

**Universite Jean Monnet**

**Optimization and Operational Research**

---

**Constrained optimization**

---

**Practical Session Report**

**Leila Jamshidian sales**

**M1 Master's in Machine Learning and Data Mining**

**Guided by  
Prof. Amaruy Habrard**



**Objective:**

In Optimization lab session the goal was to learn to formulate (simplified) real-world problems as optimization problems and use the software AMPL to solve them.

**Constrained optimization**

In mathematical optimization, constrained optimization (in some contexts called constraint optimization) is the process of optimizing an objective function with respect to some variables in the presence of constraints on those variables. The objective function is either a cost function or energy function which is to be minimized, or a reward function or utility function, which is to be maximized. Constraints can be either hard constraints which set conditions for the variables that are required to be satisfied, or soft constraints which have some variable values that are penalized in the objective function if, and based on the extent that, the conditions on the variables are not satisfied<sup>1</sup>.

**1 Using AMPL****1.1 The AMPL program**

Below is defined some command line code that is used for this software and our practical session.

**Command :** option solver loqo

Choosing loqo solver

LOQO is a powerful solver for smooth constrained optimization problems, based on interior-point method applied to a sequence of quadratic approximations. For convex problems, LOQO finds a globally optimal solution; otherwise, it iterates from the given starting point to find a locally optimal solution.

**Command:** model problem.mod;

Load the optimization file which was created with variables and the minimization function.

**Command:** solve;

Solver the minimize profit problem based on the loqo solver and displays the optimal solution of the problem.

**Command:** display x1;

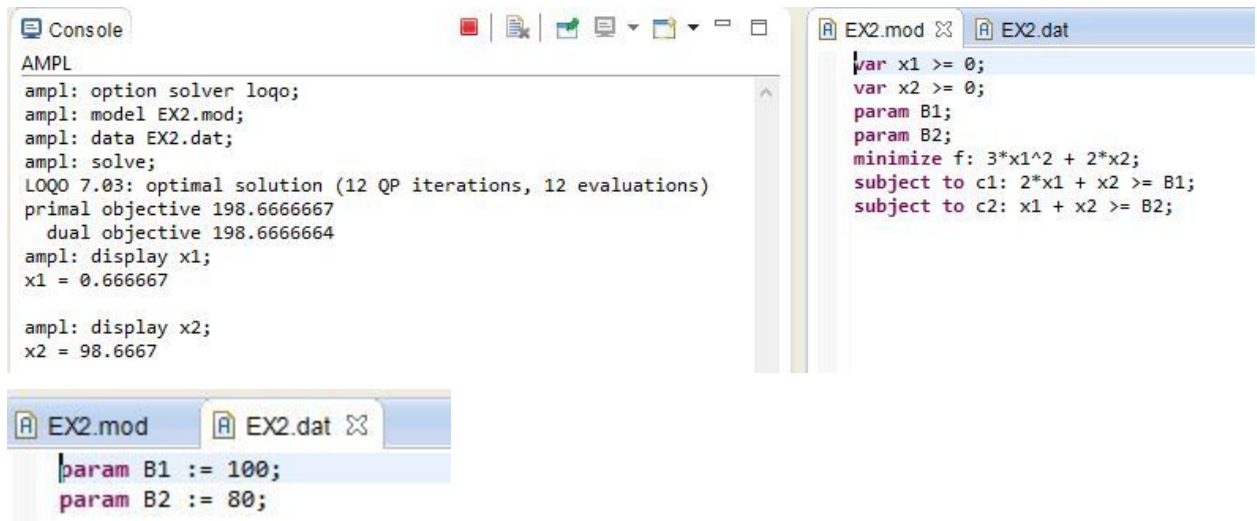
To display the value of x1 which was obtained by solving the minimum profit using the Loqo solver.

---

<sup>1</sup> [https://en.wikipedia.org/wiki/Constrained\\_optimization](https://en.wikipedia.org/wiki/Constrained_optimization)

## 1.2 Syntax for model and data files

Based on exercise of course slides (some examples of syntax), we implemented one example for using command **model** and **data** and learn how to use them.



