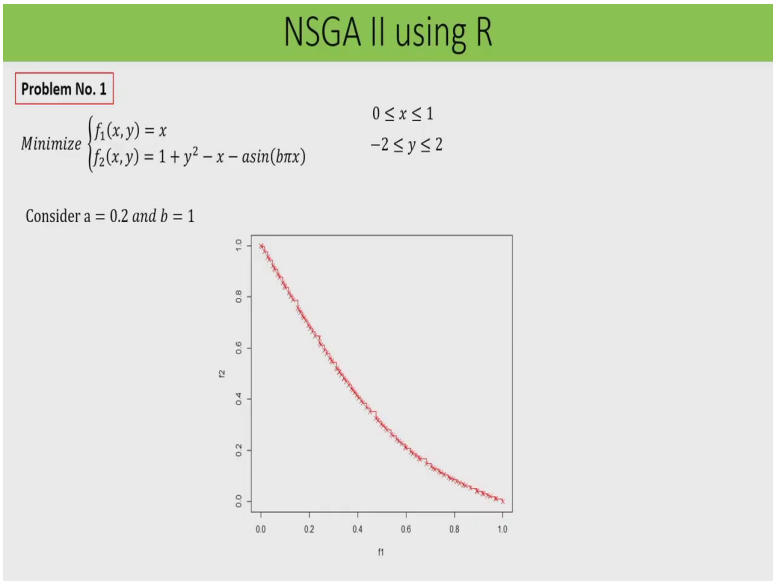
Optimization Methods for Civil Engineering
Prof. Rajib Kumar Bhattacharjya
Department of Civil Engineering
Indian Institute of Technology, Guwahati

Lecture - 34
NSGA II Using R

Hello student welcome back to the course on Optimization Methods for Civil Engineering. So, today we will solve few problems using NSGA II algorithms. We will solve 4 multi objective optimization problems here, 2 problem without constrain; that means, unconstrained multi objective optimization problem and 2 with constrains. So, let us see the 4 problems.

(Refer Slide Time: 00:59)

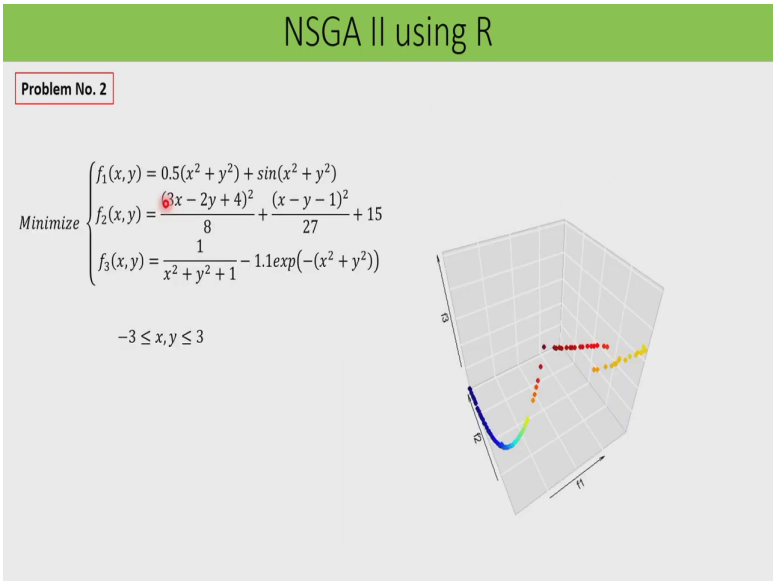


The first problem we have 2 objective functions so, both are minimization type. So, you can see f 1 equal to x and f 2 equal to 1 plus y squared minus x minus asin b pi x. So, we have considered a equal to 0.2 and b equal to 1 and ranges for x and y x is between 0 and 1 and y is

between minus 2 and plus 2. So, y is varying between minus 2 and plus 2. So, range of y is minus 2 and plus 2 and range of x is 0 and 1.

So, if I solve this problem using NSGA II. So, I should get this pareto optimal font. So, here in the x axis we have objective function 1 and y axis objective function 2 and I should get this pareto optimal font.

(Refer Slide Time: 01:55)

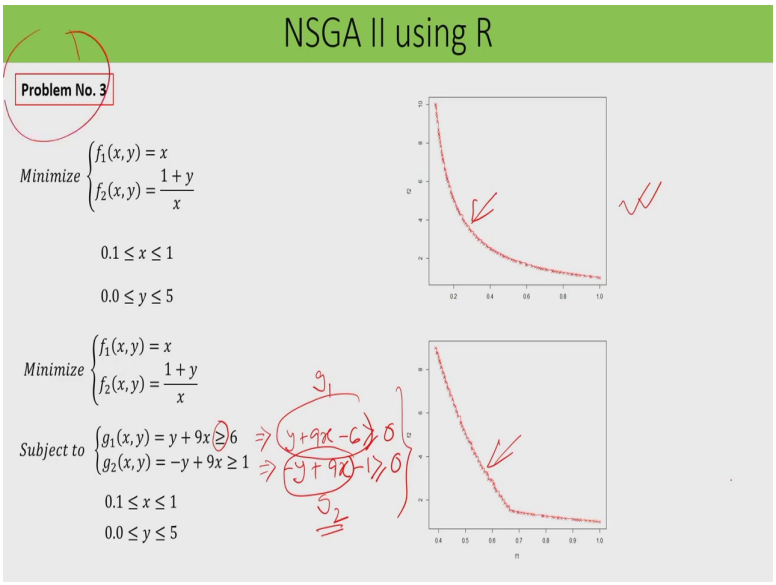


The second problem has 3 objective function so, all of them are minimization type. So, you can see that f 1 is 0.5 x square plus y square plus sin within bracket x square plus y square. And objective function 2 is thrice x minus twice y plus 4 whole square divided by 8 plus x minus y minus 1 whole square divided by 27 plus 15.

And there are another objective function that is f3 and that is one by x square plus y square plus 1 minus 1 point 1 e x p minus of x square plus y square ok. And the range for x and y is minus 3 to plus 3; that means, x is between minus 3 and plus 3 and y is also between minus 3 and plus 3.

So, if I plot these 3 objective function and I should get this pareto optimal font. So, this is in the x axis it is f 1 then y axis this is f 2 and z axis f 3. So, I should get this pareto optimal font. So, NSGA can solve more than 3 objectives ok so we can we can have 4th objective function 5th objective function like that, but here we have shown up to the 3rd one so, that we can plot it. So, this is the pareto optimal font for this particular problem. Then the 3rd problem is a constrain multi objective optimization problem.

(Refer Slide Time: 03:31)



So, you can see that it is a minimization type function that minimize f 1 equal to x and f 2 equal to 1 plus y by x and range for x 1 is between point 1 and 1; that means, x is between point 1 and 1 and y is between 0 and 0 and 5. So, it has 2 constraints g 1 and g 2 so you can see g 1 equal to y plus 9x it is a greater than equality type constrain. So, y plus 9x greater than 6 and g 2 is minus y plus 9x greater than equal to 1 ok.

And as I said that range for x 1 is 0.1 and 1 and that for y is 0 and 5. So, I have solved this problem with constrain and without constrain. So, you can see that first graph so, whatever I have shown here. So, this is without constrain; that means, I have solved this problem using NSGA 2 and I should get this pareto optimal font and if I use these constrain that is g 1 and g 2 and you can you should get this particular pareto optimal font.

So, when we are using NSGA so, you note that the objective function should be minimization type for the package we are using in r. So, objective function should be minimization type; that means minimize so, anyway. So, I can convert a minimization problem to maximization problem or maximization problem to minimization problem, but in this particular package.

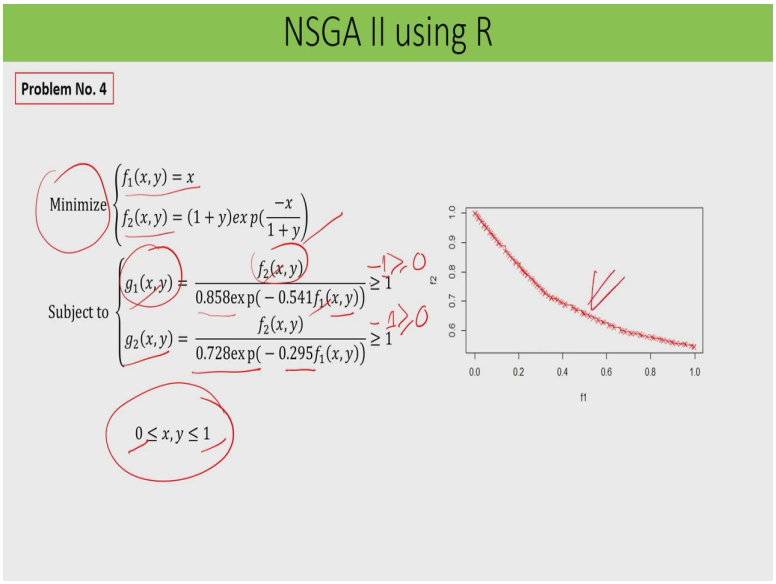
So, we should use minimization type; that means, if you have a maximization type problem then I should multiply it by minus 1 and convert it to minimization type. Similarly, the constrain should be greater than equality type ok. So, in this case this is y plus 9x greater than 6 and similarly g 2 equal to minus y plus 9x greater than equal to 1.

