Intro to Simulation and Modeling

Homework #01

Gamblers Ruin and Family of ODEs

James Deromedi
2-1-2019

Question 1 (Gambler's Ruin).

For each, compute the mean and standard deviation of the number of rounds until bankruptcy.

See Figure 2 for the mean and std of 1000 people with 99999 rounds. The variable names are brokemean and standev. The standev is so large because the distribution of people going brankrupt is not a guassian cruve.

Question 2 (Gambler's Ruin).

How does the mean number of rounds until bankruptcy depend on the size of the starting wallet?

The mean amount of rounds is directly related to the starting size of the wallet, the smaller the wallet, the smaller the mean and vise versa. Some examples I ran, at $3 start, the mean was roughly 200 rounds. With a $100 start, the mean was roughly 5000 rounds. This is because there is less 'forgiveness' with a smaller wallet size or it is much easier to lose more quickly.

Editor - D:\American Work\2019 Spring\Simulations and Modeling\Homework\Homework 1\GamblersRuin.m

FamilyofODEs.m | GamblersRuin.m

```
num_tries = 99999;
num_points = 1000;
pos = (zeros(num_points, num_tries)) + 100;  %Creates array of all '25'

for n = 1: num_tries
    for m = 1: num_points
        if pos(m,n) ~= 0   %If cell is not zero, do math
            pos(m,n+1) = pos(m,n) + 2*(rand(1,1)<=0.5) - 1;
        else
            pos(m,n+1) = 0; %Sets the next cells to zero
        end
    end
end
figure; plot(1:(num_tries+1),pos(1:10,:));

[value, index] = min(pos, [], 2);    %Finds the min values of each row
                                     %and saves the value and index
broke = index(value == 0);           %Saves the indexes of 'bankrupts'
brokemean = mean(broke);             %Finds the mean of indexes
standDev = std(broke);               %Finds the std of indexes
figure; hist(broke)
```

Figure 1: The code written for GamblersRuin. Much thanks to Professor Knapp for helping with finding the position of bankrupts

Workspace

Name	Value
broke	*717x1 double*
brokemean	2.1919e+04
index	*1000x1 double*
m	1000
n	99999
num_points	1000
num_tries	99999
pos	*1000x100000 double*
standDev	2.2366e+04
value	*1000x1 double*

Figure 2: The output of the code from Figure 1.

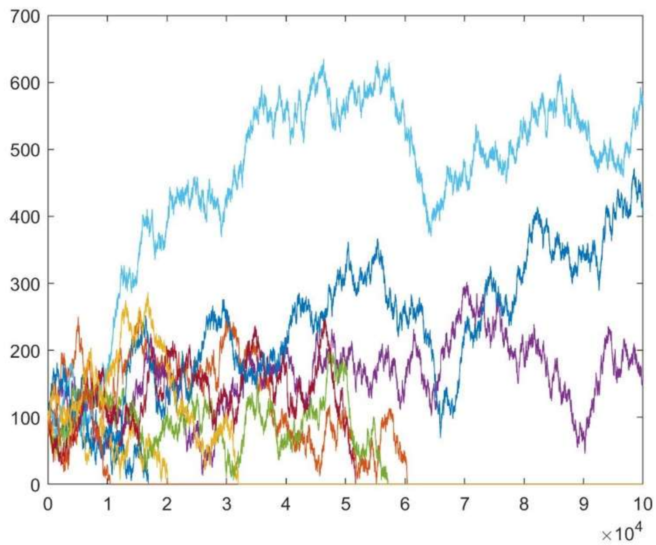


Figure 3: Line graph of only 100 'players' showing their trajectory of their money.

