2007 (Bush 2008 Stimulus) Full States EV and EC of Two Checks

This is the example vignette for function: **snw_evuvw19_jaeemk_foc** from the **PrjOptiSNW Package.** 2008 integrated over VU and VW, given optimal savings choices, unemployment shocks and various expectations.

Given 2008 policy and value functions (given expectation of 2009 crisis unemployment shocks), call SNW_V08_JAEEMK to solve for value and consumption given stimulus checks. And then integrate over 08 JAEEMK states given 07 JAEEMK states (age, endogenous savings, education, income shock, marital status, kids count). The stimulus will be provisioned based on 07 JAEEMK states. Note that **snw_evuvw19_jaeemk** does not solve the 07/08 problem.

Despite the name, this function supports solving the 2019 looking into 2020 as well as the 2007 looking into 2008 problems. The idea is that the planner only has information from 2019 and from 2007, and must allocate using those information. Stimulus, however, is given in 2020 and in 2008. So the planner needs to consider expected values in consumption or welfare given the transition probabilities of states in 2007 to 2008 and in 2019 to 2020. The **snw_evuvw19_jmky** file then aggregates the full state-space results to just JMKY state-space, which is the extend of information available to the planner.

Test SNW_EVUVW19_JAEEMK Defaults for 2019

Call the function with defaults. First, set up some parameters.

```
clear all;
% Solution types
st_biden_or_trump = 'bushchck';

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

% Solve for Unemployment Values
mp_controls('bl_print_a4chk') = false;
mp_controls('bl_print_a4chk_verbose') = false;
mp_controls('bl_print_vfi') = false;
mp_controls('bl_print_vfi_verbose') = false;
mp_controls('bl_print_ds') = false;
mp_controls('bl_print_ds_verbose') = false;
mp_controls('bl_print_precompute') = false;
mp_controls('bl_print_evuvw20_jaeemk') = false;
mp_controls('bl_print_evuvw20_jaeemk_verbose') = false;
mp_controls('bl_print_v08p08_jaeemk') = false;
mp_controls('bl_print_v08p08_jaeemk_verbose') = false;
mp_controls('bl_print_v08_jaeemk') = true;
mp_controls('bl_print_v08_jaeemk_verbose') = false;
```

Second, run initializing functions.

```
% 1. generate MP_PARAMS specific to 2008 stimulus
% Use non-default values for Bush Stimulus
mp_more_inputs = containers.Map('KeyType','char', 'ValueType','any');
mp_more_inputs('fl_ss_non_college') = 0.225;
mp_more_inputs('fl_ss_college') = 0.271;
```

```
fl_p50_hh_income_07 = 54831;
mp_more_inputs('fl_scaleconvertor') = fl_p50_hh_income_07;
% st_param_group = 'default_small';
st_param_group = 'default_dense';
st_param_group = 'default_docdense';
mp_params = snw_mp_param(st_param_group, false, 'tauchen', false, 8, 8, mp_more_inputs);
mp_params('st_biden_or_trump') = st_biden_or_trump;
mp_params('beta') = 0.95;
% 2. Solve value steady state (2009 employed)
[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_docdense;SNW_MP_CONTROL=default_test;time=495.65
```

```
V_emp_2009 = V_ss;
inc_ss = mp_valpol_more_ss('inc_VFI');
spouse_inc_ss = mp_valpol_more_ss('spouse_inc_VFI');
total_inc_ss = inc_ss + spouse_inc_ss;
% 3. Solve value unemployed 2009
mp_params('xi') = 0.532;
mp_params('b') = 0.37992;
mp_params('a2_covidyr') = mp_params('a2_greatrecession_2009');
mp_params('TR') = 100/fl_p50_hh_income_07;
[V_unemp_2009] = 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_docdense;SNW_MP_CONTROL=default_test;time
```

```
% 4. Value and Optimal choice in 2008
[V_2008, ap_2008, cons_2008, ev_empshk_2009] = ...
    snw_v08p08_jaeemk(mp_params, mp_controls, V_emp_2009, V_unemp_2009);
```

```
Completed SNW_VFI_MAIN_BISEC_VEC 1 Period Unemp Shock;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_test;time
Completed SNW_V08P08_JAEEMK;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_test;time=497.3876
```

```
% 5. matrixes to pre-compute
% Only using the SNW_A4CHK_WRK_BISEC_VEC function, no unemployment
% related matrixes needed Also don't need REF_EARN_WAGEIND_GRID,
% become unemployment not conditional on wage in 2009.
cl_st_precompute_list = {'a', ...
    'inc', 'inc_unemp', 'spouse_inc',...
    'ar_z_ctr_amz'};
% Shared: Steady-State distribution
[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_docdense;SNW_MP_CONTROL=default_test;time=1775.4241
```

```

% Shared: precompute, get Matrixes
% note, the mp_params inputs are based on unemployed in 2020 (MIT) or unemployed in 2009 (Expec
% note, however, for the 2008/9 problem, only will use inc, inc_unemp, spouse_inc
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=0.4645      0.71528      1.0335      1.5632
Completed SNW_HH_PRECOMPUTE;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_test;time cost=254.7405
```

Solve for 2019 Evuvw With 0 and 2 Checks

Solve for 0 and 2 checks, by finding the increase to asset state-space that is equivalent to the check increase, so that the problem can be solved without increasing the state-space.

```
% Call Function
welf_checks = 0;
[ev07_jaeemk_check0, ec07_jaeemk_check0, ev08_jaeemk_check0, ec08_jaeemk_check0] = snw_evuvw19_
    welf_checks, mp_params, mp_controls, ...
    ap_ss, V_2008, cons_2008, mp_precompute_res);
```

```
Solve for V_2008_check for 0 stimulus checks
Completed SNW_A4CHK_WRK_BISEC_VEC;SNW_MP_PARAM=bushchck;welf_checks=0;TR=0.0018238;SNW_MP_PARAM=default_docdense;SNW
Completed SNW_V08_JAEEMK;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_test;timeEUEC=6.99e-05
Completed SNW_EVUVW19_JAEEMK_FOC;st_biden_or_trump=bushchck;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_tes
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_outcomes ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                   i    idx    ndim      numel       rowN      colN          sum          mean        std       coefva
                   _    ___    ____    __________    ____    _________    ___________    _______    ______    ______

    ec07_jaeemk    1     1      6      4.3173e+07     82     5.265e+05     1.9685e+08     4.5597    5.3244     1.167
    ec08_jaeemk    2     2      6        4.37e+07     83     5.265e+05     2.3277e+08     5.3267    8.4419     1.584
    ev07_jaeemk    3     3      6      4.3173e+07     82     5.265e+05    -6.4618e+08    -14.967     21.06    -1.407
    ev08_jaeemk    4     4      6        4.37e+07     83     5.265e+05    -6.6426e+08    -15.201     21.85    -1.437

xxx TABLE:ec07_jaeemk xxxxxxxxxxxxxxxxxx
              c1          c2          c3          c4          c5       c526496    c526497    c526498    c526499    c
           ________    ________    ________    ________    ________    _______    _______    _______    _______    _

    r1     0.039543    0.039543    0.039978    0.042271     0.04854    12.009     12.281     12.561     12.849     1
    r2     0.039889    0.039889    0.040323    0.043305    0.049735    12.251     12.524     12.806     13.095     1
    r3     0.041432    0.041432    0.041608    0.043991    0.050734    12.485     12.759     13.042     13.331     1
    r4     0.042935    0.042935    0.043023    0.045459    0.052199    12.742     13.017       13.3     13.588     1
    r5     0.044395    0.044395    0.044399    0.046895    0.053615     12.99     13.266     13.548     13.834     1
    r78      0.2016      0.2016      0.2016      0.2016     0.20214    27.775     28.774     29.785     30.965     3
    r79      0.2016      0.2016      0.2016      0.2016      0.2016     30.43     31.663     32.736     33.958     3
    r80      0.2016      0.2016      0.2016      0.2016      0.2016     33.68     35.501     37.368     38.956     4
    r81      0.2016      0.2016      0.2016      0.2016      0.2016    40.118     41.397     43.175     45.605     4
    r82      0.2016      0.2016      0.2016      0.2016      0.2016      52.1     55.541     58.464     60.094     6

xxx TABLE:ec08_jaeemk xxxxxxxxxxxxxxxxxx
              c1          c2          c3          c4          c5       c526496    c526497    c526498    c526499    c
           ________    ________    ________    ________    ________    _______    _______    _______    _______    _

    r1     0.036218    0.036736    0.038184    0.042735    0.048545    12.256     12.541     12.835     13.136     1
    r2     0.036271    0.036736    0.038385    0.043404    0.049852    12.491     12.778     13.072     13.374     1
    r3     0.036717    0.037251    0.039845    0.044907    0.051515    12.744     13.032     13.327     13.628     1
    r4     0.038144    0.038678    0.041269    0.046371    0.053128    12.989     13.277     13.573     13.872     1
    r5     0.039534    0.040068    0.042653    0.047793    0.054687    13.224     13.513     13.809     14.105     1
    r79      0.2016     0.20214     0.20586     0.21598     0.23568     35.82     37.367     39.414     41.705     4
    r80      0.2016     0.20214     0.20586     0.21598     0.23568    40.755     42.955     45.289      47.95     5
    r81      0.2016     0.20214     0.20586     0.21598     0.23568    48.912     52.041     55.022     57.919     6
    r82      0.2016     0.20214     0.20586     0.21598     0.23568    66.719     69.201     72.373     77.005     8
    r83      0.2016     0.20214     0.20586     0.21598     0.23568    116.83     122.65     128.67     134.89

xxx TABLE:ev07_jaeemk xxxxxxxxxxxxxxxxxx
             c1         c2         c3         c4         c5        c526496     c526497     c526498     c526499
           _______    _______    _______    _______    _______    _________    _________    _________    _________

    r1     -282.59    -282.59    -282.23    -278.38     -270.8     -4.3833      -4.2888      -4.1954      -4.1032
    r2     -272.49    -272.49    -272.13    -268.93    -261.76     -4.2754      -4.1844      -4.0945      -4.0056
```

