

**University of Tehran**

**Department of Electrical and Computer Engineering (ECE)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Data and Information fusion**

**Assignment No.3**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Mohammad Hossein Badiei**

**Student ID: 810199106**

**Majors: Electrical Engineering and Computer Engineering**

**Fields: Artificial Intelligence in CE and Control in EE**

**Supervisor: Prof. Behzad Moshiri**

**April 27, 2021**

**Part 1**

* **Section A**

First, let me calculate λ parameter by values.

Let us solve above equation with the help of matlab tool. The result is shown below.

close;  
clear;  
clc;

## Definition of variable lambda

syms x

## Part 1-a

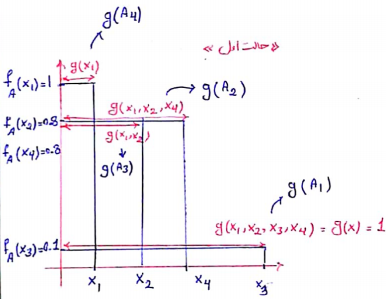
C = coeffs(expand((1+0.3\*x)\*(1+0.6\*x)\*(1+0.7\*x)\*(1+0.3\*x) - x - 1),'All');  
digits(6)  
r = vpa(roots(C))

r =  
   
 0  
 -0.914477  
 - 4.42371 - 2.54302i  
 - 4.42371 + 2.54302i

[*Published with MATLAB® R2018a*](https://www.mathworks.com/products/matlab)

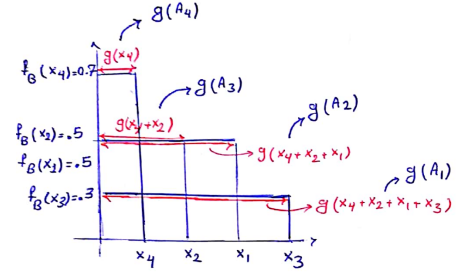
Paying attention to the sum of ’s, which is higher than one, we conclude the acceptable value of Lambda is approximately equal to -0.914 .

**Assigning Sugeno technic to movie A**



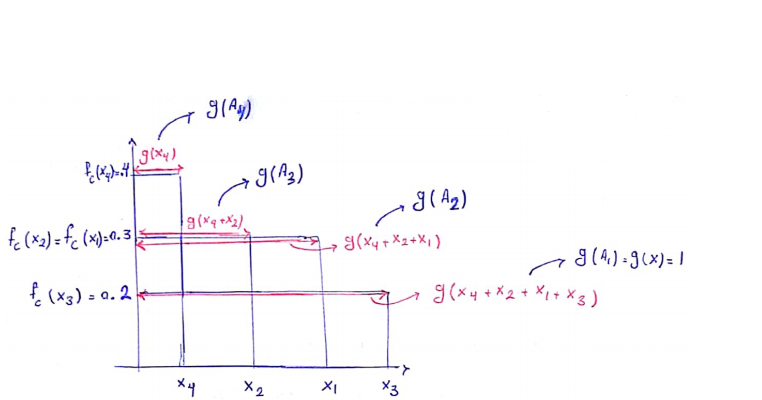
**Assigning Choquet technic to movie A**

**Assigning Sugeno technic to movie B**



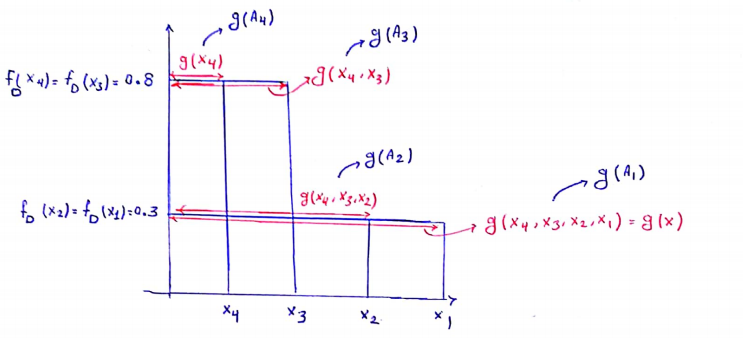
**Assigning Choquet technic to movie B**

**Assigning Sugeno technic to movie C**



**Assigning Choquet technic to movie C**

**Assigning Sugeno technic to movie D**



**Assigning Choquet technic to movie D**

Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Movie A | Movie B | Movie C | Movie D |
| Sugeno |  |  |  |  |
| Choquet |  |  |  |  |

Ranking:

* Sugeno
* Rank 1: Movie A & Movie D
* Rank 2: Movie B
* Rank 3: Movie C
* Choquet
* Rank 1: Movie A
* Rank 2: Movie D
* Rank 2: Movie B
* Rank 3: Movie C

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

* **Section B**

Now, let us calculate λ parameter for this section by values.