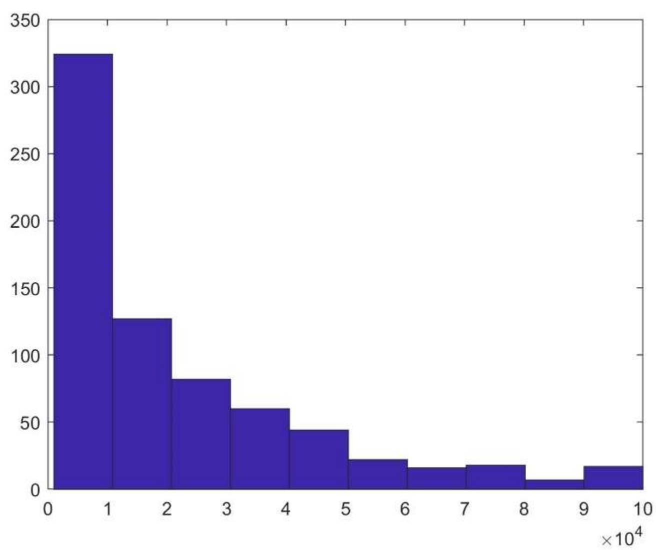


Figure 4: Histogram of the ending wallet size of each player. As it can be seen, it looks like an exponentially decreasing curve

Question 3 (Family of ODEs).

Fix a = 1 and experiment with various values of b. Find the value of b for which the qualitative behavior of the ODE changes.

See Figure 6, which shows A is fixed at 1 and is changing B from -10 to 10. As seen in the graph, it is avoiding zero and never going higher than 2.

Question 4 (Family of ODEs).

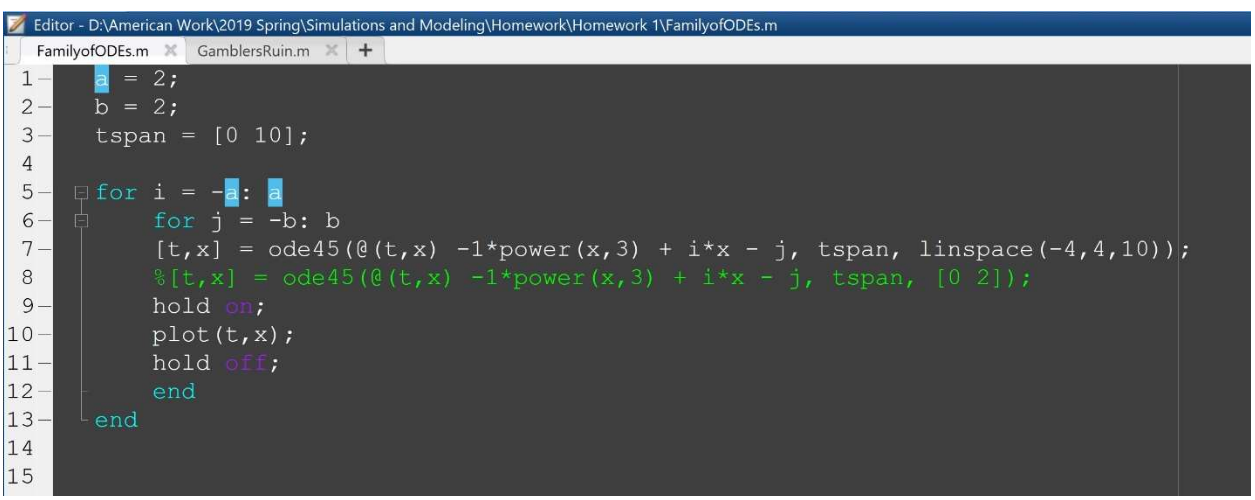
Fix b = 0 and experiment with various values of a. What kind of behavior do we have when a < 0 and a > 0? How does this relate to the graph of f(x, a, 0) = −x 3 + a?

See Figure 7, were B is set to zero and A is changing from -10 to 10. As seen in the graph, it seems to be converging at zero. See Figure 8, for A < 0 and see Figure 9, for A > 0. To compare to function in the problem see Figure 10.

Question 5 (Family of ODEs).

Bonus: Formulate and test a conjecture about the (a, b) pairs at which qualitative behavior change.