r3	-262.36	-262.36	-262.23	-259.89	-253.05	-4.1651	-4.0777	-3.9913	-3.9057
r4	-253.22	-253.22	-253.16	-250.92	-244.6	-4.0512	-3.9675	-3.8846	-3.8025
r5	-244.95	-244.95	-244.95	-242.81	-236.93	-3.9436	-3.8633	-3.7837	-3.7047
r78	-13.362	-13.362	-13.362	-13.362	-13.349	-0.27313	-0.26104	-0.24971	-0.2389
r79	-12.032	-12.032	-12.032	-12.032	-12.032	-0.21855	-0.20781	-0.19875	-0.19057
r80	-10.388	-10.388	-10.388	-10.388	-10.388	-0.16126	-0.15407	-0.14734	-0.14134
r81	-8.1801	-8.1801	-8.1801	-8.1801	-8.1801	-0.10114	-0.097396	-0.093438	-0.089426
r82	-4.9651	-4.9651	-4.9651	-4.9651	-4.9651	-0.044201	-0.041462	-0.039412	-0.038348

xxx TABLE:ev08_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c526496**	**c526497**	**c526498**	**c526499**
r1	-295.66	-295.26	-292.66	-286.62	-277.22	-4.3615	-4.2673	-4.1741	-4.0822
r2	-286.11	-285.71	-283.12	-277.16	-268.03	-4.2548	-4.1641	-4.0744	-3.9858
r3	-276.49	-276.09	-273.59	-267.84	-259.11	-4.1461	-4.0589	-3.9727	-3.8874
r4	-266.77	-266.41	-264.08	-258.7	-250.49	-4.0342	-3.9507	-3.868	-3.7862
r5	-257.99	-257.65	-255.48	-250.43	-242.69	-3.9287	-3.8485	-3.769	-3.6903
r79	-13.356	-13.343	-13.253	-13.025	-12.638	-0.22088	-0.21055	-0.20083	-0.1917
r80	-12.025	-12.012	-11.923	-11.695	-11.308	-0.16977	-0.1618	-0.15428	-0.14721
r81	-10.382	-10.369	-10.28	-10.052	-9.6651	-0.11711	-0.11162	-0.10645	-0.10156
r82	-8.1742	-8.1611	-8.0716	-7.844	-7.457	-0.065329	-0.062239	-0.059357	-0.056632
r83	-4.9602	-4.9471	-4.8576	-4.6301	-4.2431	-0.020966	-0.019971	-0.019037	-0.01816

```
% Call Function
welf_checks = 2;
[ev07_jaeemk_check2, ec07_jaeemk_check2, ev08_jaeemk_check2, ec08_jaeemk_check2] = snw_evuvw19_
    welf_checks, mp_params, mp_controls, ...
    ap_ss, V_2008, cons_2008, mp_precompute_res);
```

```
Solve for V_2008_check for 2 stimulus checks
Completed SNW_A4CHK_WRK_BISEC_VEC;SNW_MP_PARAM=bushchck;welf_checks=2;TR=0.0018238;SNW_MP_PARAM=default_docdense;SNW
Completed SNW_V08_JAEEMK;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_test;timeEUEC=2.82e-05
Completed SNW_EVUVW19_JAEEMK_FOC;st_biden_or_trump=bushchck;SNW_MP_PARAM=default_docdense;SNW_MP_CONTROL=default_tes
----------------------------------------
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
CONTAINER NAME: mp_outcomes ND Array (Matrix etc)
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

	i	**idx**	**ndim**	**numel**	**rowN**	**colN**	**sum**	**mean**	**std**	**coefva**
ec07_jaeemk	1	1	6	4.3173e+07	82	5.265e+05	1.9688e+08	4.5603	5.3245	1.167
ec08_jaeemk	2	2	6	4.37e+07	83	5.265e+05	2.328e+08	5.3273	8.442	1.584
ev07_jaeemk	3	3	6	4.3173e+07	82	5.265e+05	-6.4561e+08	-14.954	21.011	-1.405
ev08_jaeemk	4	4	6	4.37e+07	83	5.265e+05	-6.6365e+08	-15.187	21.798	-1.435

xxx TABLE:ec07_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c526496**	**c526497**	**c526498**	**c526499**	**c**
r1	0.040827	0.040827	0.040972	0.04389	0.049551	12.009	12.281	12.561	12.849	1
r2	0.042096	0.042096	0.0424	0.045024	0.050802	12.251	12.524	12.806	13.095	1
r3	0.043624	0.043624	0.043746	0.045821	0.051869	12.485	12.76	13.042	13.331	1
r4	0.04511	0.04511	0.04517	0.047304	0.053361	12.742	13.017	13.3	13.588	1
r5	0.04655	0.04655	0.046553	0.048751	0.054802	12.99	13.266	13.549	13.834	1
r78	0.20525	0.20525	0.20525	0.20525	0.20579	27.775	28.775	29.786	30.966	3
r79	0.20525	0.20525	0.20525	0.20525	0.20525	30.431	31.664	32.737	33.96	3
r80	0.20525	0.20525	0.20525	0.20525	0.20525	33.681	35.503	37.369	38.957	4
r81	0.20525	0.20525	0.20525	0.20525	0.20525	40.119	41.4	43.178	45.607	
r82	0.20525	0.20525	0.20525	0.20525	0.20525	52.103	55.545	58.468	60.098	6

xxx TABLE:ec08_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c526496**	**c526497**	**c526498**	**c526499**	**c**

r1	0.037941	0.038148	0.039819	0.043807	0.049244	12.256	12.541	12.835	13.136	1
r2	0.038108	0.038344	0.040188	0.044594	0.050571	12.492	12.778	13.073	13.374	1
r3	0.03941	0.039781	0.041664	0.046126	0.052244	12.745	13.032	13.327	13.628	1
r4	0.040834	0.041205	0.043102	0.047618	0.053867	12.989	13.278	13.573	13.872	1
r5	0.04222	0.042589	0.0445	0.049065	0.055435	13.224	13.513	13.809	14.105	1
r79	0.20525	0.20579	0.20951	0.21963	0.23776	35.821	37.368	39.415	41.707	
r80	0.20525	0.20579	0.20951	0.21963	0.23776	40.756	42.957	45.29	47.952	5
r81	0.20525	0.20579	0.20951	0.21963	0.2378	48.914	52.043	55.024	57.92	6
r82	0.20525	0.20579	0.20951	0.21963	0.23814	66.72	69.203	72.375	77.007	8
r83	0.20525	0.20579	0.20951	0.21963	0.23933	116.84	122.66	128.68	134.89	1

xxx TABLE:ev07_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c526496**	**c526497**	**c526498**	**c526499**
r1	-280.23	-280.23	-279.88	-276.5	-269.35	-4.3833	-4.2887	-4.1954	-4.1031
r2	-270.21	-270.21	-269.89	-267.08	-260.35	-4.2753	-4.1844	-4.0944	-4.0055
r3	-260.26	-260.26	-260.13	-258.1	-251.71	-4.165	-4.0777	-3.9912	-3.9057
r4	-251.26	-251.26	-251.2	-249.25	-243.33	-4.0511	-3.9674	-3.8846	-3.8025
r5	-243.12	-243.12	-243.12	-241.24	-235.73	-3.9436	-3.8633	-3.7837	-3.7047
r78	-13.274	-13.274	-13.274	-13.274	-13.262	-0.27312	-0.26103	-0.2497	-0.23889
r79	-11.944	-11.944	-11.944	-11.944	-11.944	-0.21854	-0.2078	-0.19875	-0.19056
r80	-10.3	-10.3	-10.3	-10.3	-10.3	-0.16125	-0.15406	-0.14733	-0.14134
r81	-8.0921	-8.0921	-8.0921	-8.0921	-8.0921	-0.10113	-0.097391	-0.093433	-0.089423
r82	-4.8771	-4.8771	-4.8771	-4.8771	-4.8771	-0.044198	-0.04146	-0.03941	-0.038346

xxx TABLE:ev08_jaeemk xxxxxxxxxxxxxxxxxx

	c1	**c2**	**c3**	**c4**	**c5**	**c526496**	**c526497**	**c526498**	**c526499**
r1	-293.09	-292.72	-290.49	-284.88	-275.86	-4.3615	-4.2672	-4.1741	-4.0821
r2	-283.55	-283.18	-280.98	-275.47	-266.73	-4.2548	-4.164	-4.0743	-3.9857
r3	-274.01	-273.65	-271.52	-266.23	-257.88	-4.146	-4.0589	-3.9726	-3.8874
r4	-264.47	-264.13	-262.14	-257.19	-249.33	-4.0341	-3.9506	-3.8679	-3.7861
r5	-255.84	-255.53	-253.66	-249	-241.59	-3.9286	-3.8484	-3.769	-3.6903
r79	-13.268	-13.255	-13.171	-12.954	-12.578	-0.22088	-0.21054	-0.20082	-0.1917
r80	-11.937	-11.924	-11.841	-11.623	-11.248	-0.16976	-0.16179	-0.15428	-0.1472
r81	-10.294	-10.281	-10.198	-9.9804	-9.6057	-0.11711	-0.11162	-0.10644	-0.10156
r82	-8.0862	-8.0735	-7.9895	-7.7724	-7.3981	-0.065327	-0.062237	-0.059355	-0.056631
r83	-4.8723	-4.8595	-4.7755	-4.5584	-4.1854	-0.020965	-0.01997	-0.019036	-0.018159

Differences between Checks in Expected Value and Expected Consumption

```
mn_V_U_gain_check_07 = ev07_jaeemk_check2 - ev07_jaeemk_check0;
mn_MPC_U_gain_share_check_07 = (ec07_jaeemk_check2 - ec07_jaeemk_check0)./(welf_checks*mp_param
```

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_07, true, ["mean"], 4, 1, cl_mp_dataseto
```

```
xxx  MEAN(MN_V_U_GAIN_CHECK(A,Z)), welf_checks=2, TR=0.0018238   xxxxxxxxxxxxxxxxxxxxxxxxxxx
    group      savings       mean_eta_1    mean_eta_2    mean_eta_3    mean_eta_4    mean_eta_5    mean_eta_6    mean_
    _____    __________    __________    __________    __________    __________    __________    __________    _____

      1               0       0.52526       0.48873       0.44813       0.40715        0.3683       0.33276       0.
      2      0.00051498       0.52526       0.48873       0.44813       0.40715        0.3683       0.33276       0.
      3       0.0041199       0.52469       0.48792       0.44711       0.40616       0.36748        0.3321       0.
      4        0.013905       0.47091       0.43897       0.40471       0.37007         0.337       0.30644       0.
      5        0.032959       0.39774        0.3737       0.34722       0.32037       0.29449       0.27025       0.
      6        0.064373       0.32479       0.30724       0.28795       0.26815       0.24878       0.23037       0.
```