The screenshot shows the AMPL software interface. The Console window on the left displays the following output:

```
AMPL
ampl: option solver loqo;
ampl: model EX2.mod;
ampl: data EX2.dat;
ampl: solve;
LOQO 7.03: optimal solution (12 QP iterations, 12 evaluations)
primal objective 198.6666667
dual objective 198.6666664
ampl: display x1;
x1 = 0.666667

ampl: display x2;
x2 = 98.6667
```

There are two file editors open on the right. The 'EX2.mod' editor contains the following code:

```
var x1 >= 0;
var x2 >= 0;
param B1;
param B2;
minimize f: 3*x1^2 + 2*x2;
subject to c1: 2*x1 + x2 >= B1;
subject to c2: x1 + x2 >= B2;
```

The 'EX2.dat' editor contains the following code:

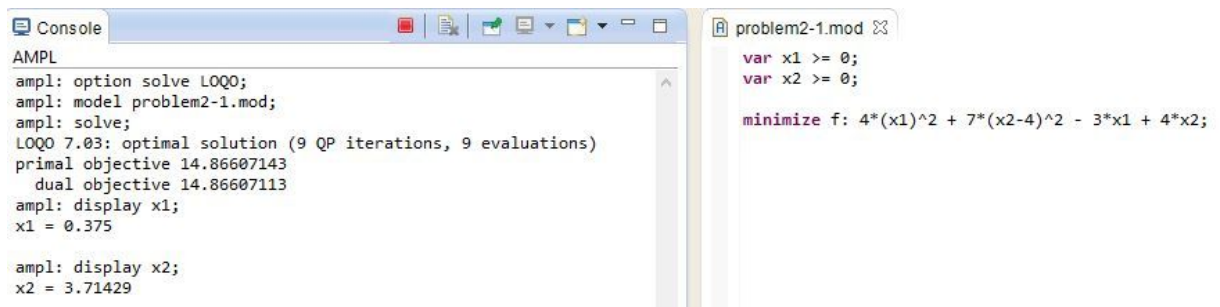
```
param B1 := 100;
param B2 := 80;
```

In this example, on separate database in defined ( EX2.dat) and there is no need to define value for B1 and B2 in model, just we call the command (data EX2.dat) to use values.

## 2 Unconstrained optimization

We start with some simple examples of optimization without any constrained and limitation for the values of variables.

### 2.1 minimize profit: $4x_1x_1 + 7(x_2 - 4)^2 - 3x_1 + 4x_2$ ;



The screenshot shows the AMPL software interface. The Console window on the left displays the following output:

```
AMPL
ampl: option solve LOQO;
ampl: model problem2-1.mod;
ampl: solve;
LOQO 7.03: optimal solution (9 QP iterations, 9 evaluations)
primal objective 14.86607143
dual objective 14.86607113
ampl: display x1;
x1 = 0.375

ampl: display x2;
x2 = 3.71429
```

The 'problem2-1.mod' editor on the right contains the following code:

```
var x1 >= 0;
var x2 >= 0;

minimize f: 4*(x1)^2 + 7*(x2-4)^2 - 3*x1 + 4*x2;
```

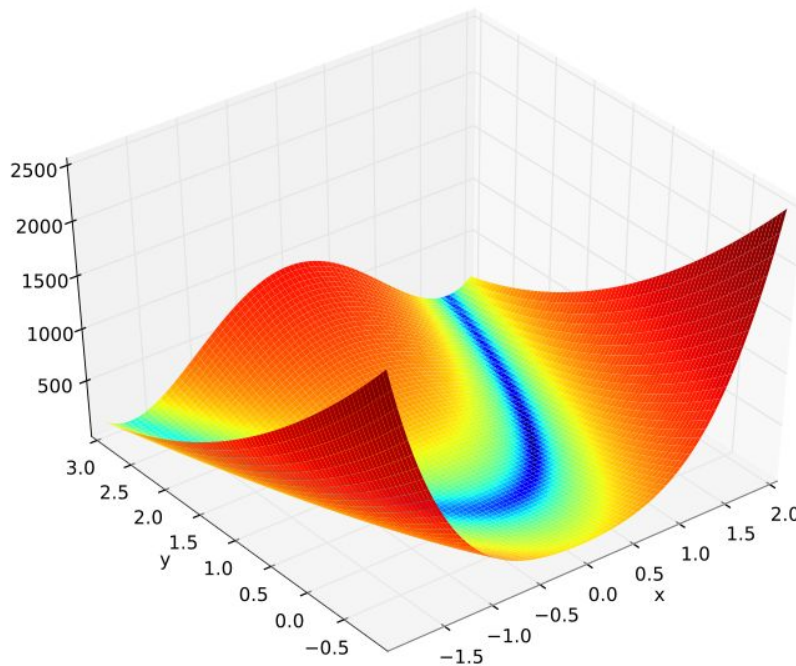
The minimum optimal value of the function is to be 14.86607 and the result is correct as we solved it by hand.

