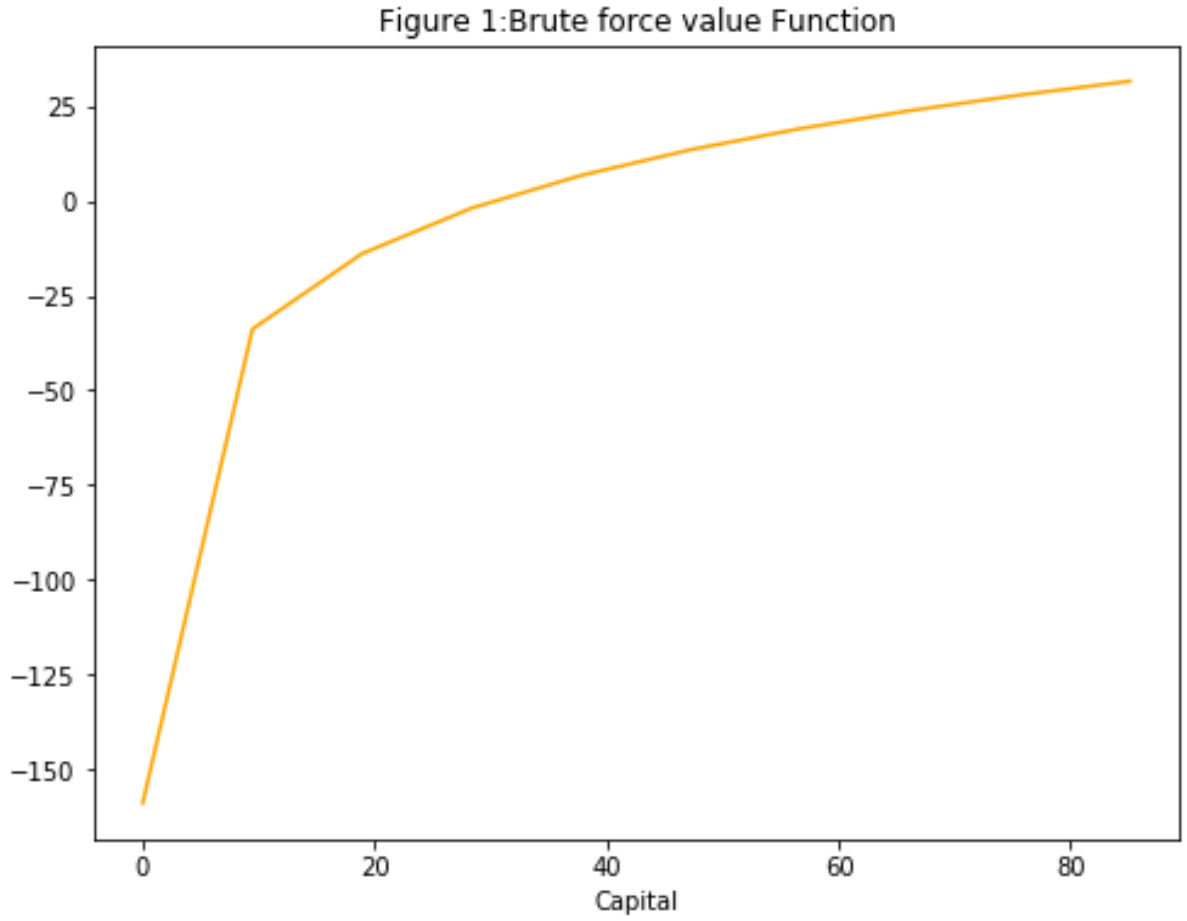
Macroeconomics Homework 4
Name: Giada Bozzelli, Abhinav Khemka, Richard Sparkes

**Question 1.1**

All the value functions were evaluated with an initial v=0.

Brute force Value function graph:



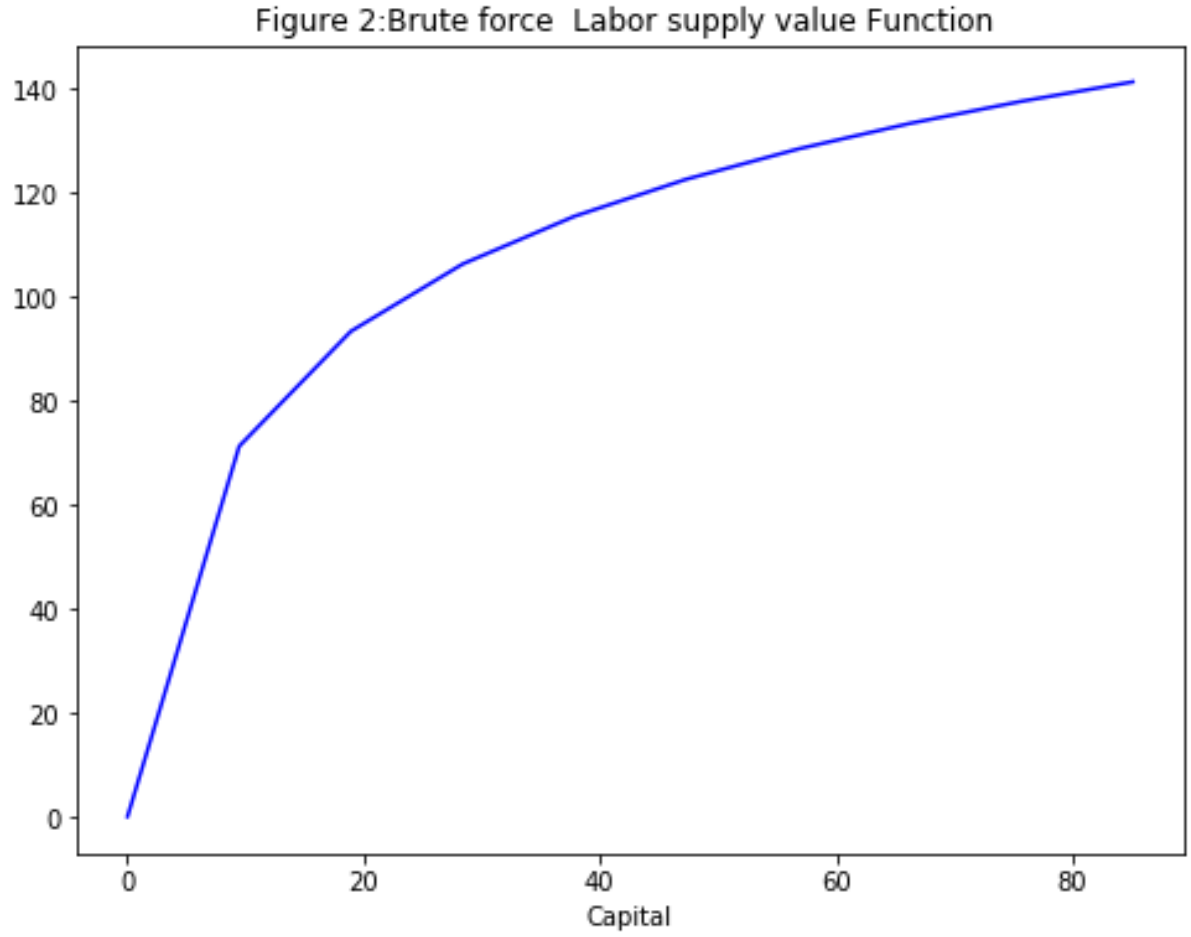
| Part | Value function conditions | Time | Time Improvements |
|---|---|---|---|
| a | Brute force | 0.0184 | |
| b | Monotonicity | 0.0124 | Including monotonicity reduces slightly the time of iteration. |
| c | Concavity | 0.0097 | Including concavity the time of iterations shows a strong improvement. |
| d | Local Search | 0.0107 | Local search also shows an improvement from both brute force and monotonicity. |
| e | Monotonicity and Concavity | 0.0109 | Similar to local search in terms of number of iterations and time. |
| f | Howard Policy | 0.0216 | Howard policy starting from 100 iterations takes longer than other conditions. |
| g | Howard Policy with different iterations | | We can observe that as we introduce steps the time improves considerably when compared to all other conditions. As steps increase time improves considerably. |
| | 5 steps | 0.0010 | |
| | 10 steps | 0.0120 | |
| | 20 steps | 0.0082 | |
| | 50 steps | 0.0074 | |

| Part | Value function conditions | Iterations |
|---|---|---|
| a | Brute force | 9 |
| b | Monotonicity | 226 |
| c | Concavity | 398 |
| d | Local Search | 400 |
| e | Monotonicity and Concavity | 400 |
| f | Howard Policy | 82 |
| g | Howard Policy with different iterations | |
| | 5 steps | 40 |
| | 10 steps | 40 |
| | 20 steps | 40 |
| | 50 steps | 40 |

We have tried different grid sizes and reducing the size greatly reduces the number of iterations.  Monotonicity reduced the number of iterations compared with local search and method e. However, brute force shows a really low number of iterations.
Howard policy reduces the number of iterations compared to all conditions except brute force.