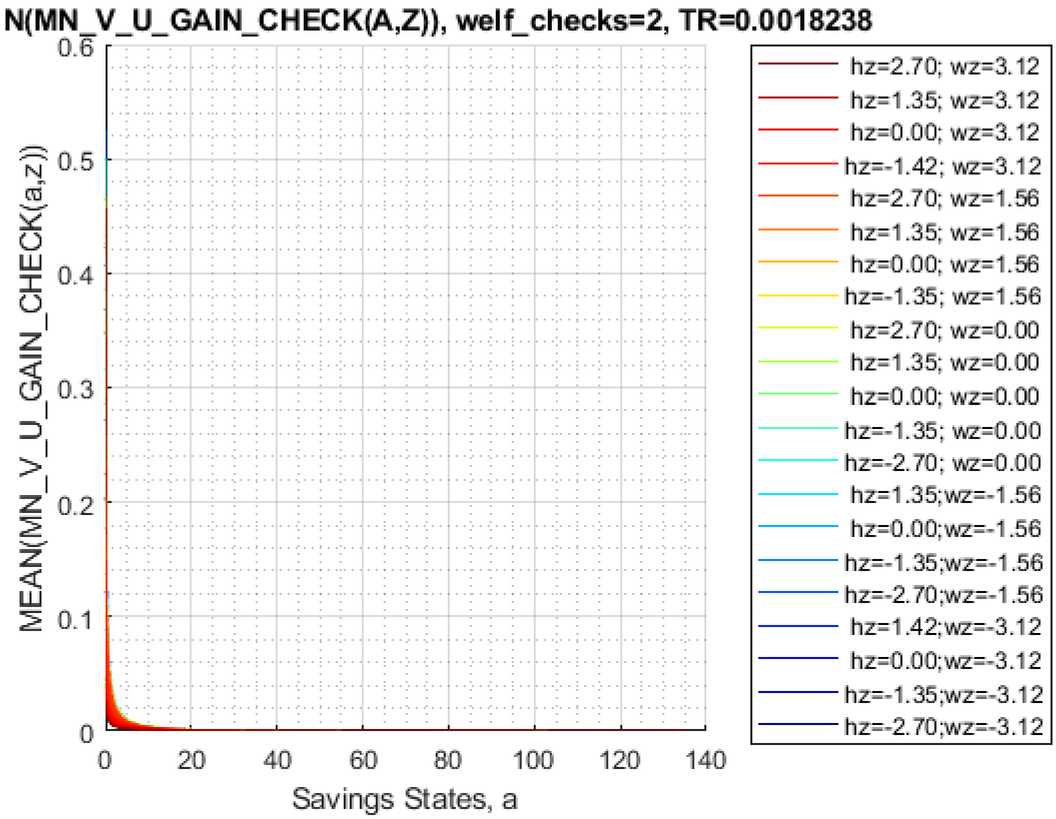
```
% Consumption
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(A,Z)), welf_checks=' num2str(welf_checks) ', TR=' num2str
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check_07, true, ["mean"], 4, 1, cl_mp_
```

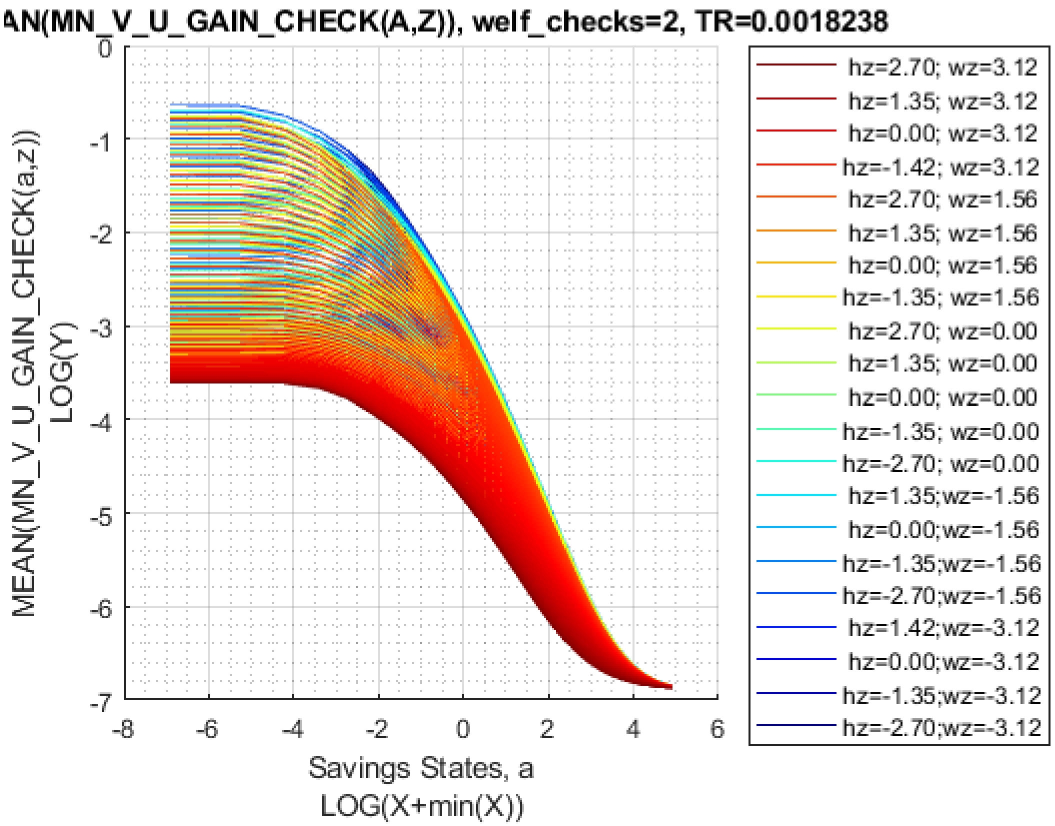
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(A,Z)), welf_checks=2, TR=0.0018238   xxxxxxxxxxxxxxxxxxxxxxxxxxx
    group      savings      mean_eta_1    mean_eta_2    mean_eta_3    mean_eta_4    mean_eta_5    mean_eta_6    mean_
    _____    __________    __________    __________    __________    __________    __________    __________    _____

      1               0      0.74182       0.72635       0.71164       0.69952       0.69052       0.68421       0.67
      2      0.00051498      0.74182       0.72635       0.71164       0.69952       0.69052       0.68421       0.67
      3       0.0041199      0.74131       0.72479       0.70912       0.69655       0.68793       0.68245       0.67
      4        0.013905      0.69437        0.6887       0.68682        0.6856       0.68485       0.68443       0.68
      5        0.032959      0.62728        0.6255       0.62438       0.62423       0.62492       0.62615       0.62
      6        0.064373      0.55114       0.54963       0.54908       0.54953       0.55084       0.55275       0.55
```

Graph Mean Values:

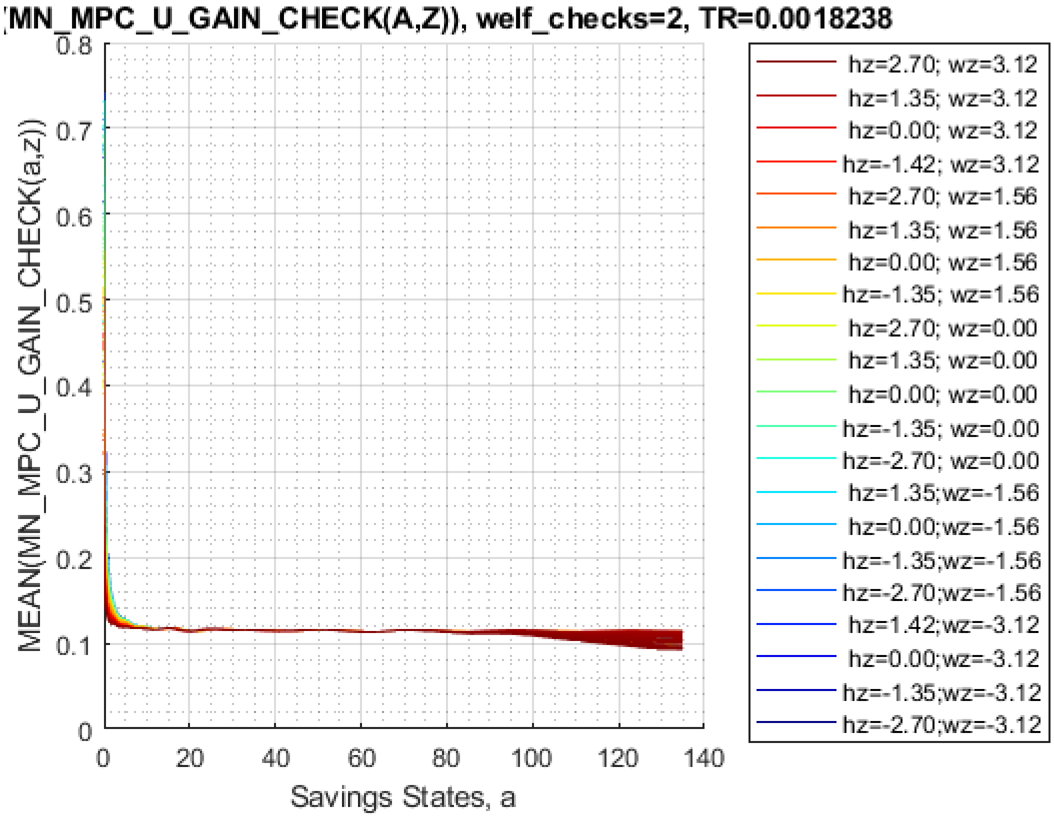
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(A,Z)), welf\_checks=' num2str(welf_checks) ', TR=' num2
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(a,z))'};
ff_graph_grid((tb_az_v{1:end, 3:end})', ar_st_eta_HS_grid, agrid, mp_support_graph);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

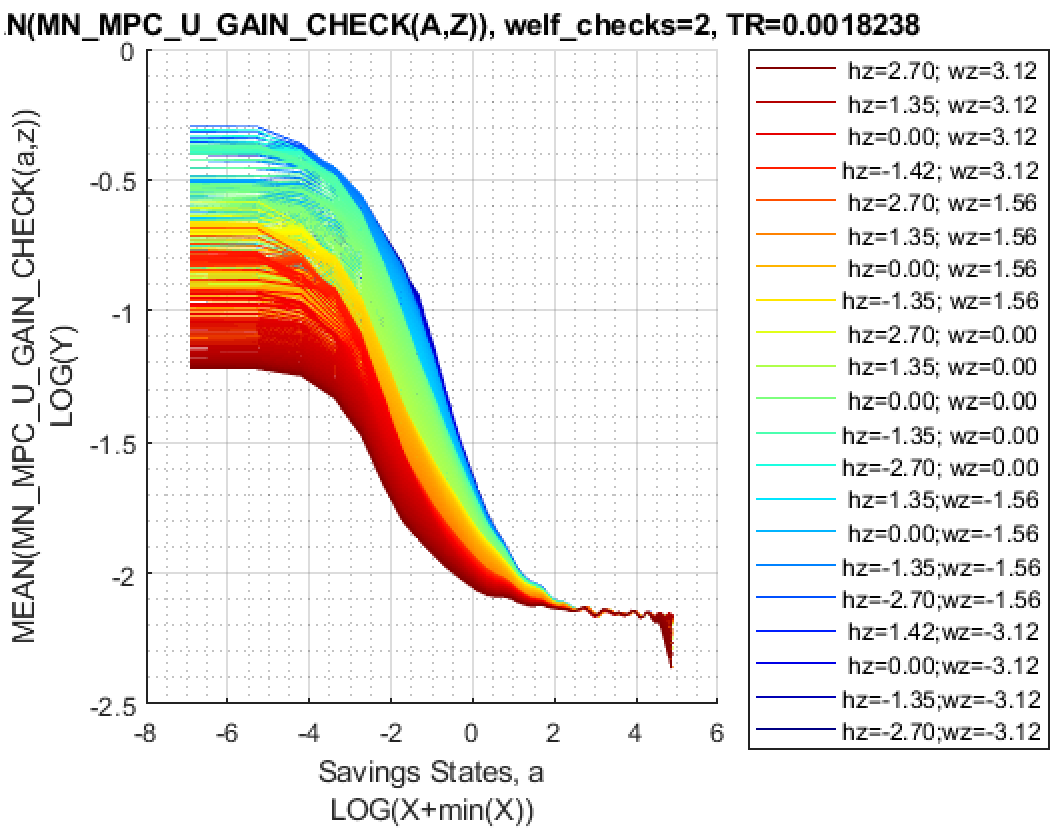




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
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_ss = total_inc_ss(19,:,1:mp_params('n_eta_H_grid'),1,1,1);
mn_V_W_gain_check_use_07 = ev07_jaeemk_check2 - ev07_jaeemk_check0;
mn_C_W_gain_check_use_07 = ec07_jaeemk_check2 - ec07_jaeemk_check0;
```