In order to solve above equation, I get help from matlab tool as shown below.

## Part 1-b

C\_b = coeffs(expand((1+0.3\*x)\*(1+0.6\*x) - x - 1),'All');  
digits(6)  
r\_b = vpa(roots(C\_b))

r\_b =  
   
 0  
 0.555556

[*Published with MATLAB® R2018a*](https://www.mathworks.com/products/matlab)

Paying attention to the sum of ’s, which is lower than one, we conclude the acceptable value of Lambda is approximately equal to 0.555 .

**Assigning Sugeno technic to movie A**

**Assigning Choquet technic to movie A**

**Assigning Sugeno technic to movie B**

**Assigning Choquet technic to movie B**

**Assigning Sugeno technic to movie C**

**Assigning Choquet technic to movie C**

**Assigning Sugeno technic to movie D**

**Assigning Choquet technic to movie D**

Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Movie A | Movie B | Movie C | Movie D |
| Sugeno |  |  |  |  |
| Choquet |  |  |  |  |

Ranking:

* Sugeno
* Rank 1: Movie A
* Rank 2: Movie B
* Rank 3: Movie C & Movie D
* Choquet
* Rank 1: Movie A
* Rank 2: Movie B
* Rank 3: Movie C & Movie D

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

* **Section C**

Now, let us calculate λ parameter for this section by values.

In order to solve above equation, I get help from matlab tool as shown below.

## Part 1-c

C\_c = coeffs(expand((1+0.7\*x)\*(1+0.3\*x) - x - 1),'All');  
digits(6)  
r\_c = vpa(roots(C\_c))

r\_c =  
   
 0  
 0

[*Published with MATLAB® R2018a*](https://www.mathworks.com/products/matlab)

Paying attention to the sum of ’s, which is equal to one, we conclude the acceptable value of Lambda is equal to zero .

**Assigning Sugeno technic to movie A**

**Assigning Choquet technic to movie A**

**Assigning Sugeno technic to movie B**

**Assigning Choquet technic to movie B**

**Assigning Sugeno technic to movie C**

**Assigning Choquet technic to movie C**

**Assigning Sugeno technic to movie D**

**Assigning Choquet technic to movie D**

Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Movie A | Movie B | Movie C | Movie D |
| Sugeno |  |  |  |  |
| Choquet |  |  |  |  |

Ranking:

* Sugeno
* Rank 1: Movie D
* Rank 2: Movie A & Movie B & Movie C
* Choquet
* Rank 1: Movie D
* Rank 2: Movie B
* Rank 3: Movie A
* Rank 4: Movie C

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

* **Section D**

Now, let us calculate λ parameter for this section by values.

In order to solve above equation, I get help from matlab tool as shown below.

## Part 1-c

C\_d = coeffs(expand((1+0.6\*x)\*(1+0.7\*x) - x - 1),'All');  
digits(6)  
r\_d = vpa(roots(C\_d))

r\_d =  
   
 0  
 -0.714286

[*Published with MATLAB® R2018a*](https://www.mathworks.com/products/matlab)

Paying attention to the sum of ’s, which is higher than one, we conclude the acceptable value of Lambda is approximately equal to -0.714 .

**Assigning Sugeno technic to movie A**

**Assigning Choquet technic to movie A**

**Assigning Sugeno technic to movie B**

**Assigning Choquet technic to movie B**

**Assigning Sugeno technic to movie C**

**Assigning Choquet technic to movie C**

**Assigning Sugeno technic to movie D**

**Assigning Choquet technic to movie D**

Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Movie A | Movie B | Movie C | Movie D |
| Sugeno |  |  |  |  |
| Choquet |  |  |  |  |

Ranking:

* Sugeno
* Rank 1: Movie D
* Rank 2: Movie A
* Rank 2: Movie B
* Rank 2: Movie C
* Choquet
* Rank 1: Movie D
* Rank 2: Movie A
* Rank 3: Movie B
* Rank 4: Movie C

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Part 2**

We define two functions named Sugeno and Choquet and tset them as shown below:

* Sugeno tests
* clear;close;clc;  
    
  colNames = {'Movie\_A' 'Movie\_B' 'Movie\_C' 'Movie\_D'};  
  rowNames = {'1st\_section', '2nd\_section', '3rd\_section3', '4th\_section4'};  
  Table = array2table([Sugeno([0.3,0.6,0.7,0.3],[1.0,0.8,0.1,0.8]),...  
   Sugeno([0.3,0.6,0.7,0.3],[0.5,0.5,0.3,0.7]),...  
   Sugeno([0.3,0.6,0.7,0.3],[0.3,0.3,0.2,0.4]),...  
   Sugeno([0.3,0.6,0.7,0.3],[0.3,0.3,0.8,0.8]);...  
   Sugeno([0.3,0.6],[1.0,0.8]),...  
   Sugeno([0.3,0.6],[0.5,0.5]),...  
   Sugeno([0.3,0.6],[0.3,0.3]),...  
   Sugeno([0.3,0.6],[0.3,0.3]);...  
   Sugeno([0.7,0.3],[0.1,0.8]),...  
   Sugeno([0.7,0.3],[0.3,0.7]),...  
   Sugeno([0.7,0.3],[0.2,0.4]),...  
   Sugeno([0.7,0.3],[0.8,0.8]);...  
   Sugeno([0.6,0.7],[0.8,0.1]),...  
   Sugeno([0.6,0.7],[0.5,0.3]),...  
   Sugeno([0.6,0.7],[0.3,0.2]),...  
   Sugeno([0.6,0.7],[0.3,0.8]),...  
   ],'VariableNames',colNames,'RowNames',rowNames)  
    
    
  function res = Sugeno(mu, f)  
  syms x  
    
  n = size(mu,2);  
    
  lambdaCalFunc = 1;  
  for i=1:1:n  
   lambdaCalFunc = lambdaCalFunc\*(1+ x\*mu(1,i));  
  end  
  C = coeffs(expand(lambdaCalFunc - x - 1),'All');  
  digits(6)  
  r = vpa(roots(C));  
  mr = size(r,1);  
    
  lambda = 0;  
  if sum(mu) == 1  
   lambda = 0;  
  elseif sum(mu) > 1  
   for i=1:1:mr  
   if isreal(r(i,1)) && r(i,1)>-1 && r(i,1)<0  
   lambda = r(i,1);  
   end  
   end  
  else  
   for i=1:1:mr  
   if isreal(r(i,1)) && r(i,1)>0  
   lambda = r(i,1);  
   end  
   end  
  end  
    
  F = [mu' f'];  
  A = zeros(n,2);  
  for i=1:1:n  
   [a, index]=max(F(:,2));  
   A(i,:) = [F(index,1) a];  
   F(index,:)=[];  
  end  
    
  GAi = zeros(n,2);  
  Gi = 0;  
  for i=1:1:n  
    
   Gi = Gi + A(i,1) + lambda\*Gi\*A(i,1);  
    
   GAi(i,:) = [Gi, A(i,2)];  
  end  
    
  minComparison = ones(n,1);  
  for i=1:1:n  
   minComparison(i,1) = min(minComparison(i,1),min(GAi(n-i+1,:)));  
  end  
  res = max(minComparison);  
    
  end
* Table =  
    
   4×4 table  
    
   Movie\_A Movie\_B Movie\_C Movie\_D  
   \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_  
    
   1st\_section 0.8 0.5 0.3 0.8   
   2nd\_section 0.8 0.5 0.3 0.3   
   3rd\_section3 0.3 0.3 0.3 0.8   
   4th\_section4 0.6 0.5 0.3 0.7
* [*Published with MATLAB® R2018a*](https://www.mathworks.com/products/matlab)

As you see above, all the answers of sugeno approach for solve the problem of all sections are exactly the same as what we manually calculated.

* Choquet tests
* clear;close;clc;  
    
