

```
Documents/GitHub/DynamicProgramming2024/problem_sets/problem_set_1/problem_set_1_code.m
1 %{}
2 -----
3 Purpose: Coding part of problem set 1
4 Created: Nico Rotundo 2024-10-28
5 -----
6 %}
7
8 %{
9 -----
10 Define parameters
11 -----
12 %}
13
14 % Terminal consumption at time T
15 C_T = 2.0;
16
17 % Initial interest rate
18 r_0 = 0.05;
19
20 % Growth rate of interest rate
21 alpha = 0.01;
22
23 %{
24 % Interest rate function as a handle function of time t
25 function r = interest_rate_function(t, r_0, alpha)
26     r = r_0 + alpha * t;
27 end
28 %}
29
30 % Inverse of the intertemporal elasticity of substitution
31 theta = 2.0;
32
33 % Time discount factor
34 rho = 0.03;
35
36 % Terminal time
37 T = 10;
38
39 % Display the result of interest rate function at specific time to check
40 fprintf('The interest rate at t = %d is: %.3f\n', 20, interest_rate_function(20, r_0, alpha));
41
42 % Checking how assert statements work in this matlab
43 %assert(interest_rate_function(20, r_0, alpha) == .25);
44
45
46 %{
47 -----
48 1. Solve the equation analytically using the integrating factor method. Then, plot the analytical solution as a function
49    of time t and consumption C(t)
50
51    - solved equation: C(t) = C_T*exp[(1/theta)*((r_0-rho)(t-T)+(alpha/2)(t^2-T^2))]
52 -----
53 %}
54
55 % Time vector from 0 to T with 100 points
56 t = linspace(0, T, 100);
57
58 % Define value for consumption at time t
59 C_t_analytical = C_T * exp((1 / theta) * ((r_0 - rho) * (t - T) + (alpha / 2) * (t.^2 - T^2)));
60
61 % Generate figure for analytical solution
62 figure;
63 plot(t, C_t_analytical, 'LineWidth', 2, 'Color', [41/255, 182/255, 164/255]);
64     xlabel('Time (t)');
65     set(gca, 'TickDir', 'out');
66     box off;
67     ylabel('Consumption C(t)');
68     title('Analytical Solution of C(t) over Time');
69     grid on;
70
71 % Export
72 exportgraphics(gcf, '/Users/nicorotundo/Documents/GitHub/DynamicProgramming2024/problem_sets/problem_set_1/output/question_1.pdf', 'ContentType', 'vector');
73
74 %{
75 -----
76 2. Solve the equation numerically using the finite difference method with 100 time steps. Plot the numerical and analytical
77    solutions together in a single figure for comparison.
78
79    - equation: C(t-dt) = (theta-dt(r_t-rho))/theta * C(t)
80 -----
81 %}
82
83 % Time step (\triangle t)
84 dt = 10/99;
85
86 % Number of time steps
87 N = 100;
88
89 % Vector to store consumption values
90 C_t_numeric = zeros(N, 1);
91
92 % Set terminal consumption
93 C_t_numeric(end) = C_T;
94
95 % Time vector from 0 to T with 100 points
96 t = linspace(0, T, N);
97
98 % Time stepping using the finite difference method
99 for n = N:-1:2
100
101     % Compute consumption at the next time step using backward difference
102     C_t_numeric(n-1) = (theta - dt*(r_0 + alpha*t(n)-rho))/theta * C_t_numeric(n);
103
104 end
105
106 % Generate figure for numeric solution
107 figure;
108 plot(t, C_t_numeric, 'LineWidth', 2, 'Color', [250/255, 165/255, 35/255]);
109     xlabel('Time (t)');
110     set(gca, 'TickDir', 'out');
111     box off;
112     ylabel('Consumption C(t)');
113     title('Numerical Solution of C(t) over Time');
114     grid on;
115
116 % Generate both analytical and numeric sultions on the same plot
117 figure;
118
119     % Plot analytical solution
120     plot(t, C_t_analytical, 'LineWidth', 2, 'Color', [41/255, 182/255, 164/255]);
121
122     hold on;
123
124     % Plot numerical solution
125     plot(t, C_t_numeric, 'bo--', 'LineWidth', 2, 'Color', [250/255, 165/255, 35/255]);
126
127     % Add labels and title
128     xlabel('Time (t)');
129     ylabel('Consumption C(t)');
130     title('Analytical vs Numerical Solutions of C(t) over Time; 100 Time Steps');
131
132     % Add a legend
133     legend({'Analytical Solution', 'Numerical Solution'}, 'Location', 'southeast');
134
135     % Adjust plot settings
136     set(gca, 'TickDir', 'out');
137     box off;
138     grid on;
139
140     hold off;
141
142 % Export
143 exportgraphics(gcf, '/Users/nicorotundo/Documents/GitHub/DynamicProgramming2024/problem_sets/problem_set_1/output/question_2.pdf', 'ContentType', 'vector');
144
145 %{
146 -----
147 3. Repeat the numerical solution using 10 time steps. Plot the numerical and analytical solutions together in a single
148    figure.
149 -----
150 %}
151
152 % Time step (\triangle t)
153 dt = 10/9;
154
155 % Number of time steps
156 N = 10;
157
158 % Vector to store consumption values
159 C_t_numeric = zeros(N, 1);
160
161 % Set terminal consumption
162 C_t_numeric(end) = C_T;
163
164 % Time vector from 0 to T with 10 points
165 t_10 = linspace(0, T, N);
166
167 % Time stepping using the finite difference method
168 for n = N:-1:2
169
170     % Compute consumption at the next time step using backward difference
171     C_t_numeric(n-1) = (theta - dt*(r_0 + alpha*t_10(n)-rho))/theta * C_t_numeric(n);
172
173 end
174
175 % Generate both analytical and numeric sultions on the same plot
176 figure;
177
178     % Plot analytical solution
179     plot(t, C_t_analytical, 'LineWidth', 2, 'Color', [41/255, 182/255, 164/255]);
180
181     hold on;
182
183     % Plot numerical solution
184     plot(t_10, C_t_numeric, 'bo--', 'LineWidth', 2, 'Color', [250/255, 165/255, 35/255]);
185
186     % Add labels and title
187     xlabel('Time (t)');
188     ylabel('Consumption C(t)');
189     title('Analytical vs Numerical Solutions of C(t) over Time; 10 Time Steps');
190
191     % Add a legend
192     legend({'Analytical Solution', 'Numerical Solution'}, 'Location', 'southeast');
193
194     % Adjust plot settings
195     set(gca, 'TickDir', 'out');
196     box off;
197     grid on;
198
199     hold off;
200
201 % Export
202 exportgraphics(gcf, '/Users/nicorotundo/Documents/GitHub/DynamicProgramming2024/problem_sets/problem_set_1/output/question_3.pdf', 'ContentType', 'vector');
203
```