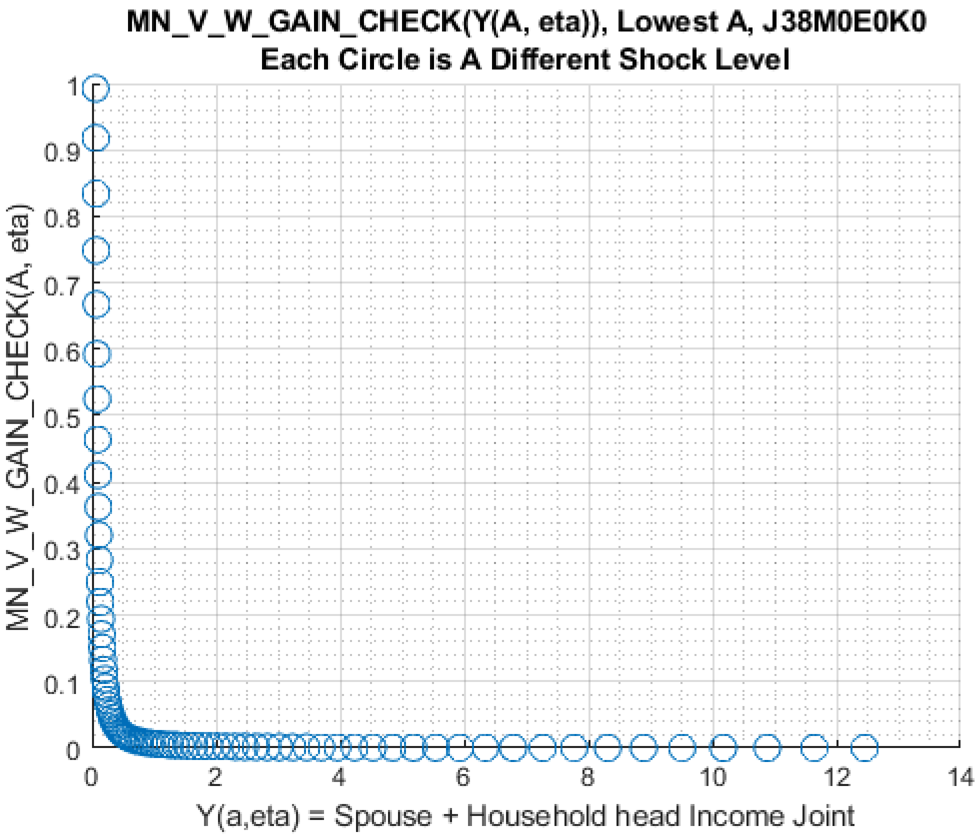
Select Age, Education, Marital, Kids Count:s

```
% 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_07 = mn_C_W_gain_check_use_07(it_age, :, 1:mp_params('n_eta_H_grid'), it
mn_V_W_gain_check_jemk_07 = mn_V_W_gain_check_use_07(it_age, :, 1:mp_params('n_eta_H_grid'), it
% Reshape, so shock is the first dim, a is the second
mt_total_inc_jemk = permute(mn_total_inc_jemk_ss,[3,2,1]);
mt_C_W_gain_check_jemk_07 = permute(mn_C_W_gain_check_jemk_07,[3,2,1]);
mt_C_W_gain_check_jemk_07(mt_C_W_gain_check_jemk_07<=1e-10) = 1e-10;
mt_V_W_gain_check_jemk_07 = permute(mn_V_W_gain_check_jemk_07,[3,2,1]);
mt_V_W_gain_check_jemk_07(mt_V_W_gain_check_jemk_07<=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);
```

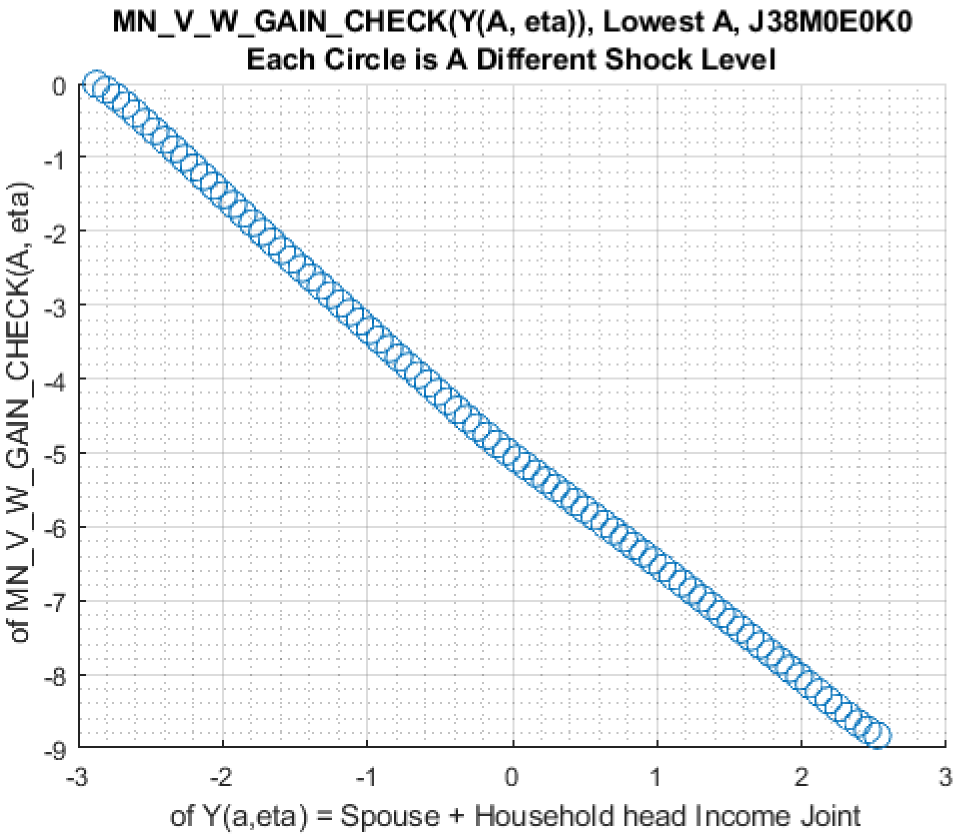
Marginal Value Gains, Color as Shock, Conditional on Age, Marital, Kids, and Education

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_07(:,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;
```

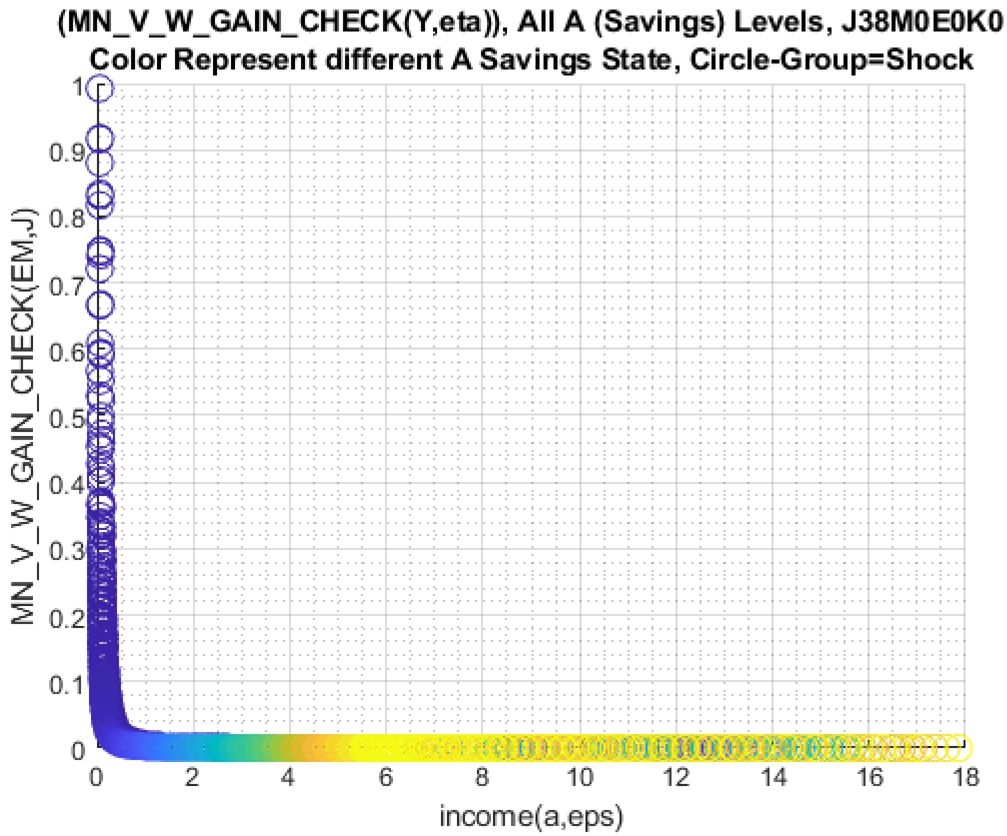