  colNames = {'Movie\_A' 'Movie\_B' 'Movie\_C' 'Movie\_D'};  
  rowNames = {'1st\_section', '2nd\_section', '3rd\_section3', '4th\_section4'};  
  Table = array2table([Choquet([0.3,0.6,0.7,0.3],[1.0,0.8,0.1,0.8]),...  
   Choquet([0.3,0.6,0.7,0.3],[0.5,0.5,0.3,0.7]),...  
   Choquet([0.3,0.6,0.7,0.3],[0.3,0.3,0.2,0.4]),...  
   Choquet([0.3,0.6,0.7,0.3],[0.3,0.3,0.8,0.8]);...  
   Choquet([0.3,0.6],[1.0,0.8]),...  
   Choquet([0.3,0.6],[0.5,0.5]),...  
   Choquet([0.3,0.6],[0.3,0.3]),...  
   Choquet([0.3,0.6],[0.3,0.3]);...  
   Choquet([0.7,0.3],[0.1,0.8]),...  
   Choquet([0.7,0.3],[0.3,0.7]),...  
   Choquet([0.7,0.3],[0.2,0.4]),...  
   Choquet([0.7,0.3],[0.8,0.8]);...  
   Choquet([0.6,0.7],[0.8,0.1]),...  
   Choquet([0.6,0.7],[0.5,0.3]),...  
   Choquet([0.6,0.7],[0.3,0.2]),...  
   Choquet([0.6,0.7],[0.3,0.8]),...  
   ],'VariableNames',colNames,'RowNames',rowNames)  
    
    
  function res = Choquet(mu, f)  
  syms x  
    
  n = size(mu,2);  
    
  lambdaCalFunc = 1;  
  for i=1:1:n  
   lambdaCalFunc = lambdaCalFunc\*(1+ x\*mu(1,i));  
  end  
  C = coeffs(expand(lambdaCalFunc - x - 1),'All');  
  digits(6)  
  r = vpa(roots(C));  
  mr = size(r,1);  
    
  lambda = 0;  
  if sum(mu) == 1  
   lambda = 0;  
  elseif sum(mu) > 1  
   for i=1:1:mr  
   if isreal(r(i,1)) && r(i,1)>-1 && r(i,1)<0  
   lambda = r(i,1);  
   end  
   end  
  else  
   for i=1:1:mr  
   if isreal(r(i,1)) && r(i,1)>0  
   lambda = r(i,1);  
   end  
   end  
  end  
    
  F = [mu' f'];  
  A = zeros(n,2);  
  for i=1:1:n  
   [a, index]=max(F(:,2));  
   A(i,:) = [F(index,1) a];  
   F(index,:)=[];  
  end  
    
  GAi = zeros(n,2);  
  Gi = 0;  
  for i=1:1:n  
    
   Gi = Gi + A(i,1) + lambda\*Gi\*A(i,1);  
    
   GAi(i,:) = [Gi, A(i,2)];  
  end  
    
  result = 0;  
  previousVal = 0;  
  for i=1:1:n  
   result = result+(GAi(n-i+1,2) - previousVal)\*GAi(n-i+1,1);  
   previousVal = GAi(n-i+1,2);  
  end  
  res = result;  
    
  end
* Table =  
    
   4×4 table  
    
   Movie\_A Movie\_B Movie\_C Movie\_D  
   \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_  
    
   1st\_section 0.74355 0.52673 0.31336 0.70398  
   2nd\_section 0.86 0.5 0.3 0.3  
   3rd\_section3 0.31 0.42 0.26 0.8  
   4th\_section4 0.52 0.42 0.26 0.65
* [*Published with MATLAB® R2018a*](https://www.mathworks.com/products/matlab)

As you see above, all the answers of sugeno approach for solve the problem of all sections are approximately the same as what we manually calculated.

It’s time to compare the results of 1st part vs 2nd part.

**Comparison Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Movie | Approach | Result of part 1 | Result of matlab |
|  | A | Sugeno |  |  |
|  |  | Choquet |  | 0.74355 |
|  | B | Sugeno |  |  |
| Section A |  | Choquet |  | 0.52673 |
|  | C | Sugeno |  |  |
|  |  | Choquet |  | 0.31336 |
|  | D | Sugeno |  |  |
|  |  | Choquet |  | 0.70398 |
|  | A | Sugeno |  |  |
|  |  | Choquet |  | 0.86 |
|  | B | Sugeno |  |  |
| Section B |  | Choquet |  | 0.5 |
|  | C | Sugeno |  |  |
|  |  | Choquet |  | 0.3 |
|  | D | Sugeno |  |  |
|  |  | Choquet |  | 0.3 |
|  | A | Sugeno |  |  |
|  |  | Choquet |  | 0.31 |
|  | B | Sugeno |  |  |
| Section C |  | Choquet |  | 0.42 |
|  | C | Sugeno |  |  |
|  |  | Choquet |  | 0.26 |
|  | D | Sugeno |  |  |
|  |  | Choquet |  | 0.8 |
|  | A | Sugeno |  |  |
|  |  | Choquet |  | 0.52 |
|  | B | Sugeno |  |  |
| Section D |  | Choquet |  | 0.42 |
|  | C | Sugeno |  |  |
|  |  | Choquet |  | 0.26 |
|  | D | Sugeno |  |  |
|  |  | Choquet |  | 0.65 |

As we see the above table, all the results of Sugeno are exactly the same in both manual and matlab calculation. Also, this is true for Choquet approach but matlab calculates the answers of these approach with more accuracy.