See Figure 11. A = 2, B = 2. Shows all possible paths of the function.




```
a = 2;
b = 2;
tspan = [0 10];

for i = -a: a
    for j = -b: b
    [t,x] = ode45(@(t,x) -1*power(x,3) + i*x - j, tspan, linspace(-4,4,10));
    %[t,x] = ode45(@(t,x) -1*power(x,3) + i*x - j, tspan, [0 2]);
    hold on;
    plot(t,x);
    hold off;
    end
end
```

Figure 5: Code for Family of ODEs. Thank you to Professor Knapp for teaching me about the linspace function.

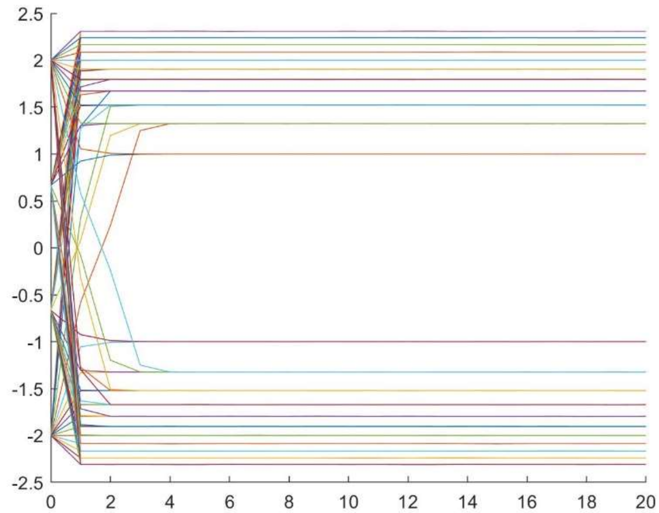


Figure 6: A = 1, B = [-10 10]. Showing that the graph is diverging away from zero.

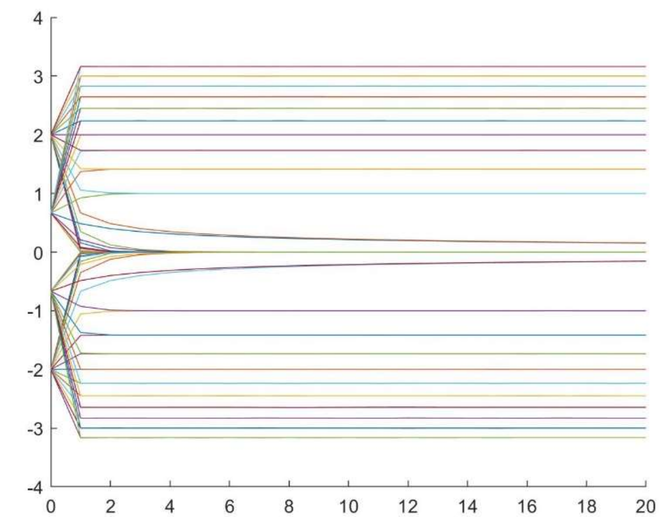


Figure 7: B = 0, A = [-10 10]. Showing that the graph is diverging towards zero.

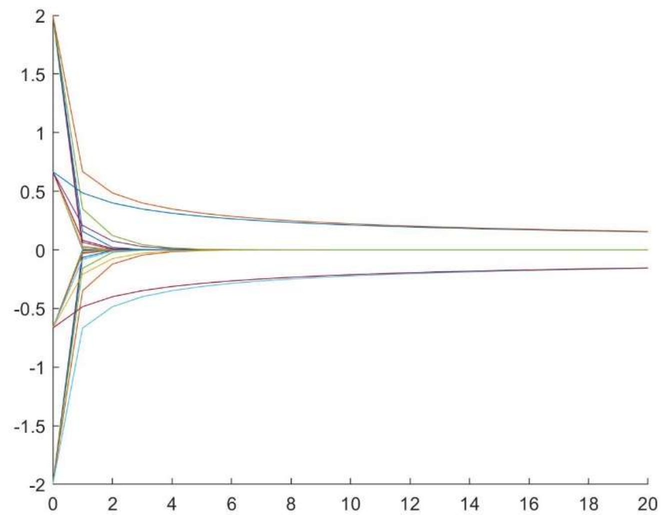


Figure 8: B = 0, A < 0. Showing when A is negative how it converges quicklyto zero.