```
figure();
it_shock = 1;
scatter(log(mt_total_inc_jemk(:,it_a)), log(mt_V_W_gain_check_jemk_07(:,it_a)), 100);
title({'MN\_V\_W\_GAIN\_CHECK(Y(A, eta)), Lowest A, J38M0E0K0', ...
    'Each Circle is A Different Shock Level'});
xlabel(' of Y(a,eta) = Spouse + Household head Income Joint');
ylabel(' of MN\_V\_W\_GAIN\_CHECK(A, eta)');
grid on;
grid minor;
```

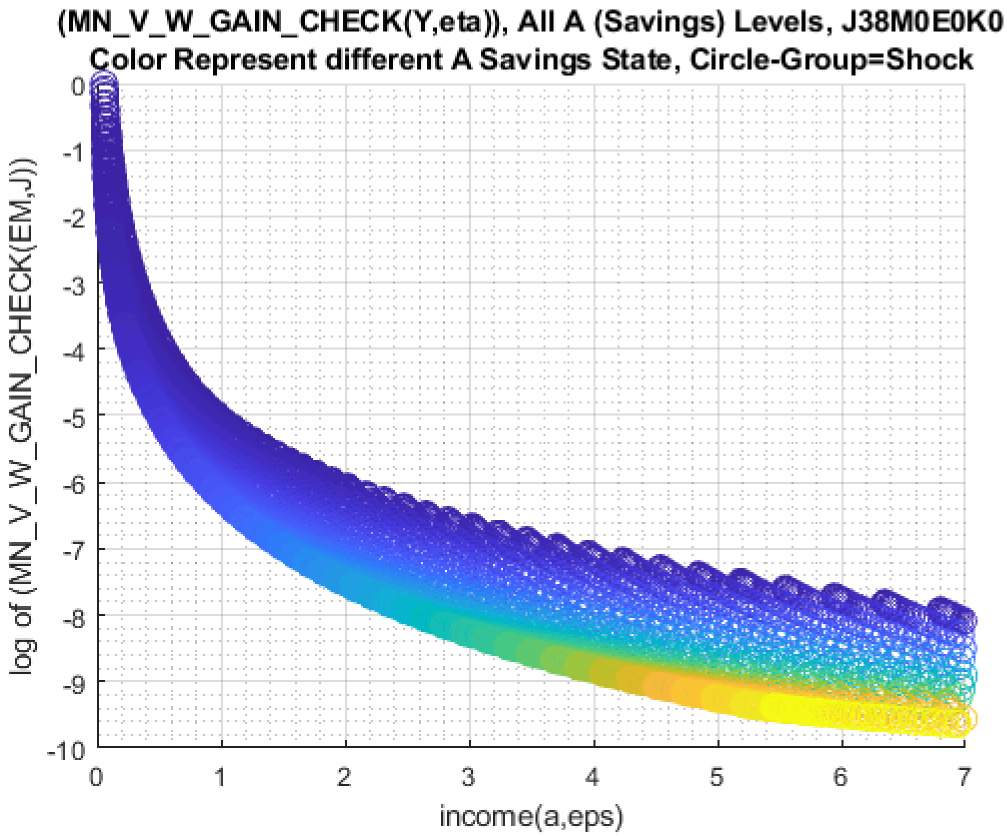


Plot all asset levels:

```
figure();
scatter((mt_total_inc_jemk(:)), (mt_V_W_gain_check_jemk_07(:)), 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;
```



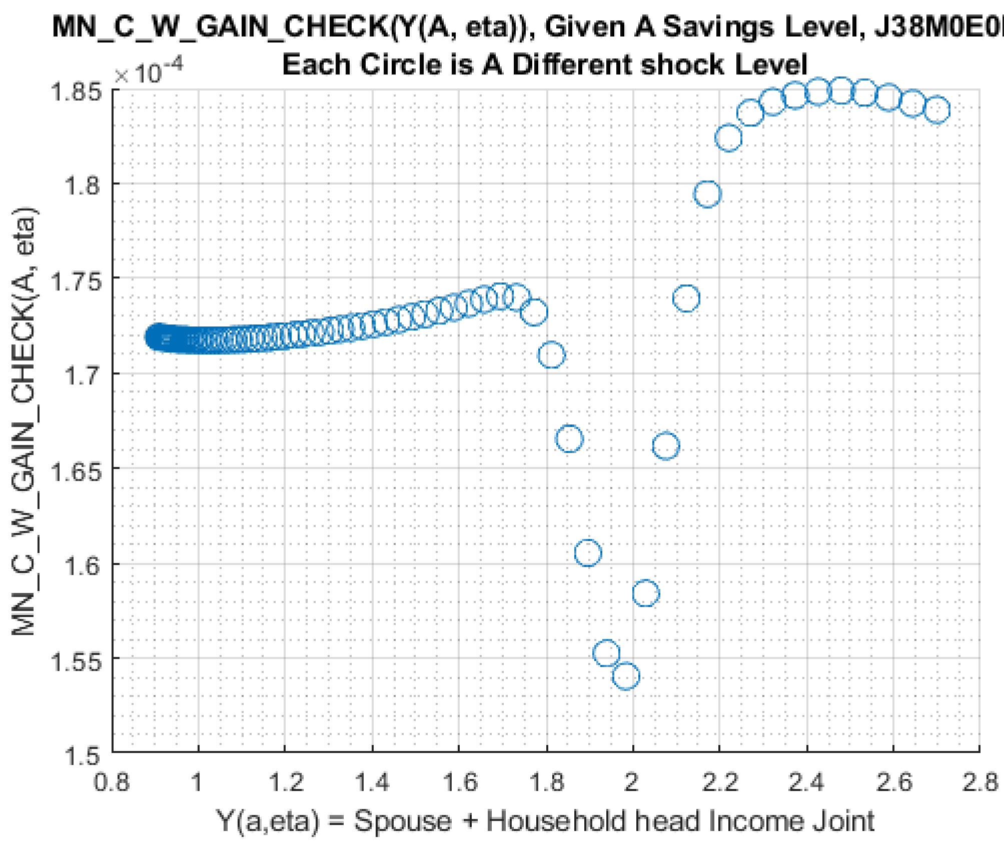

```
figure();
scatter((mt_total_inc_jemk(:)), log(mt_V_W_gain_check_jemk_07(:)), 100, mt_a(:));
title({'(MN\_V\_W\_GAIN\_CHECK(Y,eta)), All A (Savings) Levels, J38M0E0K0', ...
    'Color Represent different A Savings State, Circle-Group=Shock'});
xlabel('income(a,eps)');
ylabel('log of (MN\_V\_W\_GAIN\_CHECK(EM,J))');
xlim([0,7]);
grid on;
grid minor;
```



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

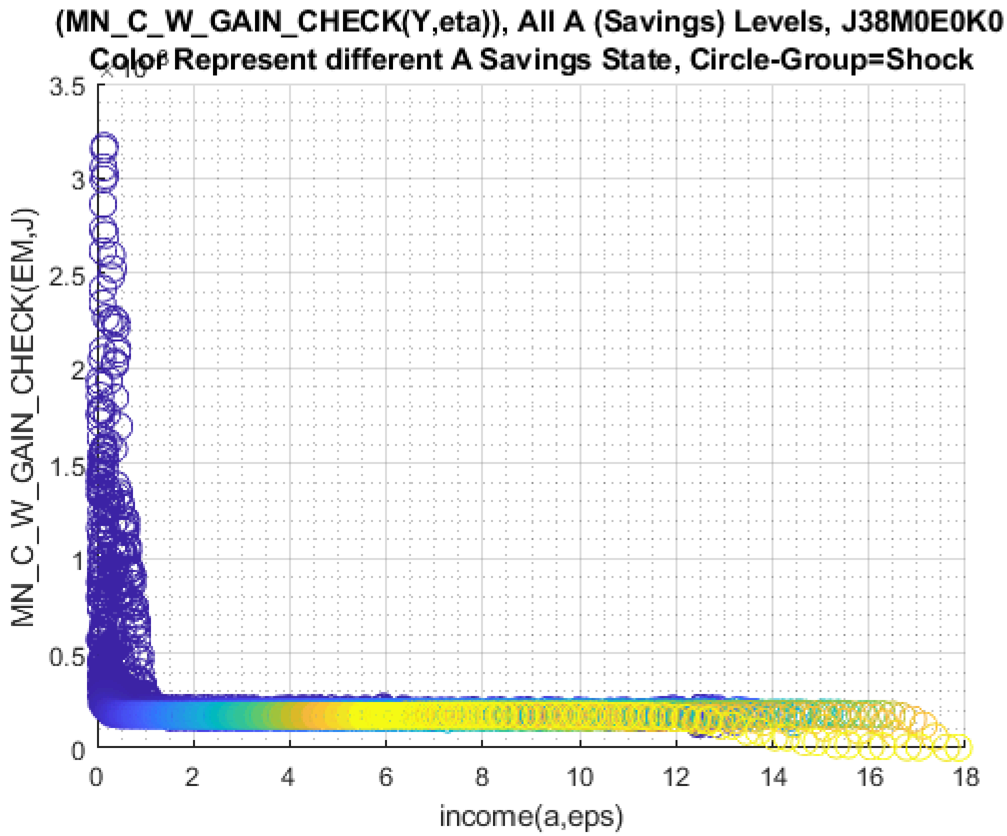
How do shocks and a impact marginal value. First plot one asset level, variation comes only from increasingly higher shocks:

```
figure();
it_a = 50;
scatter(log(mt_total_inc_jemk(:,it_a)), mt_C_W_gain_check_jemk_07(:,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;
```

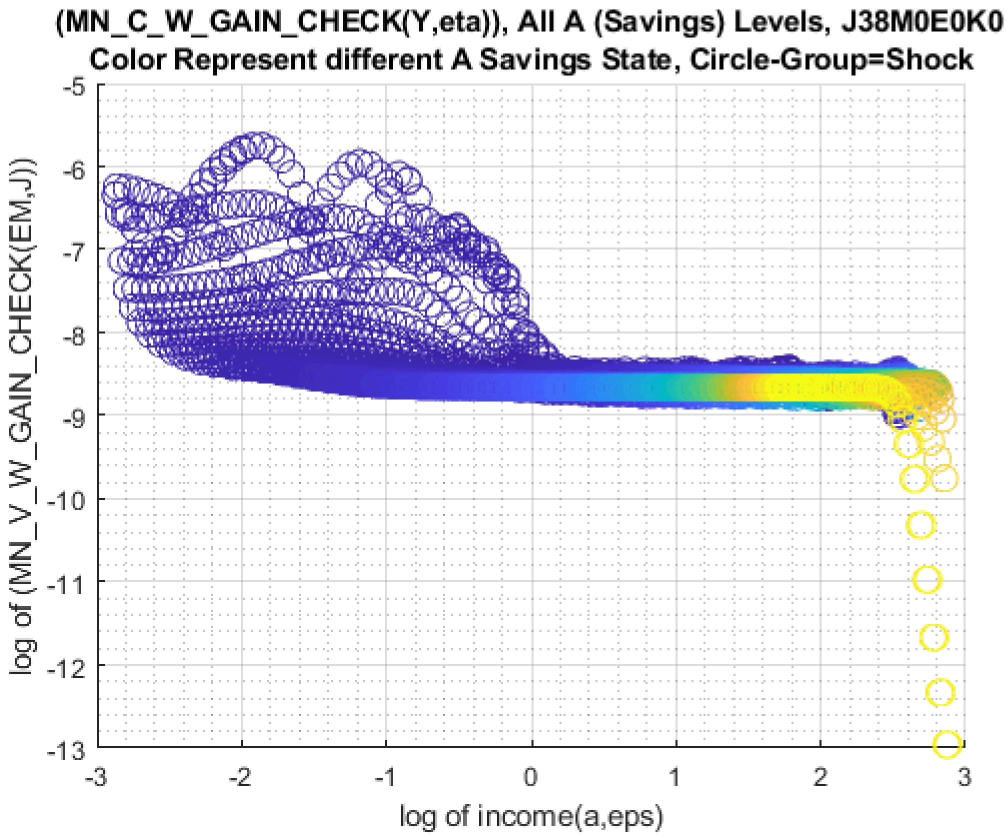


Plot all asset levels:

```
figure();
scatter((mt_total_inc_jemk(:)), (mt_C_W_gain_check_jemk_07(:)), 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_07(:)), 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_07, true, ["mean"], 3, 1, cl_mp_datasetd
```

```
xxx  MEAN(MN_V_U_GAIN_CHECK(KM,J)), welf_checks=2, TR=0.0018238  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	0.031908	0.030377	0.028045	0.025708	0.023751	0.0221
2	2	0	0.043916	0.04186	0.038596	0.035292	0.032514	0.030161
3	3	0	0.051104	0.04891	0.044943	0.04117	0.038001	0.035317
4	4	0	0.057927	0.055555	0.051064	0.046812	0.043241	0.040216
5	5	0	0.06325	0.060842	0.055977	0.051401	0.047557	0.044303
6	1	1	0.005762	0.0053279	0.0048412	0.0043955	0.0040141	0.0036902
7	2	1	0.0081705	0.0075587	0.0068473	0.0061989	0.005658	0.0051977
8	3	1	0.0099025	0.0091765	0.0083275	0.0075396	0.0068765	0.0063167
9	4	1	0.012339	0.011465	0.01043	0.0094589	0.008631	0.0079256
10	5	1	0.015374	0.014388	0.013133	0.01196	0.010966	0.01013

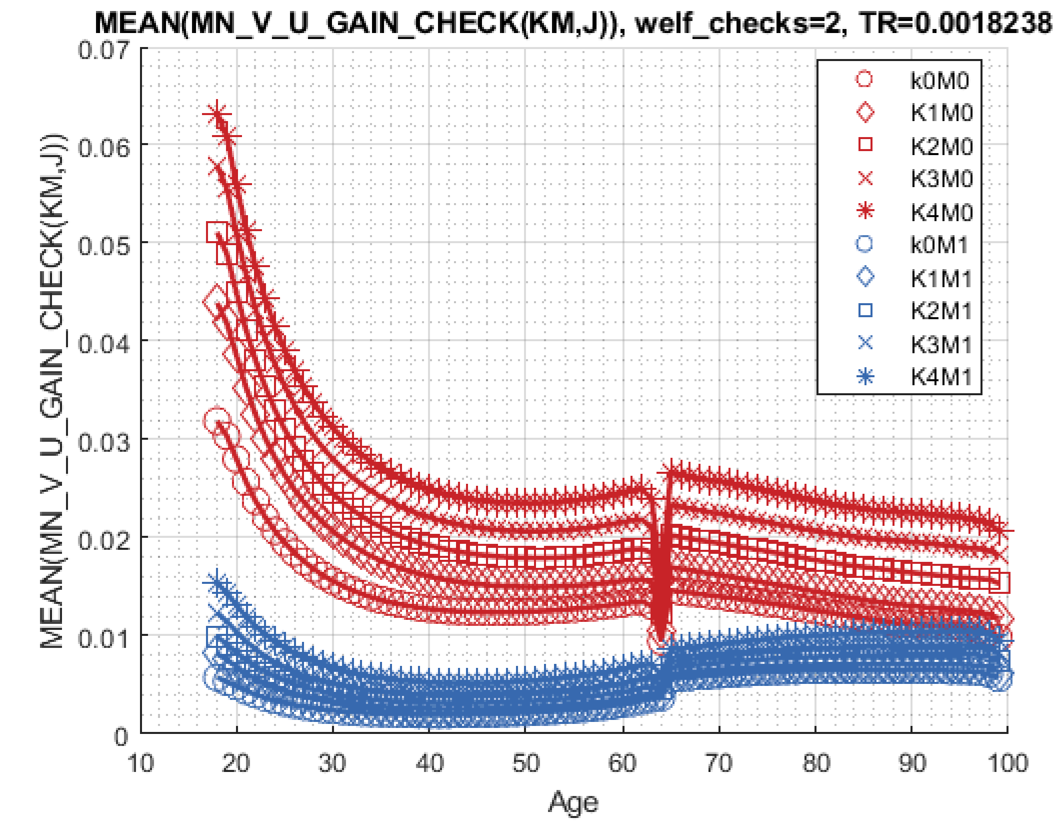
```
% Consumption Function
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(KM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2st
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check_07, true, ["mean"], 3, 1, cl_mp_
```

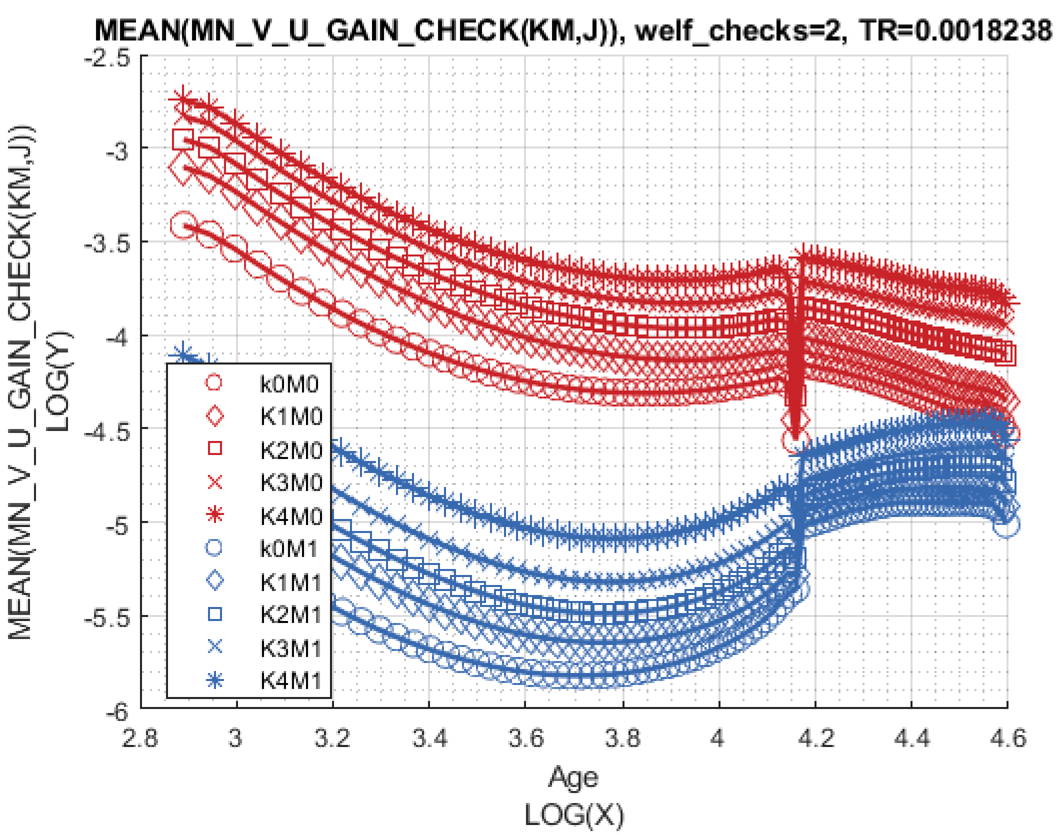
```
xxx  MEAN(MN_MPC_U_GAIN_CHECK(KM,J)), welf_checks=2, TR=0.0018238  xxxxxxxxxxxxxxxxxxxxxxxxxxx
```

group	kids	marry	mean_age_18	mean_age_19	mean_age_20	mean_age_21	mean_age_22	mean_age_23
1	1	0	0.078069	0.086102	0.087377	0.085109	0.082559	0.080767
2	2	0	0.088091	0.096475	0.099705	0.096801	0.094582	0.092454
3	3	0	0.096309	0.10647	0.11111	0.10818	0.10451	0.10207
4	4	0	0.10137	0.11245	0.11806	0.11424	0.11063	0.1079
5	5	0	0.10569	0.11834	0.12441	0.11996	0.11559	0.11262
6	1	1	0.091639	0.097614	0.095796	0.09274	0.094336	0.09373
7	2	1	0.098891	0.10394	0.10413	0.10026	0.097592	0.097894
8	3	1	0.10228	0.10811	0.10782	0.10839	0.10652	0.10731
9	4	1	0.10459	0.11283	0.11208	0.11053	0.10913	0.10771
10	5	1	0.11468	0.12506	0.12335	0.12035	0.11807	0.11752