So, as I said that for this particular package that we will use in r. So, it can take so, it can take greater equality type. So, therefore, I will write this constrain like this y plus 9x minus 6 greater than equal to 0. And similarly this one I will write minus y plus 9x minus 1 greater than equal to 0.

So, therefore, so this is your g 1 and this value is your g 2. So, we will do that one so, I will use g 1 and g 2 like that. So, if we solve this problem as I said that this is unconstrained multi objective optimization problem then I should get this particular pareto optimal font and when

we are using these two constraints so, we should get this particular pareto optimal font. So, this is your problem 3.

(Refer Slide Time: 06:35)



So, now we have another problem that is problem 4 and this is also a minimization type optimization problem. So, minimize f 1 equal to x and f 2 equal to 1 plus y e x p within bracket minus x by 1 plus y. So, this is objective function 2. And then we have two constrain the constrains are this is f 2; f 2 divided by 0.858 exp minus 0.541 f 1 x 1.

So; that means, in the to calculate constrain one so, I have to calculate first f 2 and f 1 and it is greater than equality type. And similarly the g 2 that is the second constrain is f 2 x y f 2 x and y this is 0.728 exp minus 0.295 f 1 it is also greater than equality type.

So, what I will do basically so as I said that I will convert it. So, I will put this is minus 1 and then greater than equal to 0 and that also I will put minus 1 greater than equal to type 0 ok. So, I can write and ranges for this x and y is between 0 and 1; that means, x lower limit of x is 0 and upper limit of x is 1 lower limit of y is 0 and upper limit of x is 1. So, if I use NSGA II so I should get this particular pareto optimal font ok.

So, now we will solve all these 4 problems. So, here we have to define the objective function as you have seen this objective function should be minimization type ok. So, both f 1 f 2 f 3 whatever you have actually so it should be minimization type objective function. And constrain should be greater than type; that means, g 1 should be greater than equal to 0 and g 2 should be greater than equal to 0.

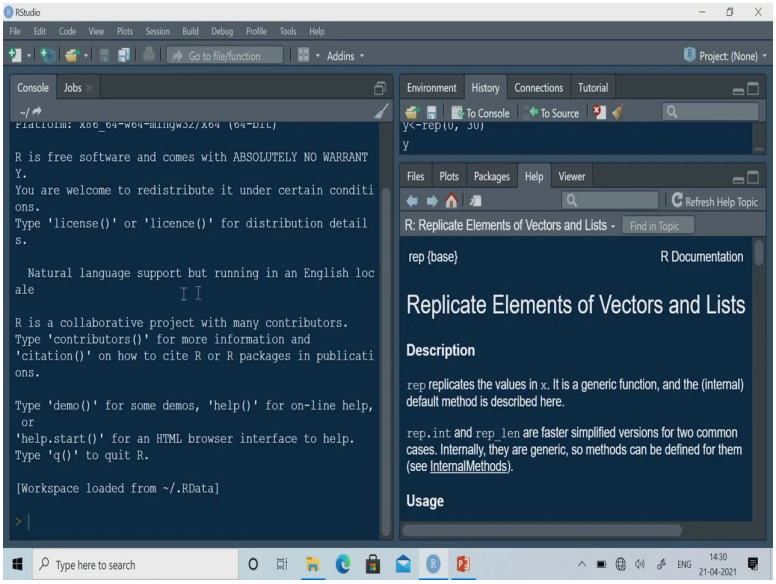
So, if your constrain is less than equality type then you have to convert it to greater than equality type and similarly if your objective function is a maximization type objective function then you have to convert it to minimization type objective function ok.

So, that we will do. So, in this particular problem whatever we have considers. So, we have considered 4 problems and objective function of this or these problems are minimization type and constrains are greater than equality type constrain. So, let us open r and solve all these 4 problems.

So, now I will solve the problem 1. So, as you have seen so, it has 2 objective functions that is minimization type that is f 1 equal to x and f 2 equal to 1 plus y square minus x minus asin b pi x value of a is 0.2 and value of b is 1 and range for x is between range for x is 0 and 1 and range for y is minus 2 plus 2 ok.

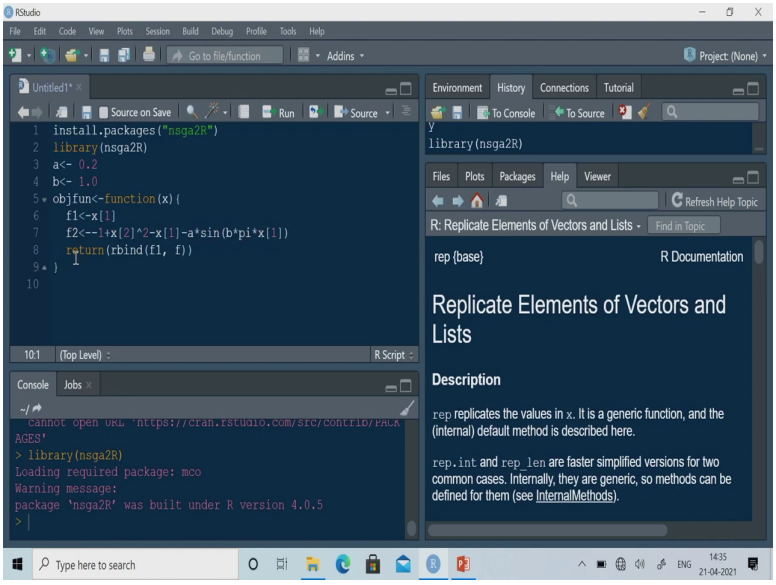
So, let us solve this problem using R.

(Refer Slide Time: 09:55)



So, I have already opened these R's. So, let us open the file so, I would like to open a new file ok. So, this is your R script.

(Refer Slide Time: 10:07)



And. So, what I have to do here so, I have to use nsga2R package. So, if you have not installed it so, you have to install it. So, install package and package name is nsga2R ok. So, this is the package you have to install. So, if it is not installed, but in my case it is already installed in my computer so I will not install it.

So, what I will do I will use the library. So, the library name is ns nsga2R. So, let me execute this thing so, I have used the nsga2R. Now, in this case what first we have to write? We have to write the objective function. So, objective function there are two objective function as I said. So, you please let us write the objective function.

So, this is obj I am writing objfun. So, this is the name of the objective function I am just putting it. So, this is the function ok, function of x and it has 2 objective functions so that is

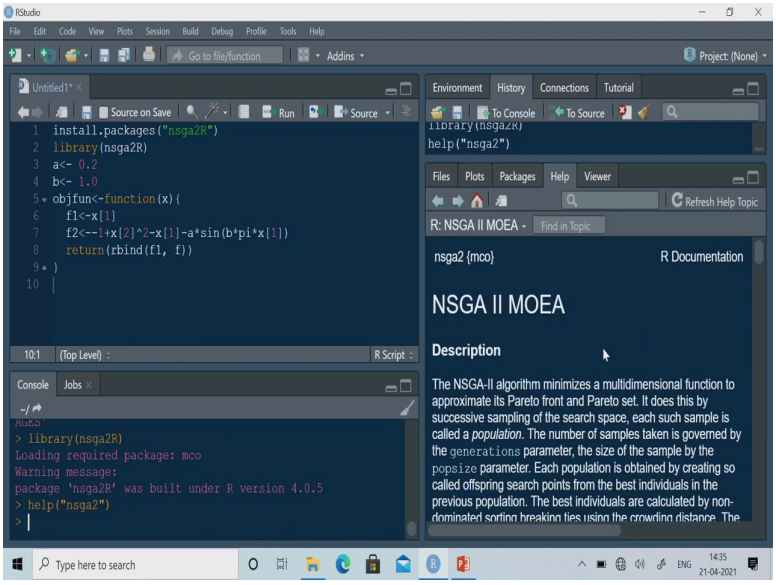
f1 so, f1 is x 1 ok. And we have f2, f2 is that is minus 1 plus y; that means, x 2 square x 2 square minus x 1 ok then, minus asin that is sorry this is a star sin.

So, yes this is b star pi star x 1 ok. So, here we have to define what is the value of a and b so, I can define that a is 0.2 and b is 1 ok so, 1.0 ok. So, this you have to define because I am using a and b here so let us say [FL]. So, this is minus 1 f2 equal to f.

So, let us say f1 equal to x 1 ok that is your x and f2 is minus 1 plus y square y square means here I am writing x 2 square minus x 1 minus asin a into sin b pi x so, x 1 ok. So, this is my objective function and then I have to return these two. So, return so, rbind.