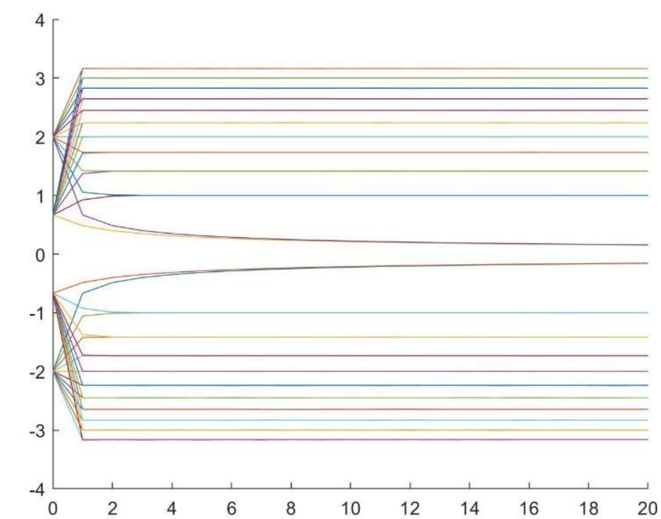


Figure 9: B = 0, A > 0. Showing how the graph converges to zero slowly

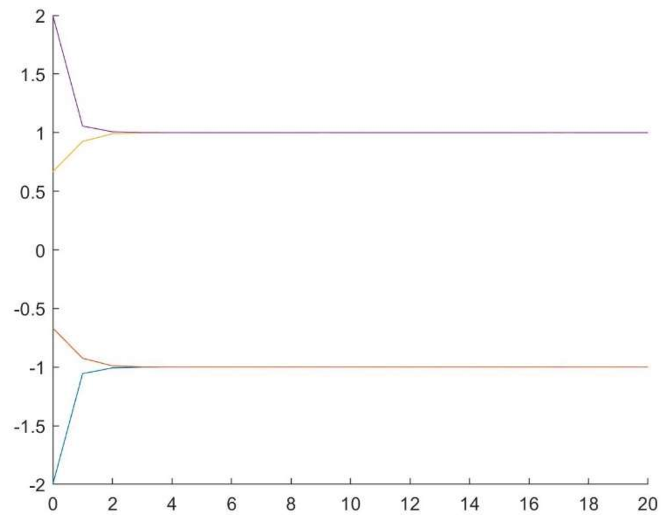


Figure 10: B =0, From function given in problem 4. To show comparison.

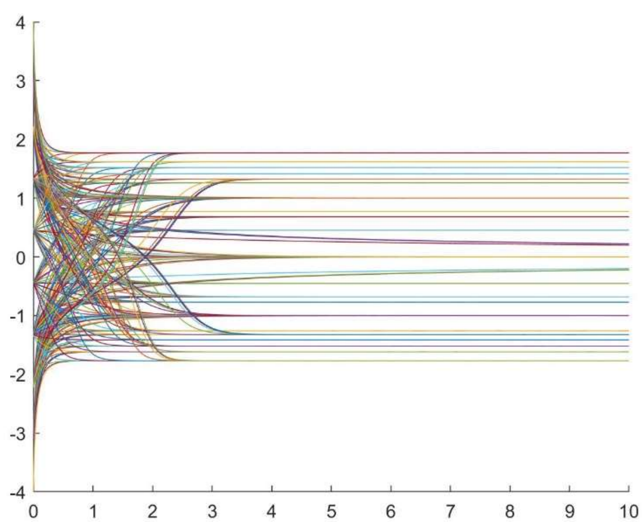


Figure 11: A = 2, B =2. Showing all possible paths of the ODE.