**Question 1.2**

Brute force Value function with labor supply graph:



| Part | Value function condtions | Time | Time Improvements |
|---|---|---|---|
| a | Brute force | 0.0370 | |
| b | Monotoncity | 0.0494 | Including monotonicity increases the time of iteration. |
| c | Concavity | 0.0292 | Including concavity time is better than brute force; |
| d | Local Search | 0.0211 | Local search also shows a massive improvement from both brute force and monotonicity. Compared to concavity there is only a slight improvement. |
| e | Monotoncity and Concavity | 0.4262 | Including monotonicity and concavity increase the time compared than all other conditions. Number of iterations are really high |
| f | Howard Policy | 0.0661 | Howard Policy takes longer than all previous conditions. |
| g | Howard Policy with different iterations | | The number of steps don't impact the timing significantly. |
| | 5 steps | 0.0607 | |
| | 10 steps | 0.0618 | |
| | 20 steps | 0.0587 | |
| | 50 steps | 0.0670 | |

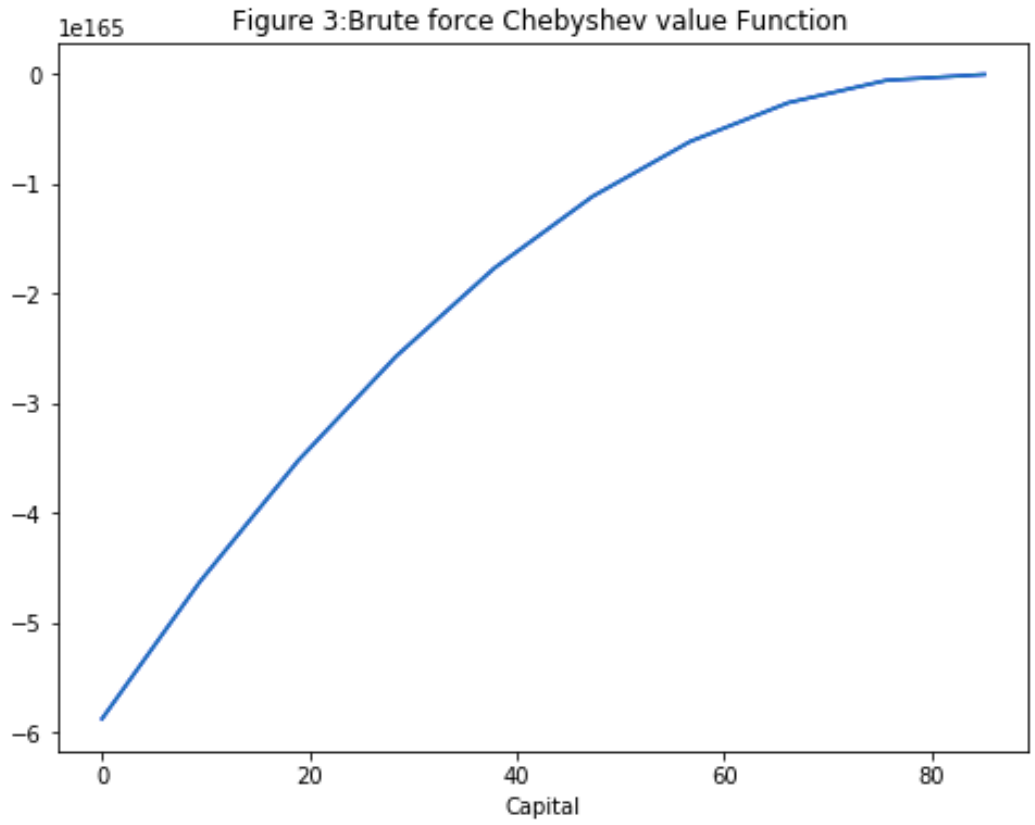
When compared to Ex 1.1, for all conditions the value function with labor supply takes longer. This is unexpected since labor is now continuous and should reduce the time but we think this happens mainly due to the fact that now there is an extra parameter to be evaluated. We can also see a similar trend for most conditions except when both monotonicity and concavity is included in comparison to the first value function.

| Part | Value function condtions | Iterations |
|---|---|---|
| a | Brute force | 399 |
| b | Monotoncity | 379 |
| c | Concavity | 390 |
| d | Local Search | 195 |
| e | Monotoncity and Concavity | 23065 |
| f | Howard Policy | 82 |
| g | Howard Policy with different iterations | 80 |
| | 5 steps | 80 |
| | 10 steps | 80 |
| | 20 steps | 80 |
| | 50 steps | 80 |

Local search is the only conditions which reduces the number of iterations drastically. Howard policy greatly reduces the iterations compared to all previous cases.

**Question 1.3**

Brute force Value function Chebyshev graph:




Brute force Chebyshev Time: 2.0595
Chebyshev Iterations :1000

The Chebyshev graph above shows higher concavity when compared to the previous two graphs. The value function iterations take much longer and the number of iterations drastically increase. We even tried to redo the exercise with different grid points and observe Chebyshev nodes are more sensitive to time when compared to Ex 1 and 2.