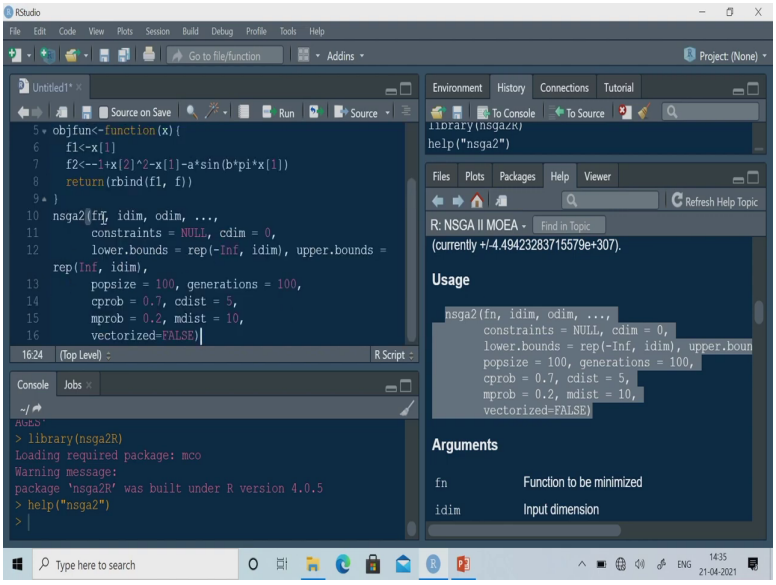
So, I have to return both f1 and f2 so, f1 and f2. So, I am writing the objective function so, here I have defined objective function one and objective function two so, both are minimization type. Now, what I have to do I have to use nsga2 ok.

(Refer Slide Time: 13:57)



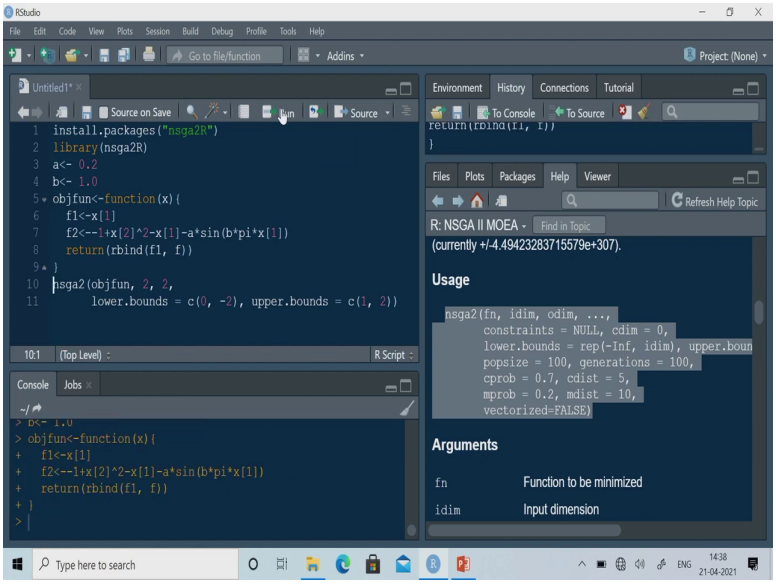
So, you can check you can go to help ok help nsga2 ok so let me check. So, this is nsga2.

(Refer Slide Time: 14:11)



So, this is the nsga2 function so I will use this one. So, what I will do I can just I will copy this one ok.

(Refer Slide Time: 14:25)



So, here the function name is in this case objective function obj, in our case objective function. The how many objective function we have that 2 then what is the dimension of this objective function so, we have 2 variables so, that is 2. So, there is no constrain here so, I am not using constrain. So, there is no constrain and so, you need not to use these one. So, by default these values is null and 0 so, I am not using this one.

Then lower bound and upper bound we have to define so lower bound I am defining. So, I am lower bound I have to define so, this is lower bound. So, lower bound is 0 and 2 ok and upper bound is so, upper bound is 1 and 2 ok. So, this is your upper bound ok so, I am defining lower bound and upper bound ok. So, what I am doing here I have defined objective function.

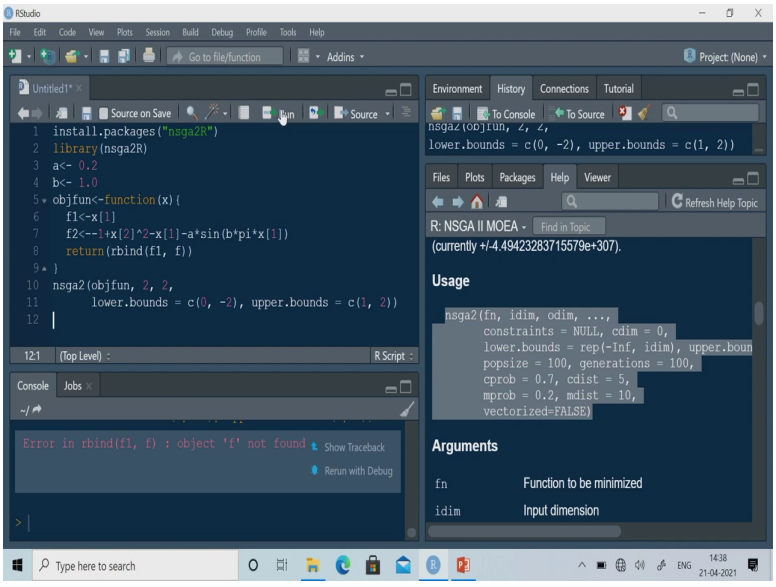
So, this is the objective function then number of functions are 2 so f1 and f2. So, number of variables are 2 then lower bound is 0 and 2, sorry lower bound is 0 and minus 1 minus 2 and

upper bound is 1 and 2 so, upper bound is 1 and 2 and you can define population size generation crossover probability. So, other things you can define, but I am not defining here so, I will use the default value. So, I can change it this these values, but I will use the default value.

So, my problem definition is over here. So, objective function I have define. So, this is the objective function name of the objective function is objfun number of functions are 2 number of variables are 2 lower bound I have define that is 0 and minus 2 upper bound I have defined that is 1 and 2 ok. And other things I am taking the default value so I hope this is fine. So, if there is no problem I should get the solution so let me check.

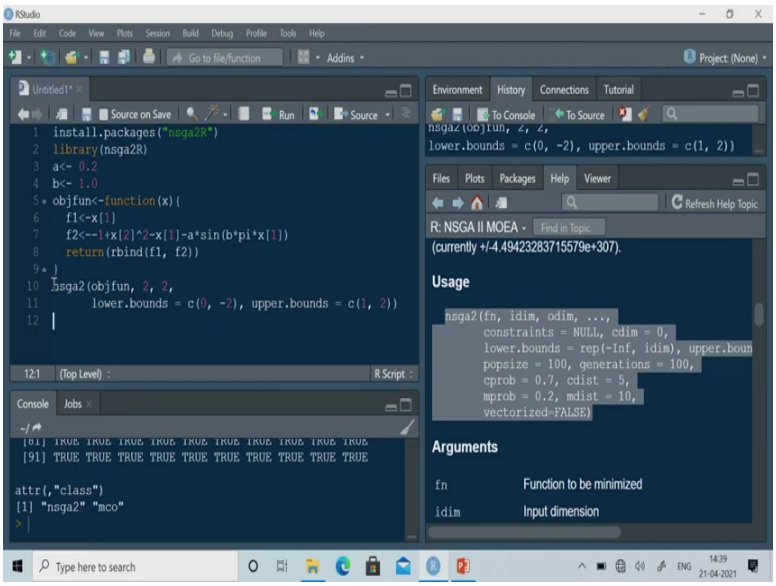
So, I have to run the objective function. So, before that I have to execute these two line that is a equal to 0.2 and b equal to 1 and then this is the objective function and then this is the nsga.

(Refer Slide Time: 17:09)



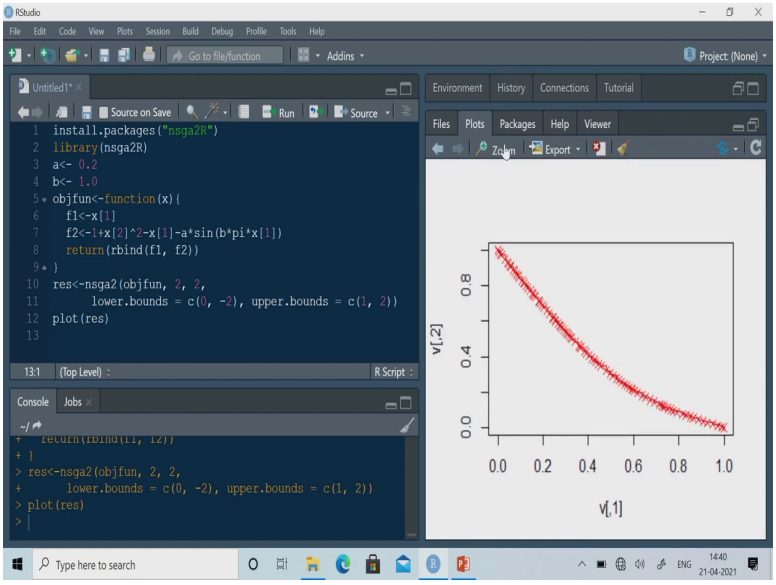
I think there is some error here let me check what is the error ok. So, I think I have not defined so, I have to define f2 ok.