Graph Mean Values:

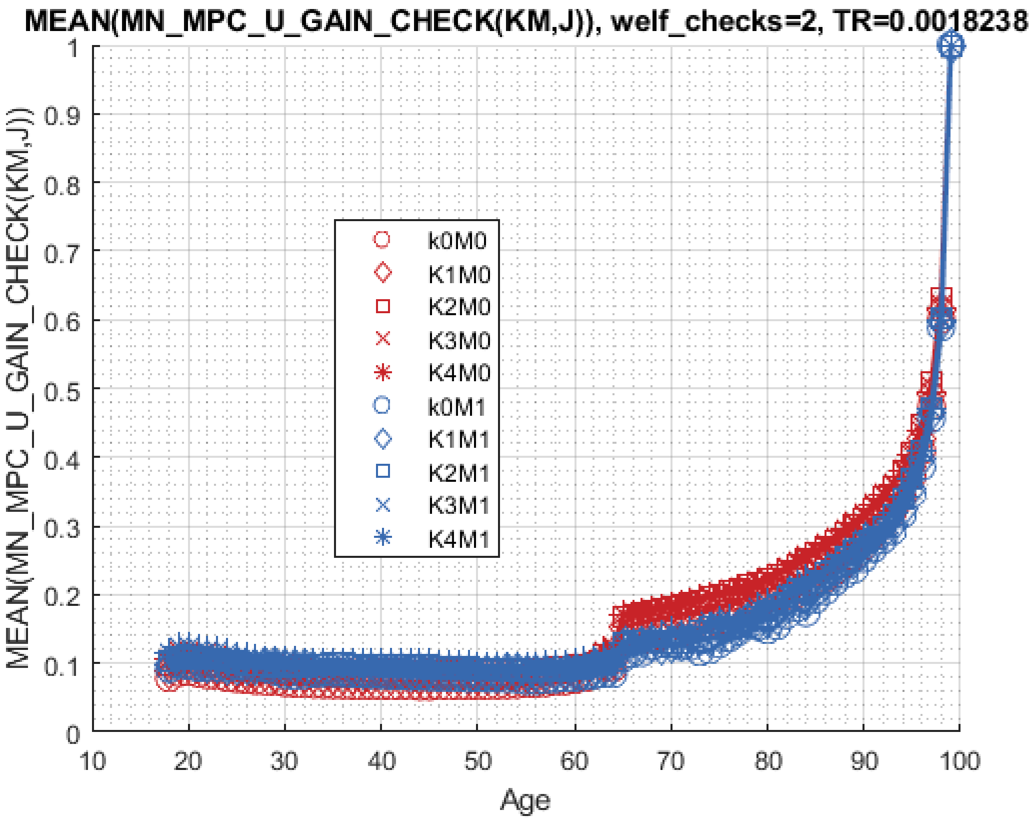
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(KM,J)), welf\_checks=' num2str(welf_checks) ', TR=' num
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(KM,J))'};
ff_graph_grid((tb_az_v{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);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

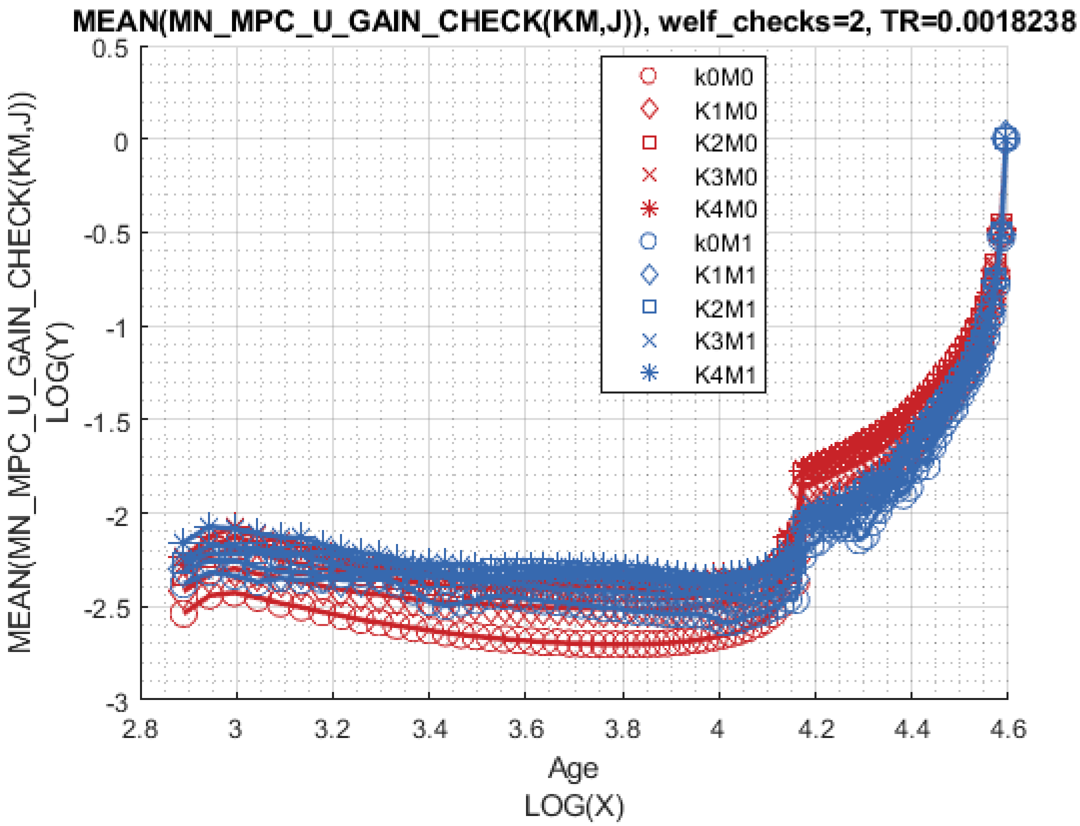




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
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_07, true, ["mean"], 3, 1, cl_mp_dataseto
```

```
xxx  MEAN(MN_V_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0018238  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	0.051256	0.049485	0.046969	0.044331	0.041985	0.039891
2	1	0	0.047986	0.045532	0.040481	0.035822	0.032041	0.028948
3	0	1	0.01129	0.010552	0.0097415	0.0089796	0.0083154	0.0077436
4	1	1	0.0093296	0.0086141	0.0076904	0.0068414	0.0061429	0.0055604

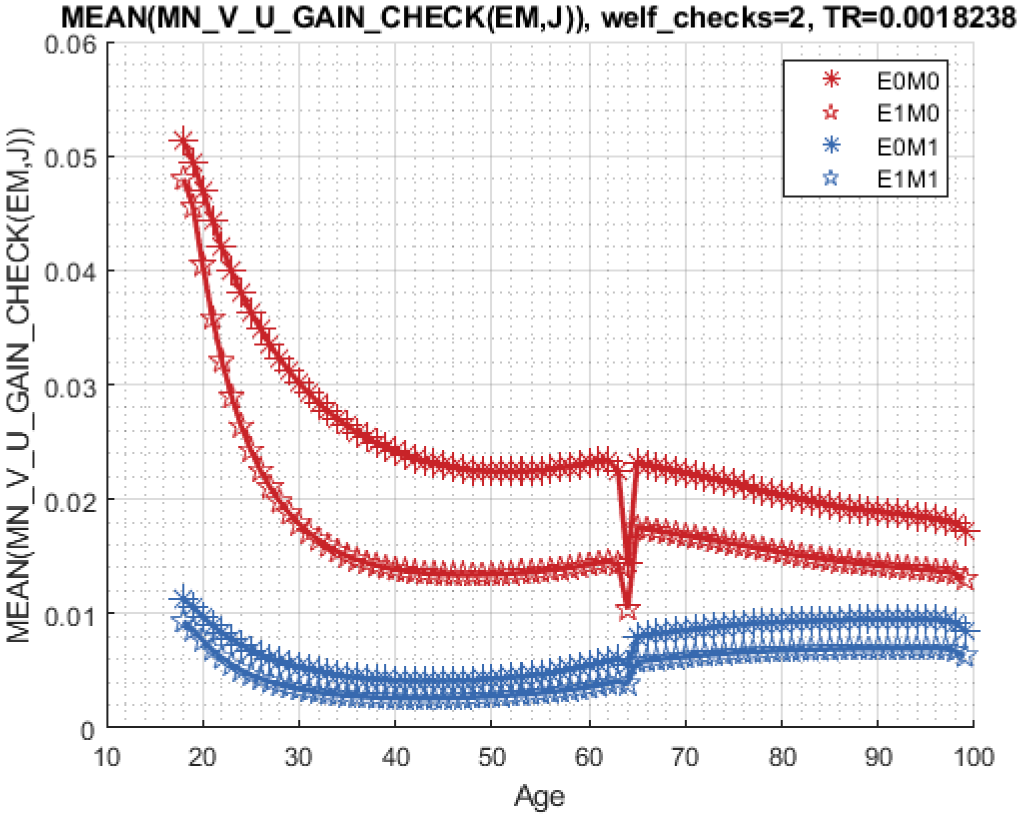
```
% Consumption
st_title = ['MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=' num2str(welf_checks) ', TR=' num2st
tb_az_c = ff_summ_nd_array(st_title, mn_MPC_U_gain_share_check_07, true, ["mean"], 3, 1, cl_mp_
```