* Report codes
* Sugeno

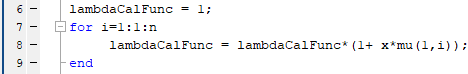
First, I define variable x as the lambda parameter in the equation . So now we have instead of previous equation.



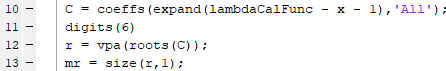
Then found the column size of f (or mu) array by the command as shown below.



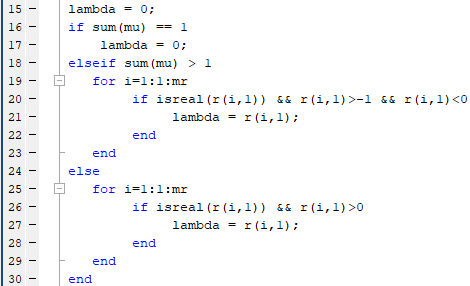
It’s time to form the phrase .



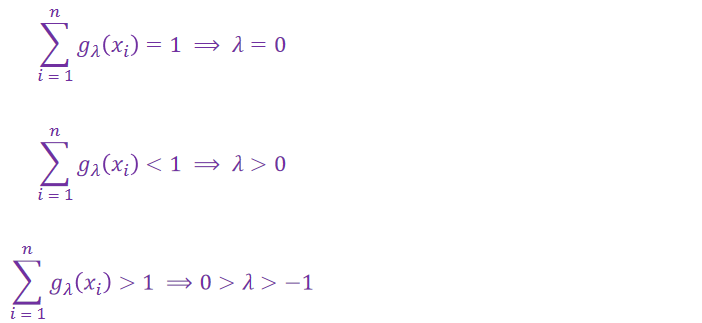
Then we calculated all the answers of the equation and also calculate the size of answer vector in order to find the correct parameter of lambda by calculation g’s summation.



We are ready to calculate the parameter lambda.



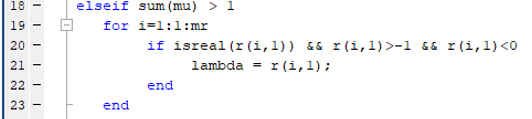
It’s exactly the same as what T.A told us. Here I remind it.



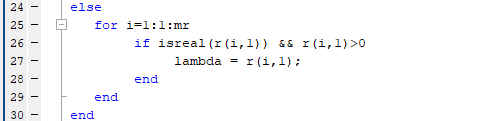
As we see here, if summation of g’s is equal to one, lambda parameter should be exactly equal to zero.



Else if the summation is higher than one, lambda should be between -1 and 0.

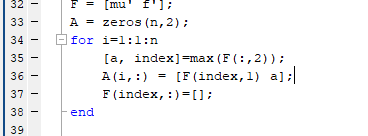


Finally, if the summation is lower than one, the parameter should be higher than zero.

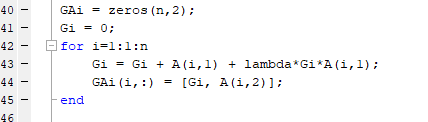


Adding the condition isreal() help us throw out the answers which are not real.

Sorting fs (matrix F(:,2)) and keeping it’s g, place in a new matrix named A. I use max function for this section in order to find the highest f and it’s index for find it’s g then delete this item in line 37 and act on the rest of matrix. Then create and place the vector in the new matrix (A).

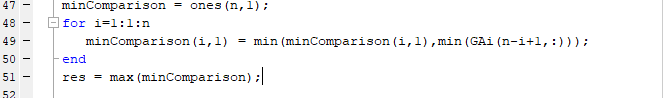


Now we can calculate G(Ai)s as shown below.



GAi indicates G(Ai)s and Gi is the previous value of GAi which equal to G(Ai-1) for i>1 and is equal to zero for i=1.

Finally we are able to use Sugeno formula as below.



As you see here, I create a vector named minComparison which valued 1 for all of its elements which is the highest possible amount. And then compare it to fi and it’s G(Ai) and then place it in minComparison vector.

Now we find the maximum of minComparison element by using max command and then put it in the res variable of Sugeno function.