(Refer Slide Time: 17:17)



So, this is f1 and f 2 so, let me execute it again yeah. So, I am getting this solution and I can plot it or I can store it in result somewhere another variable. So, this is res ok so I am just storing it and then if I execute this one and then I can plot it plot res ok. So, let us see, I should get the pareto optimal font.

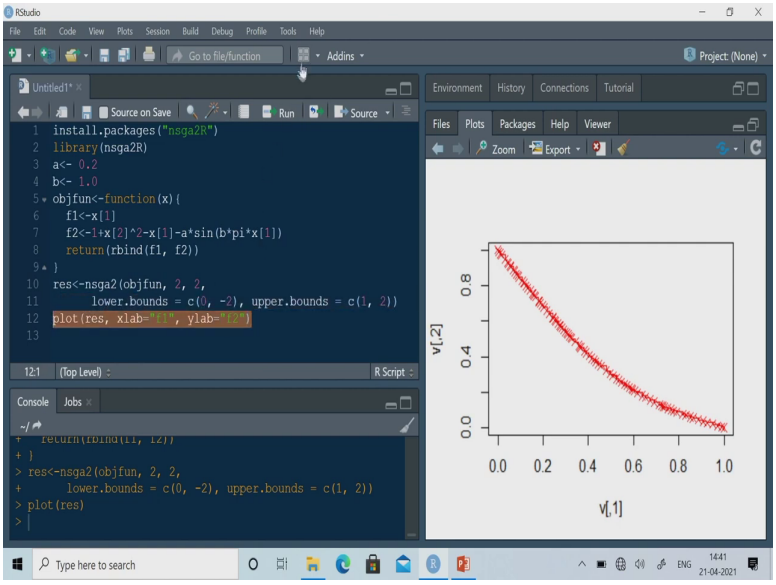
(Refer Slide Time: 17:51)



So, here I am getting the pareto optimal font sorry I did one mistake here. So, this is plus 1. So, this is plus 1 then x 2 square; that means, y square minus. So, 1 plus y square minus x minus asin b pi x so, I hope this is correct. So, this is 1 plus y square minus x minus asin b pi x now I think this is correct. So, let me check this one ok yeah this is correct now.

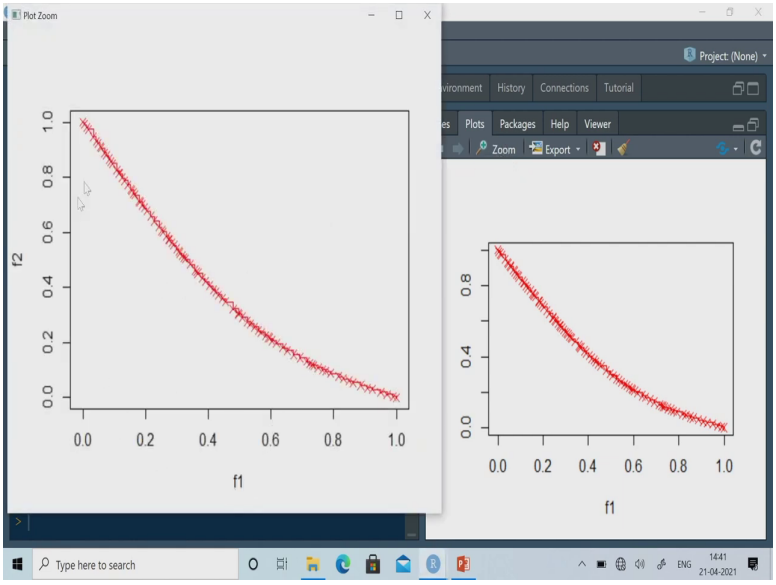
So, you just see I am getting the pareto optimal font. So, this is the pareto optimal font I am getting so, you can see. So, this is the pareto optimal font here so, I am also getting the similar pareto optimal font. So, I can define here this is objective function 1 in x axis and objective function 2 in y axis.

(Refer Slide Time: 18:55)



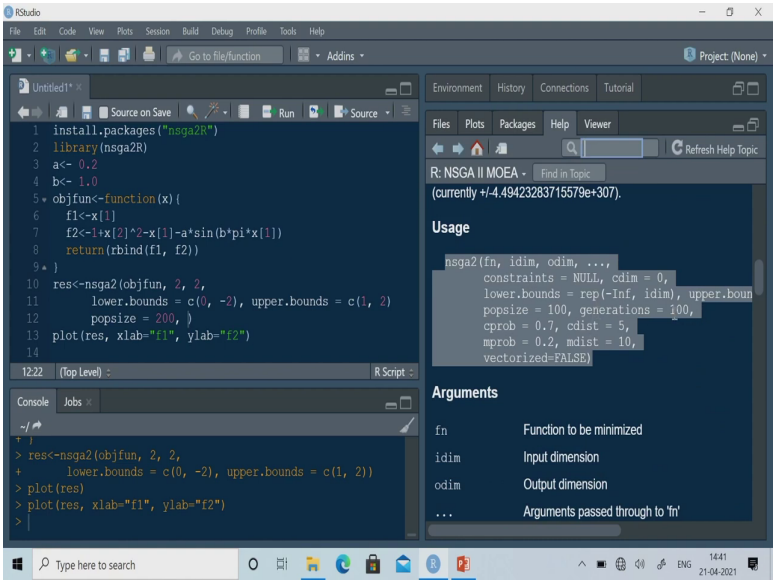
So, I can define here this is xlab x level equal to f1 and ylab equal to f2. So, now, execute this one.

(Refer Slide Time: 19:19)



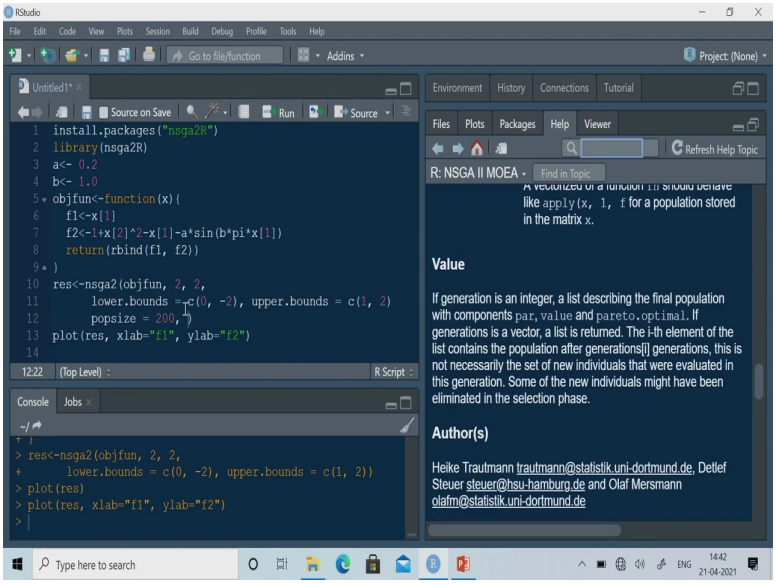
So, I am getting f1 and f2. So, this is f1 and this is f2 and this is the pareto optimal font ok. So, I got this solution. So, as I said so, I can define the population size and other thing.

(Refer Slide Time: 19:35)



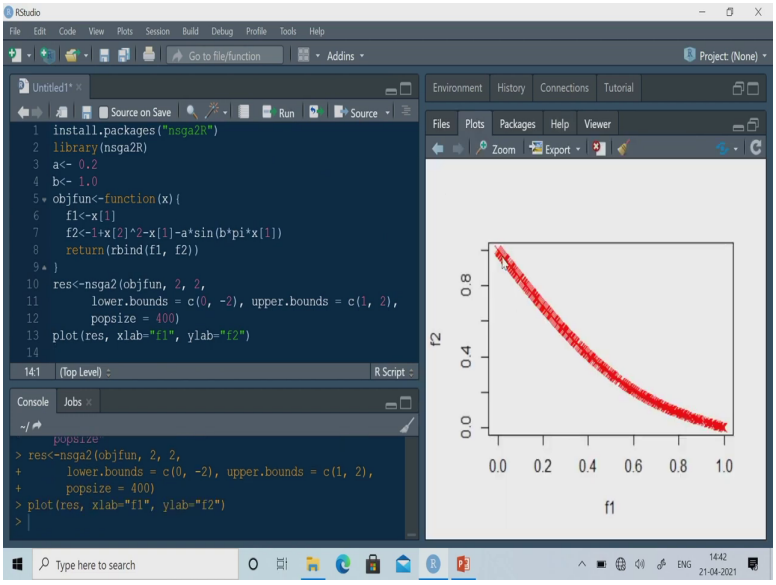
So, I can define here this is popsize ok so, popsize is 200 and other things I can define here. So, other values so, generation is by default it is 100. So, I can change it then cross over probability so, you can change it.

