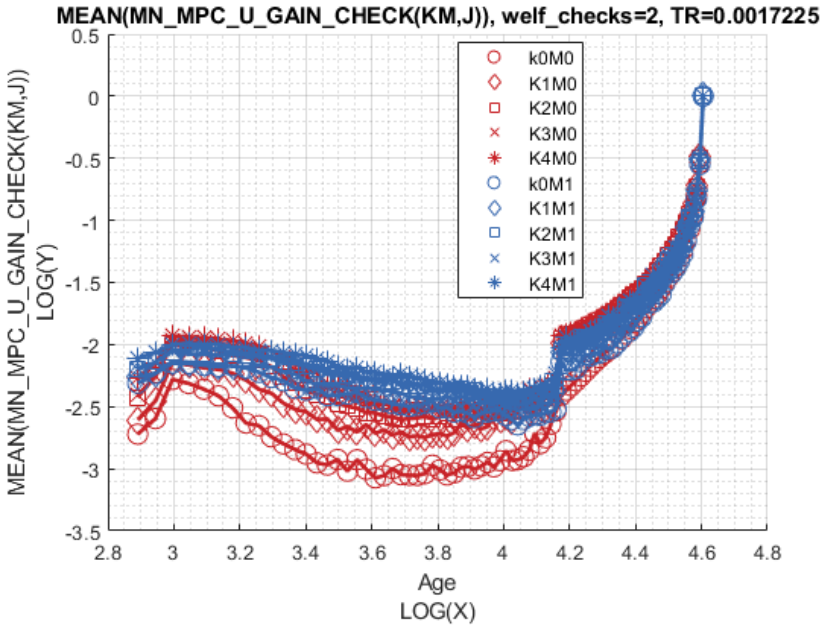




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
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          0.0284       0.027529       0.026552       0.025115       0.023836       0.022692
      2       1       0        0.025651       0.024457       0.023039       0.020461       0.018371       0.016659
      3       0       1         0.01032      0.0099016      0.0094945      0.0088461      0.0082733      0.0077704
      4       1       1       0.0090905      0.0086775      0.0082596      0.0072981      0.0065136      0.0058668
```

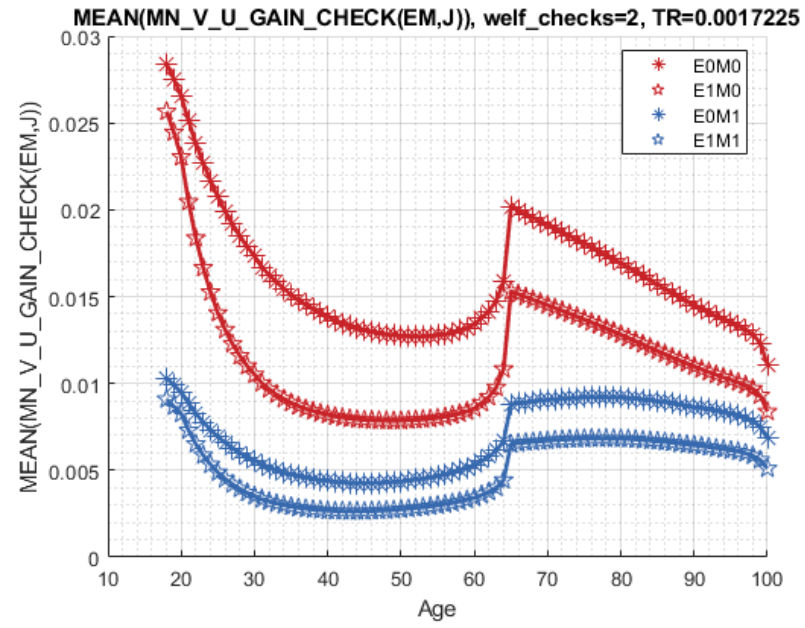
```
% Consumption
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2st
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check, true, ["mean"], 3, 1, cl_mp_dat
```

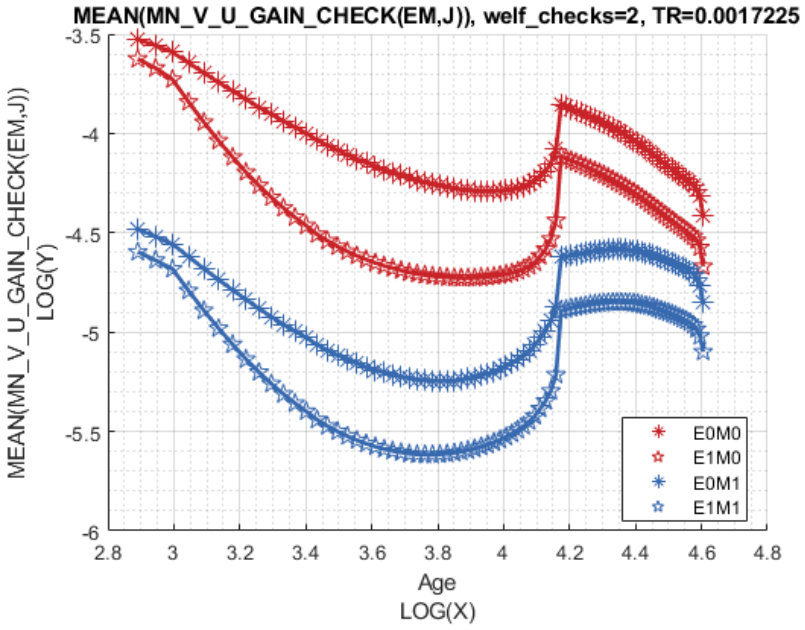
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0017225  xxxxxxxxxxxxxxxxxxxxxxxxxxx
    group    edu    marry    mean_age_18    mean_age_19    mean_age_20    mean_age_21    mean_age_22    mean_age_23
    _____    ___    _____    ___________    ___________    ___________    ___________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

      1       0       0         0.073692       0.083211         0.1068        0.10496        0.10269        0.10155
      2       1       0         0.092928        0.10951        0.14677        0.14526        0.14305        0.13912
      3       0       1          0.10086         0.1066        0.11448        0.11496        0.11372         0.1132
      4       1       1          0.11603        0.12207        0.13453        0.13306        0.13399        0.13168
```