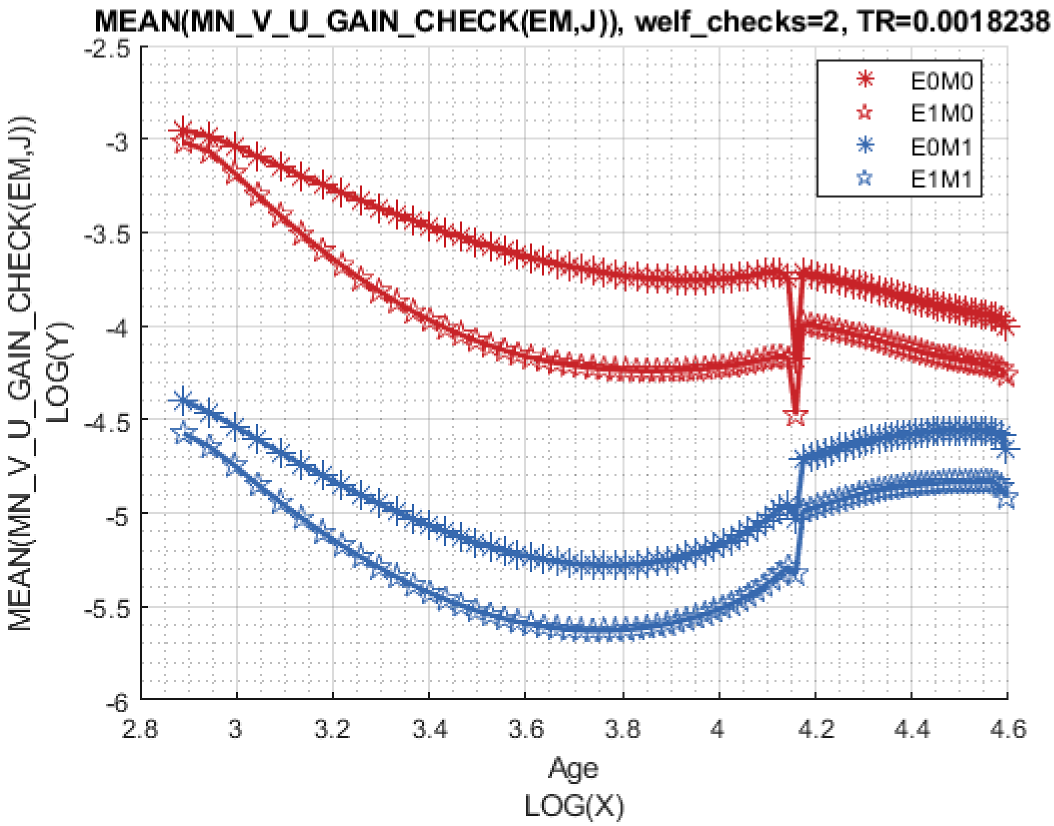
xxx MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0018238 xxxxxxxxxxxxxxxxxxxxxxxxxxx

group	edu	marry	mean_age_18	mean_age_19	mean_age_20	mean_age_21	mean_age_22	mean_age_23
1	0	0	0.081794	0.085652	0.08723	0.08665	0.085947	0.085754
2	1	0	0.10602	0.12228	0.12903	0.12307	0.1172	0.11257
3	0	1	0.093649	0.097104	0.096702	0.095946	0.095996	0.096866
4	1	1	0.11118	0.12192	0.12057	0.11696	0.11426	0.1128

Graph Mean Values:

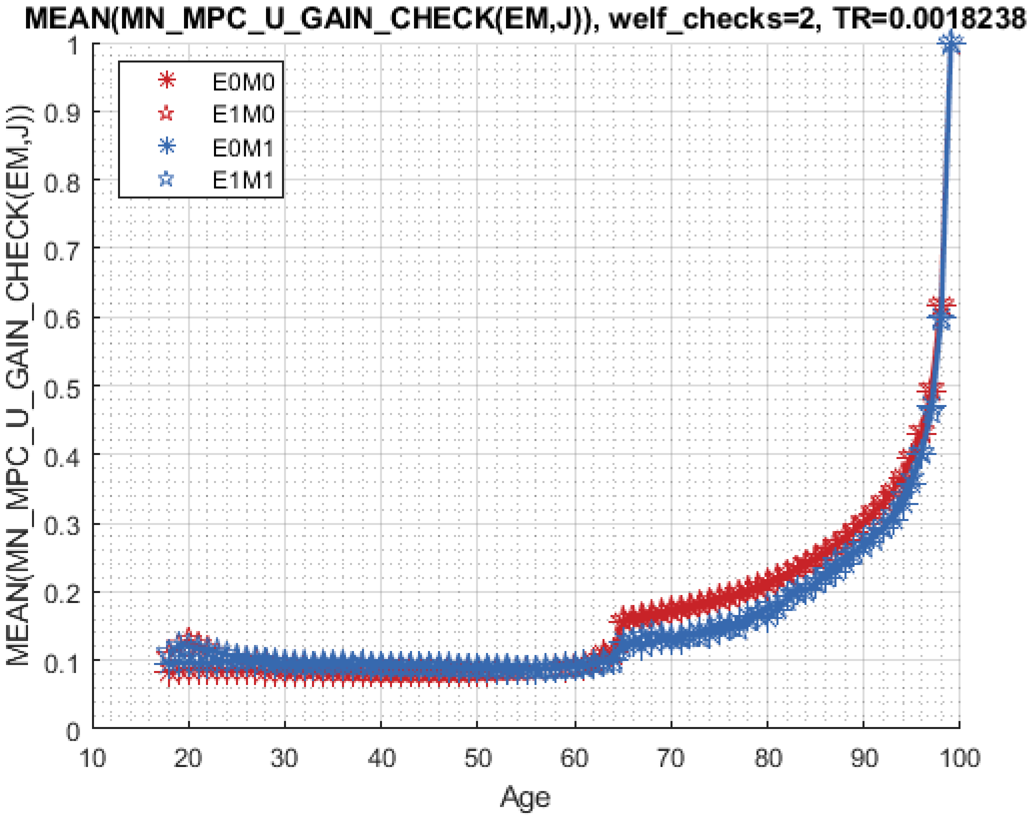
```
st_title = ['MEAN(MN\_V\_U\_GAIN\_CHECK(EM,J)), welf\_checks=' num2str(welf_checks) ', TR=' num
mp_support_graph('cl_st_graph_title') = {st_title};
mp_support_graph('cl_st_ytitle') = {'MEAN(MN\_V\_U\_GAIN\_CHECK(EM,J))'};
ff_graph_grid((tb_az_v{1:end, 4:end}), ar_row_grid, age_grid, mp_support_graph);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

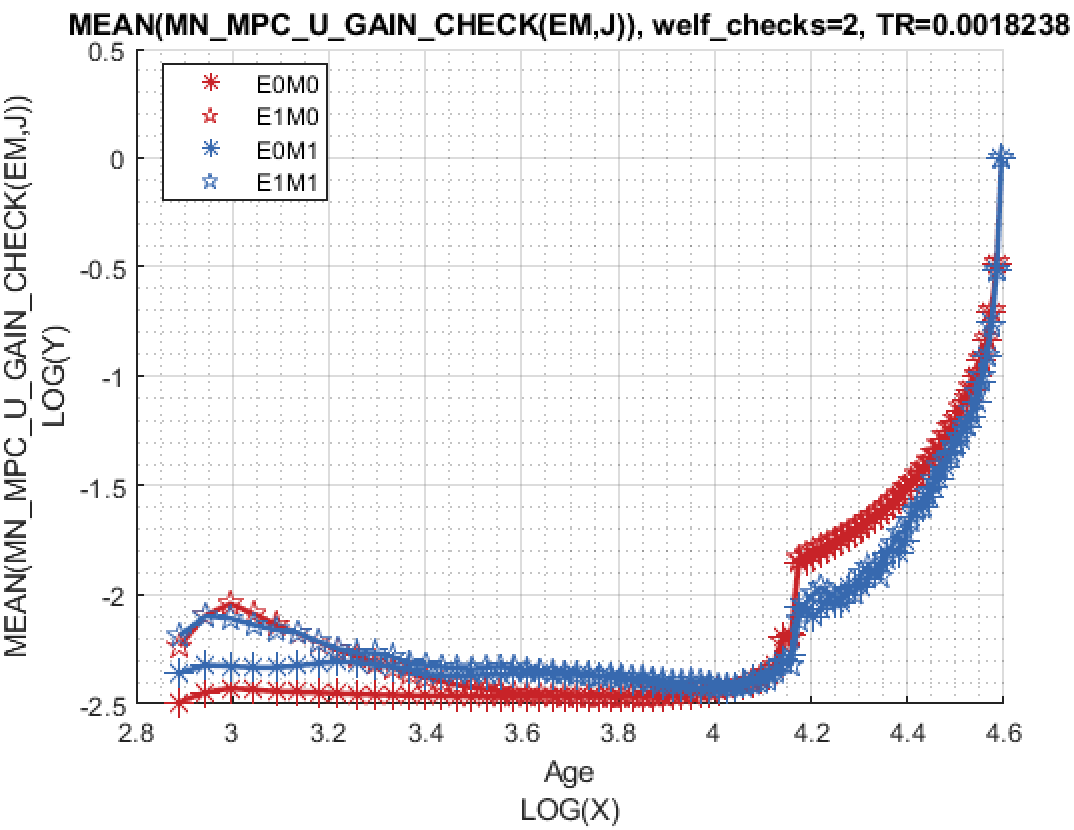




Graph Mean Consumption (***MPC: Share of Check Consumed***):

```
st_title = ['MEAN(MN\_MPC\_U\_GAIN\_CHECK(EM,J)), welf\_checks=' num2str(welf_checks) ', TR=' n
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=0.0018238
MEAN(MN_MPC_U_GAIN_CHECK(EM,J))
Age
E0M0
E1M0
E0M1
E1M1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
10
20
30
40
50
60
70
80
90
100


MEAN(MN_MPC_U_GAIN_CHECK(EM,J)), welf_checks=2, TR=0.0018238
MEAN(MN_MPC_U_GAIN_CHECK(EM,J))
LOG(Y)
Age
LOG(X)
E0M0
E1M0
E0M1
E1M1
0.5
0
-0.5
-1
-1.5
-2
-2.5
2.8
3
3.2
3.4
3.6
3.8
4
4.2
4.4
4.6