(Refer Slide Time: 19:51)



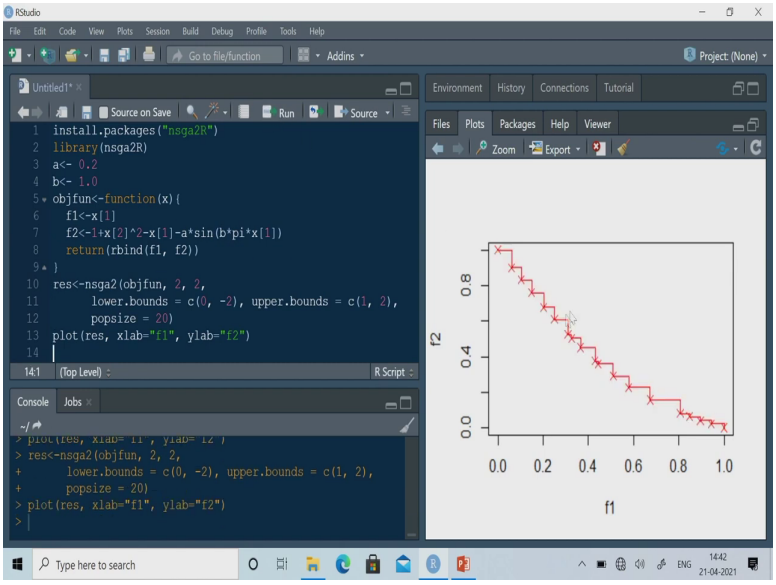
So, these are the other parameters so that you can change it from here ok. And so, I can run it again with population size of 200.

(Refer Slide Time: 20:07)



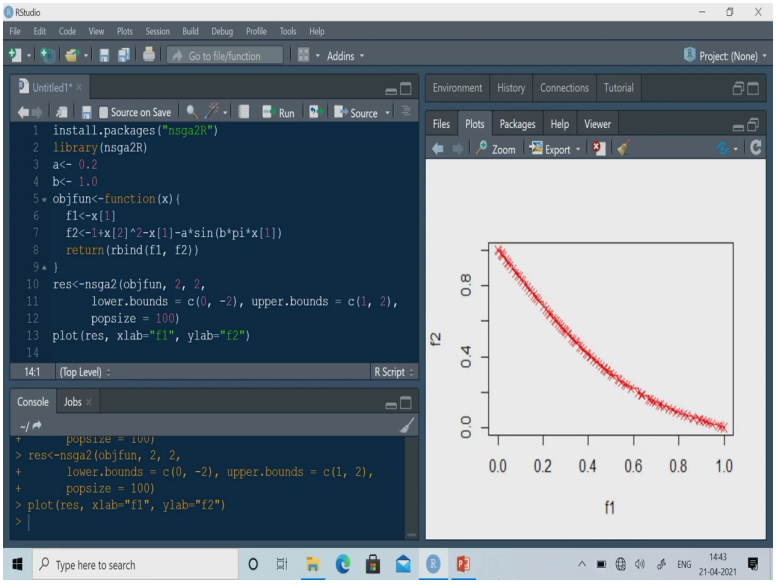
Now, or make it 400 yeah. So, you just see now there are more your population on the pareto optimal font. So, what will happen if I put only 20 ok.

(Refer Slide Time: 20:23)



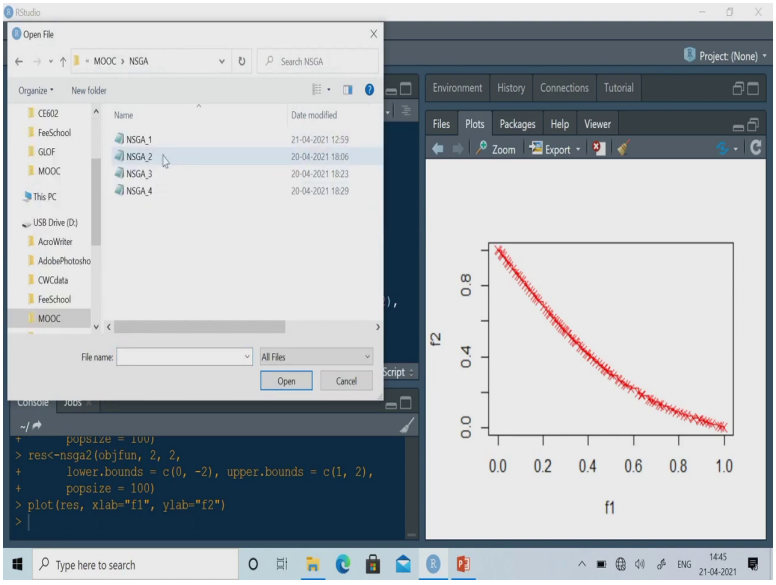
So, you will see that there are less solution in the pareto optimal font ok yeah. So, you just see there are few solution.

(Refer Slide Time: 20:37)



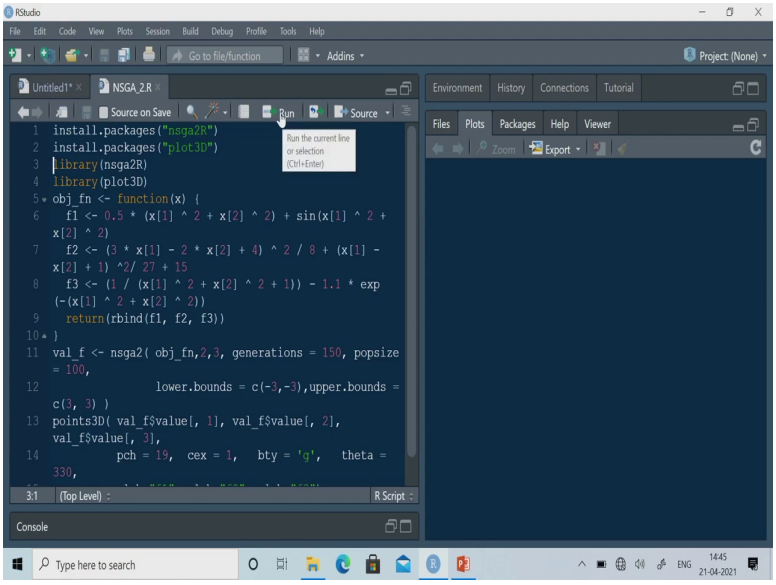
So, if I increase my population size. So, I will get yeah I will get the entire I will get more population on this form. So, I am getting this one. Now, let us solve the second problem. So, in the second problem we have 3 objective functions. So, here we have defined this objective function 1 objective function 2 another is objective function 3 and x y is varying between minus 3 and plus 3 and I should get this pareto optimal font. So, let me go to R again. So, I have already done this problem so, let me open that one.

(Refer Slide Time: 21:29)



Let me open the second problem. So, this is I have written nsga2 and just let me open it ok.

(Refer Slide Time: 21:35)



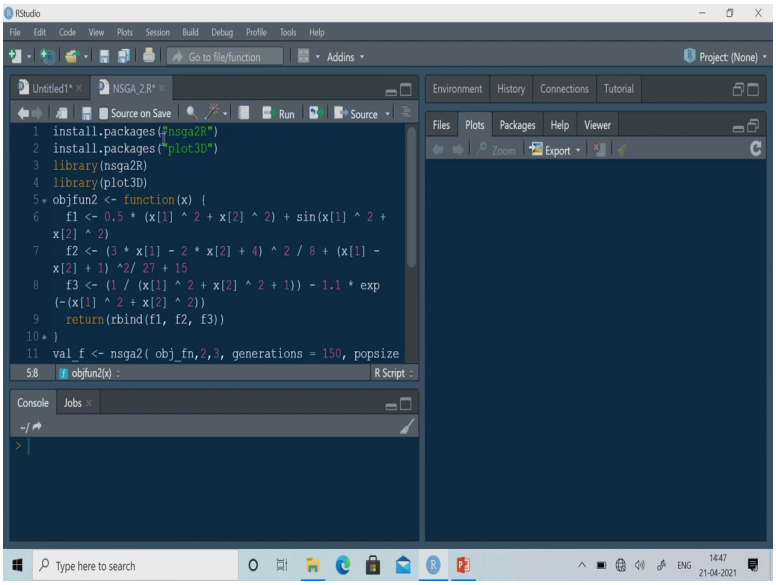
So, here let me define this one. So, here nsga2 anyway so we need it so, if you have not installed you please install it. And to plot it because this is a three dimensional plot means we will get a we will get a three dimensional plot. So, therefore, I have installed plot3D so, that is also required. And then I need library anyway after installation so, you have to include this library. So, let me include this library nsga2R and plot3D.