Graph Mean Values:

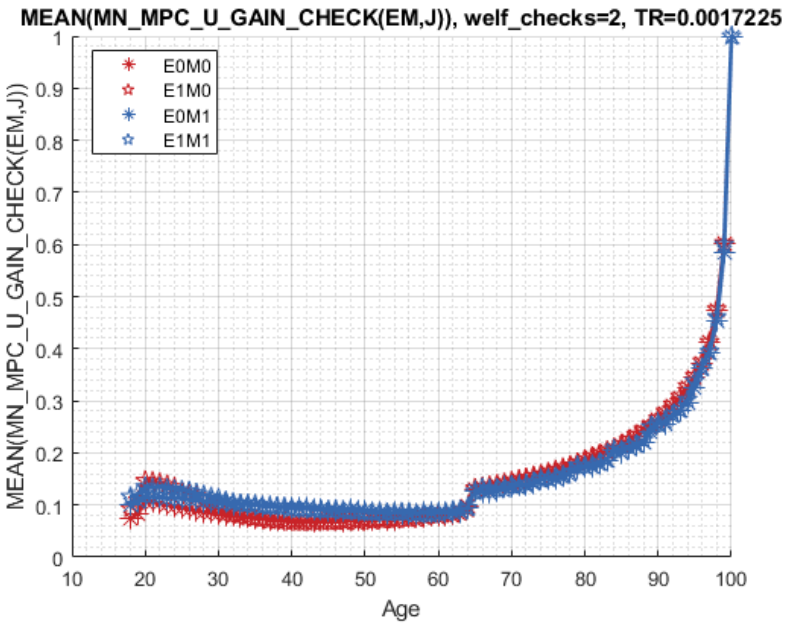
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(EM,J)), welf\_checks=' num2str(welf_checks) ', TR=' num
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(EM,J))'};
ff_graph_grid((tb_az_v{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);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

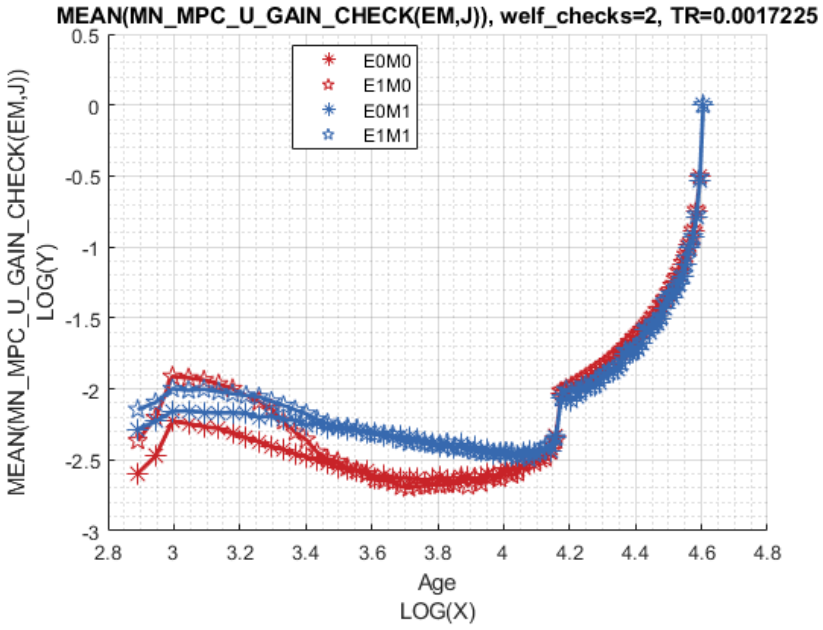




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
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)