* Choquet

Explanation of code line 1 to code line 46 is the same as what we said in Sugeno code. So we repeat these lines again.

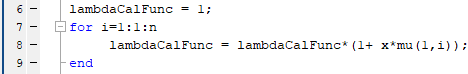
First, I define variable x as the lambda parameter in the equation . So now we have instead of previous equation.



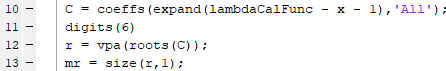
Then found the columns size of f (or mu) array by the command as shown below.



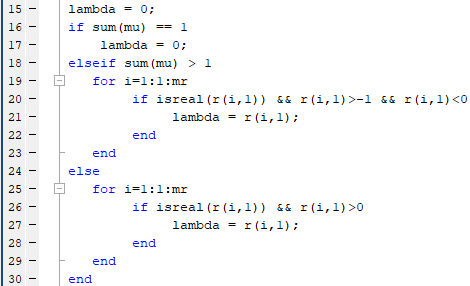
It’s time to form the phrase.



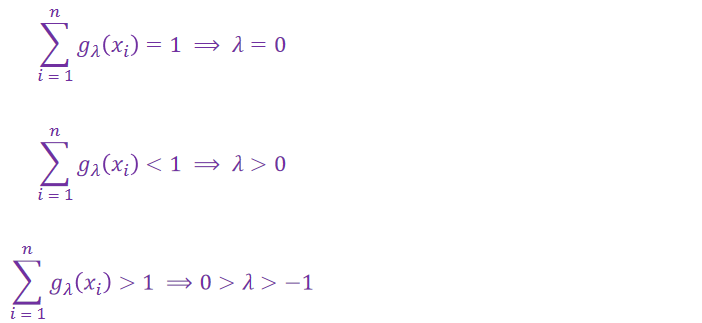
Then we calculated all the answers of the equation and also calculate the size of answer vector in order to find the correct parameter of lambda by calculation g’s summation.



We are ready to calculate the parameter lambda.



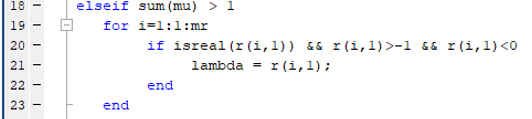
It’s exactly the same as what T.A told us. Here I remind it.



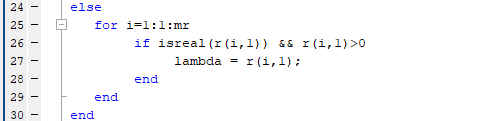
As we see here, if summation of g’s is equal to one, lambda parameter should be exactly equal to zero.



Else if the summation is higher than one, lambda should be between -1 and 0.

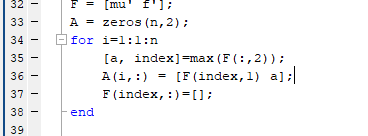


Finally, if the summation is lower than one, the parameter should be higher than zero.

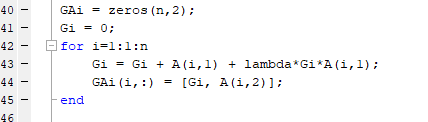


Adding the condition isreal() help us throw out the answers which are not real.

Sorting fs (matrix F(:,2)) and keeping it’s g, place in a new matrix named A. I use max function for this section in order to find the highest f and it’s index for find it’s g then delete this item in line 37 and act on the rest of matrix. Then create and place the vector in the new matrix (A).

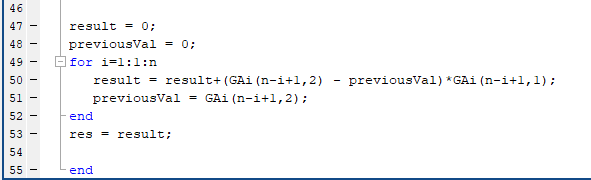


Now we can calculate G(Ai)s as shown below.

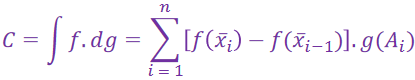


GAi indicates G(Ai)s and Gi is the previous value of GAi which equal to G(Ai-1) for i>1 and is equal to zero for i=1.

Finally we are able to use Choquet formula as below.



As you see here, I use a parameter named result which valued 0 at begin. And then I calculated the answer similar to the formula shown below.



Also I use previousVal for keeping the value of previous g(Ai) which is equal to zero at begin. Then I put the result in the output variable (res) of Choquet function.