So, here I have defined the objective function here. So, in the objective function so, we have total 3 objective function that is f1 f2 and f3. So, I have written here so, f1 equal to 0.5 into x 1 square plus x 2 square plus sin within sin x square x 1 square plus x 2 square.

And similarly f2 equal to minus 3 x 1 minus 2 x 2 plus 4 whole square divided by 8 plus x 1 minus x 2 plus 1 whole square divided by 27 plus 15. And f3 is 1 by x 1 square plus x 2

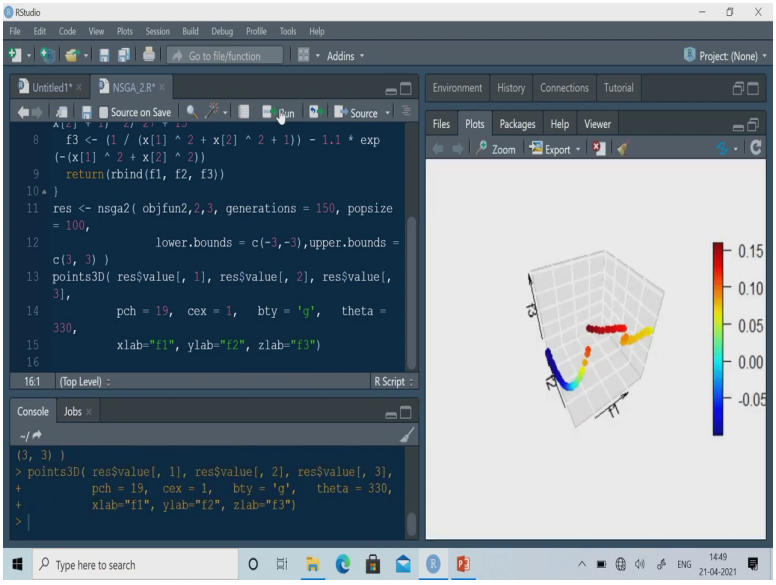
square plus 1 minus 1.1 into exp minus point minus x 1 square plus x 2 square ok. So, I have defined here f1 f2 and f3 so within this then I am returning the value of f1 f2 and f3.

(Refer Slide Time: 23:23)



So, name of this function is I am giving objective objfun and this is 2 ok the second problem. So, I hope this part is clear.

(Refer Slide Time: 23:33)



So, now so, I have use the nsga2. Now, this name of the objective function is objfunc fun2 so, objective function 2 and here we have 2 variables and 3 objective function ok. So, 2 and 3; that means, x 1 and x 2 we have and number objective functions are 3.

And generation I have used 150 population size 100 and define lower bond. So, lower bound is minus 3 and plus 3 and upper bound is 3 and 3. So, if I execute this one so, I will be executing nsga2. And then I have plotted using plot3D. So, here I have used so, this will be your res now. So, I have used res ok and this is also res ok res ok.

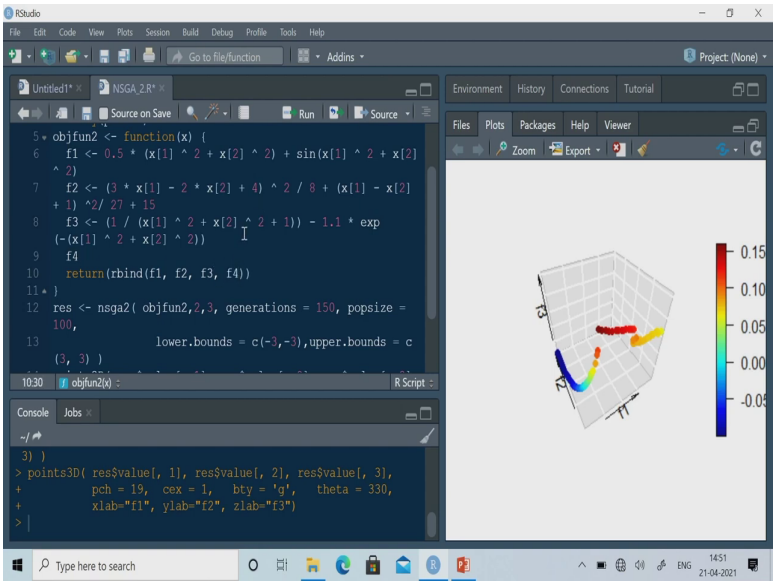
So, then let me execute this lines so, now, let me execute this objective function ok yeah and then nsga2 and then once I am getting the nsga2. So, let me plot the pareto optimal font using point 3D ok. So, I am using so, you just see so, I am getting this pareto optimal font. So, this

is a 3 dimensional plot. So, therefore, I have used points 3D and which is available under plot 3D package.

So, I am getting the solution so, you can see it. So, this is a 3D plot so, here it is f1 f2 and f3 so, I am getting this pareto optimal font of this particular problem. So, I have this is clear to you so, I can also use the 4th objective function. So, f1 f2 f3 f4 or any other objective function you have actually nsga2 is capable of taking more than 2 objective function. So, I can use it, but here I have used up to 3 that is 3rd objective function so, that I can plot the pareto optimal font.

You can also solve a problem having more than 3 objective function. So, what you have to do in the objective function you have to write the another function. Suppose I have 4th objective function.

(Refer Slide Time: 26:15)

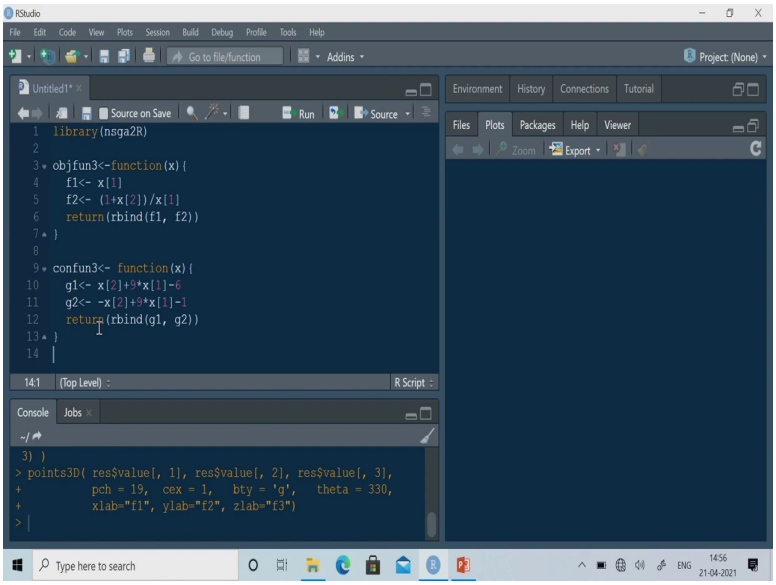


So, what I will do? So, I will write f3 here so, ok. So, f3 f4 here ok. So, I will define that one and then I have to return f4 here. So, you can try with another objective function, but in this particular problem we have considered 3 objective functions.

So, I hope this is clear so, let me solve the next problem. So, as I said then the third problem is constrained multi objective optimization problem. So, we have 2 objective functions that is f 1 and f 2 and then we have two constrain that is g 1 and g 2 and please note that so, this constrain should be greater than equality type ok. So, therefore, what I am doing here I am defining g 1 which is y plus 9x minus 9 and g2 is minus y plus 9x minus 1 ok.

So, I will define that one and let us see. So, go to R so, I would like to close it, let us open a new R script.

(Refer Slide Time: 27:25)



So, here I am because nsga2 is already there so, I will use library only nsga2R. So, what you have to define? You have to define objective function so, objective function. So, in this case this is 3 and this is a function of x and so, how many objective functions we have? So, we have two objective function that is f1 equal to x 1 and we have f2 equal to 1 plus x 2 divided by x 1.

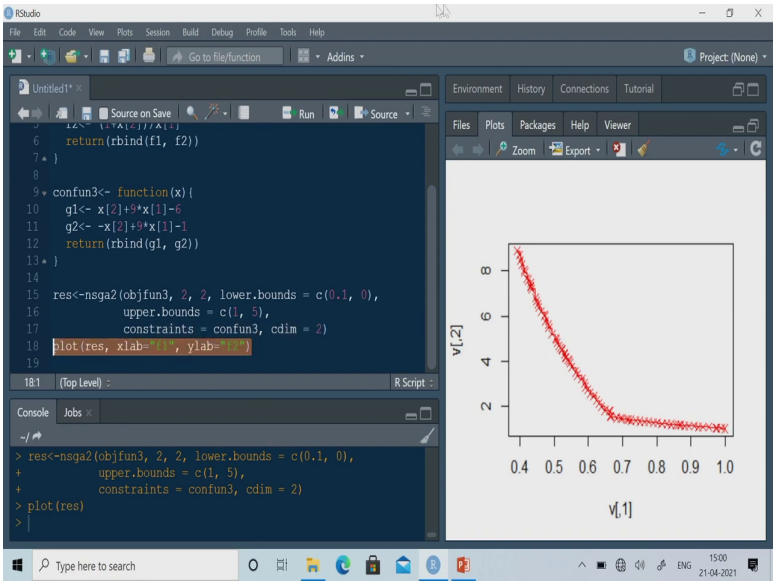
So, you can see there are 2 objective functions f1. So, f1 equal to x and f2 equal to 1 plus y by x; that means, 1 plus x 2 by x 1. So, I have define 1 plus x 2 by x 1 and f1 equal to x one. So, I have defined then I have to return this thing. So, return rbind this is f1 and f2 and this is clear.

So, now, I have to define con function. So, con function 3 here so, this should be function of x. So, we have two constrain that is your g1. So, what is g1? g1 is x 2 plus 9 star x 1 minus 6 ok. So, x 2 plus 9x 1 minus 6. So, you can see that is y plus 9x minus 6 ok y plus 9x minus 6.

So, I have defined here this is y plus 9x minus 6 and similarly I can define g2. So, g2 is minus x 2 plus 9 star x 1 minus 1 ok. So, x 2 minus x 2 plus 9 x 1 minus 1. So, you can see that is minus y plus 9x minus 1. So, this is minus y plus 9x minus 1. So, I have to return this one so, return rbind I am using and this is g1 and g2.

So, please note that I have already explained that please note that this objective function should be greater than equality type objective function. So, greater than equality type objective function if your objective function is less than equality type so, you have to convert it to greater than equality type by multiplying by minus 1. So, I have defined objective function and then I have defined constrain so, now, I can use nsga2 ok.

(Refer Slide Time: 31:15)



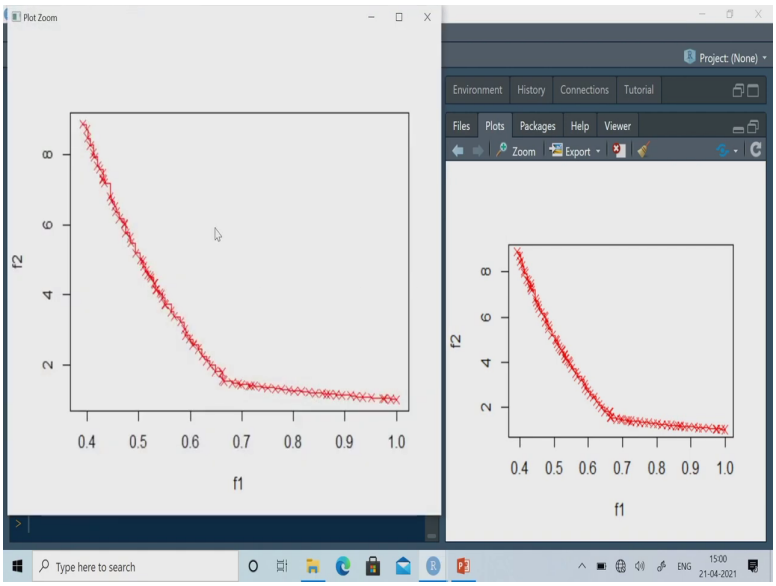
So, I am using so, result I will put nsga2 output in this result variable. So, this is nsga2 ok. So, here this is f then 2 variables 2 objective function then lower bound equal to lower bound equal to 0.1 and 0 and upper bound equal to 1 and 5. So, 1 and 5 are the upper bound, then I have to define constrain constrains equal to g and constraint dimension constraint dimension is 2; 2 means there are 2 constraints ok.

So, population is 100 by default generation is 100. So, I am not defining so, I have used the default value ok. So, I hope this is fine, so if it is fine then I should get the solution. So, let me run this particular function nsga2 ok. So, the name of the constraint function is confun3 so this is not g so, this is confun3.

I hope this is fine and objective function is not f so, objective function is objective function 3 ok. So, let me define that objective function 3 and constrain function 3. I hope this is fine now. So, let me so, I did not execute this confun ok so let me execute this one ok.

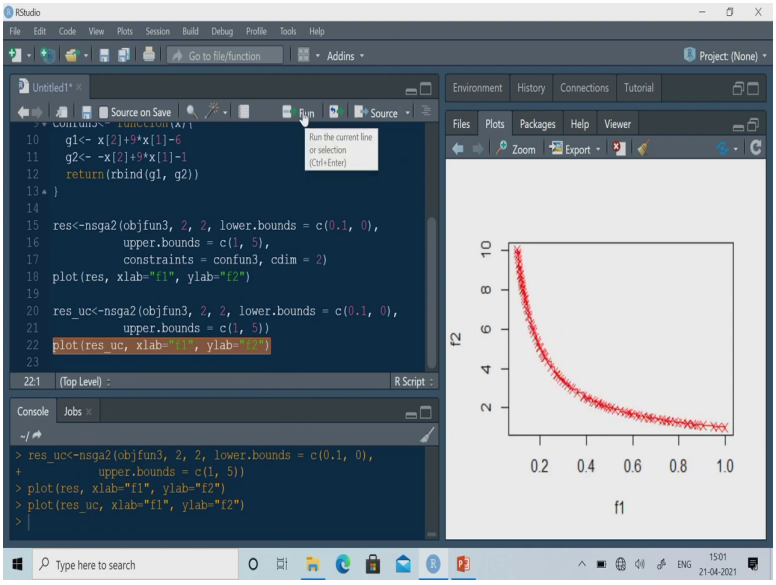
So, now, execute this yeah so I am getting the solution. So, I can plot it so, this is the plot I am using res ok. So, you can see yeah. So, I am getting I am getting this solution. So, here I can define that xlab equal to f1 and ylab equal to f2 ok.

(Refer Slide Time: 34:03)



So, I am getting the constrain solution of this problem so you can see. So, this is the pareto optimal font constrain pareto optimal font and this is your f1 and f2. So, I can also find out unconstrained one.

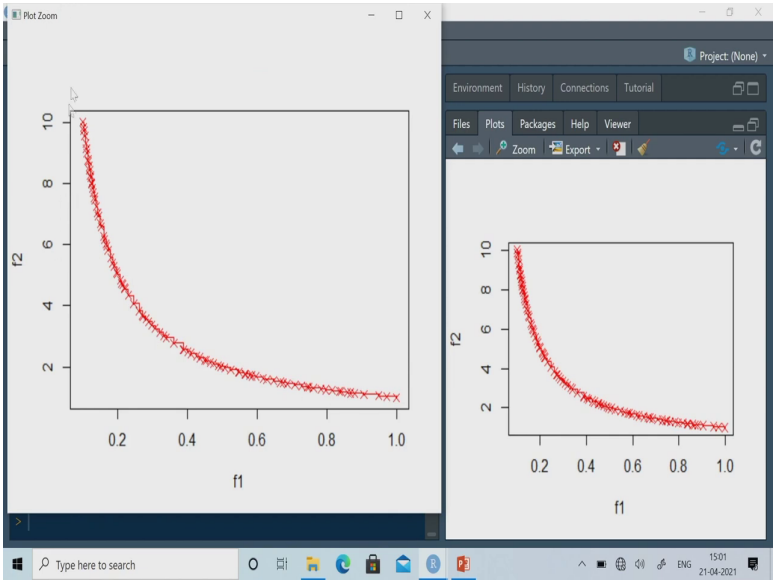
(Refer Slide Time: 34:21)



So, let me suppose if I do not consider the constrain. So, what I will I will do is basically this is not required ok the unconstrained one I would like to solve the unconstrained one ok. So, this is result unconstrained you see you can you can say unconstrained.

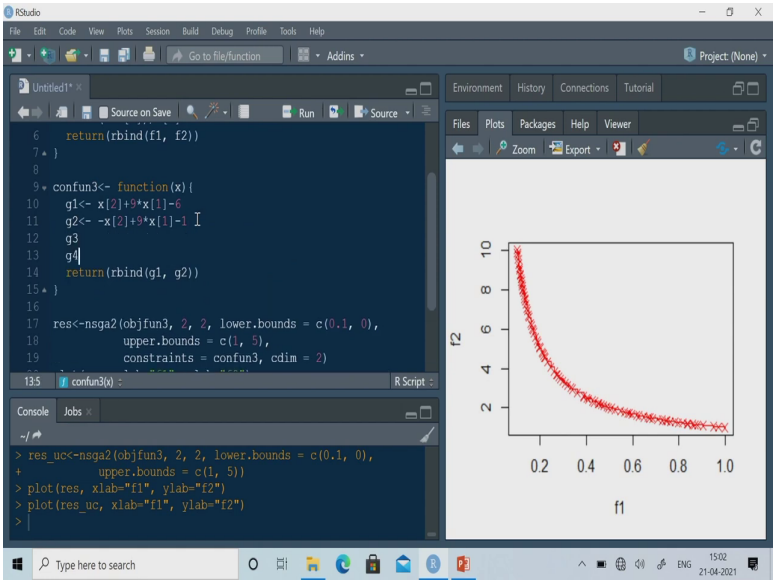
So, let me execute this one and let me plot this. So, I can copy and plot it. So, if I execute this one then I should get the unconstrained solution of this problem this is res this is res unconstrained ok. So, let me plot it yeah.

(Refer Slide Time: 35:15)



So, I am getting the unconstrained pareto optimal font of this particular problem so, this is the pareto optimal font for a unconstrained one and as you have seen so, this is for the constrained one. So, this is this solution is for the constrained one and this is for the unconstrained one. If you have more than 1 more than 2 constrains so, that also you can use.

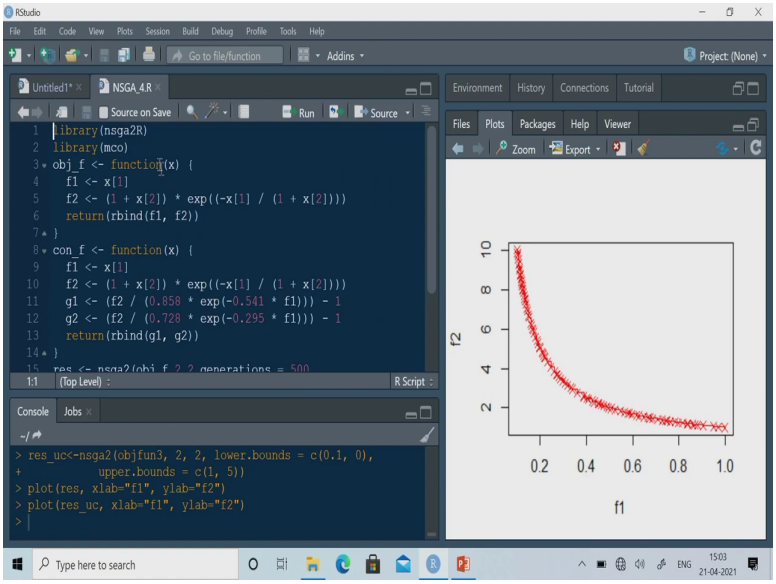
(Refer Slide Time: 35:35)



Suppose you have another one so, I can write g3 then g4. So, all the constrain I can I can define here and similarly if I have more more objective function so, I can define all these objective function ok. So, I hope this is clear to you. So, how we can handle constrained multi objective optimization problem. So, only thing you please note that these constraints are greater than equality type constraints.

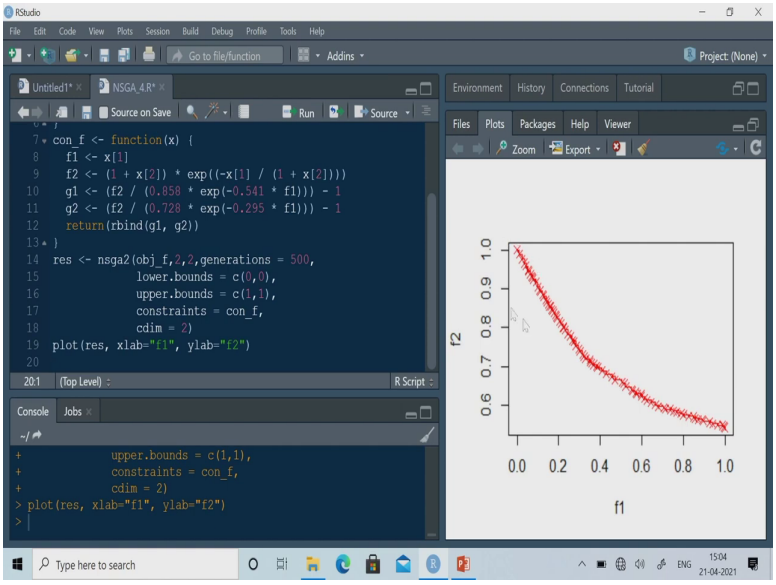
Now, let us solve the 4th problem. So, let me open the files already I did the R coding of this particular problem. So, this is the 4th one.

(Refer Slide Time: 36:15)



So, here what I need nsga2R and we have 2 objective function that is f1 and f2. So, I think this is clear to you so, let me close this thing ok.

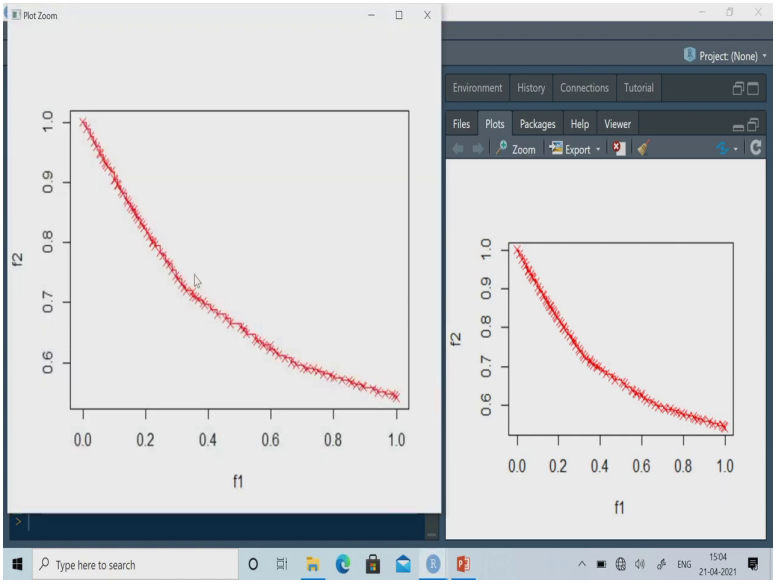
(Refer Slide Time: 36:33)



And the constrain for calculating constrain so, what you have to do? You have to calculate the objective function also. So, therefore, in the confun itself so, I am also calculating what is the objective function value and then I am defining the constrain that is g1 and g2. So, you can see then after that I am returning g1 and g2 ok.

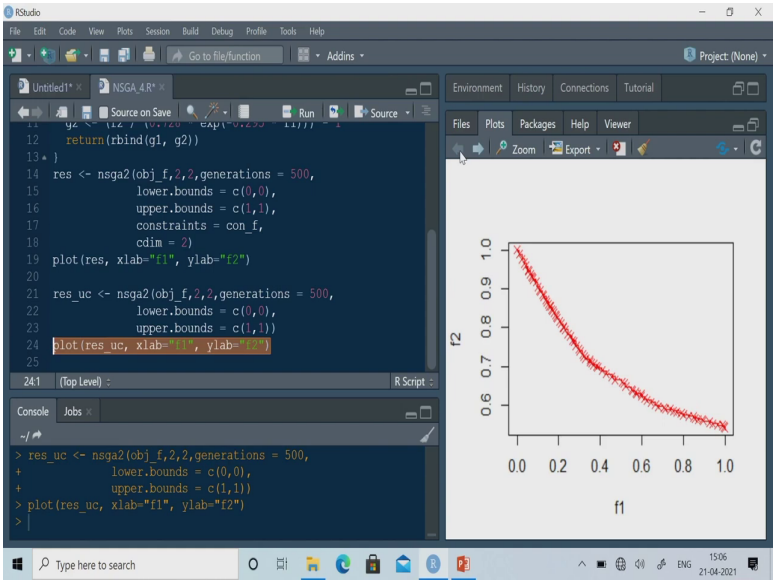
Then I have used nsga2 so here this is the objective function. So, we have 2 variable 2 function generation I have used 500 here lower bound is 0 0 upper bound is 1 1 constrain is your confun here and number of constraints are 2, so I am defining here. So, let me execute this one. So, library is already there so, I will execute objective function, then constrain function, then let me run nsga2 ok and then plot the results ok so I am getting.

(Refer Slide Time: 37:35)



So, this is the constrain solution of this particular problem. So, constrain pareto optimal font of this problem I can also obtain the unconstrained one. So, let me check what is the unconstrained one here.

(Refer Slide Time: 37:49)



So, this is I am writing unconstrained so you see ok. So, here I will not define yeah. So, this is not required and I would like to plot the unconstrained one. So, this is uc ok. So, let me execute this one so, this is the unconstrained one and then let me plot it ok this one yeah. So, this is the unconstrained pareto optimal font of this problem. So, this is the unconstrained one and this is your constrained one.

So, I hope this is clear to you. So, now, you can apply nsga2 algorithm for solving multi objective constrain or unconstrained optimization problem. So, you can find out the pareto optimal font of multi objective problem. So, as I said so, I can have in any number of constrain any number of objective function. So, I can use, but it will be difficult to plot after 3 so, after 3 objective function. So, let me stop here.

Thank you very much.