**2.2** minimize profit:  $(1-x_1)^2 + 100*(x_2-x_1^2)^2$ ;

This is the **ROSENBROCK** function that is a non-convex function where

$$f(x, y) = (a - x)^2 + b(y - x^2)^2$$

Here  $a = 1$  ,  $b = 100$  and the minimum value of zero is at  $(1, 1)$  .



At the first we will define the function

```
problem2-1.mod  problem2-2.mod ✕
var x1>=0;
var x2>=0;|
minimize f: ((1-x1)^2)+ (100*(x2-(x1^2))^2)
```

and first answer based on initial state defined by solver is as below:

```
Console
AMPL
ampl: reset;
ampl: model problem2-2.mod;
ampl: solve;
LOQO 7.03: optimal solution (11 iterations, 11 evaluations)
primal objective 3.266397081e-15
dual objective -2.509932593e-09
ampl: display x1;
x1 = 1

ampl: display x2;
x2 = 1
```

When changing the initial value for  $x_1=2$  and  $x_2=2$  , the answer is as below:

```
Console
AMPL
ampl: reset;
ampl: option solve LOQO;
ampl: model problem2-2.mod;
ampl: let x1 := 2;
ampl: let x2 := 2;
ampl: solve;
LOQO 7.03: optimal solution (29 iterations, 36 evaluations)
primal objective 8.53709544e-16
dual objective 8.537091411e-16
ampl: display x1;
x1 = 1

ampl: display x2;
x2 = 1
```

We tried for different initial values as defined in table below. Optimal numbers for  $x_1$  and  $x_2$  after minimizing function and finding optimal solution are 1, as you can see in the above graph, the minimum value for function  $F$  is achieved by  $x_1=x_2=1$ , so for every different initial value we get always the optimal minimum.

Initial $x_1$	Initial $x_2$	#iterations	Optimal $x_1$	Optimal $x_2$
By solver(0)	By solve (0)	11	1	1
2	2	29	1	1
0.5	0.5	17	1	1

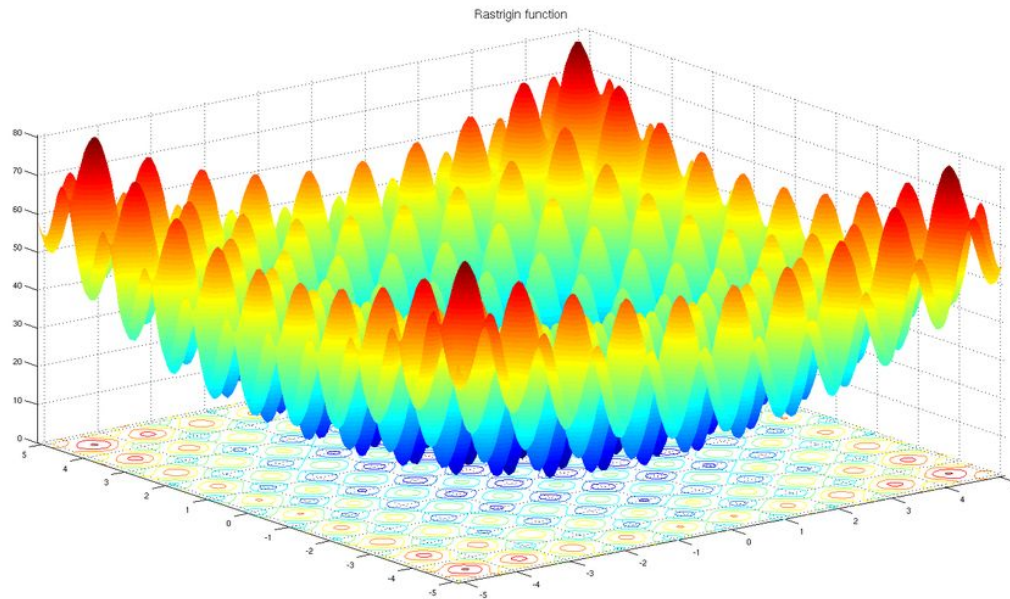
**2.3 minimize profit:**  $(10 + x_1^2 + x_2^2) - 10*(\cos(2*\pi*x_1) + \cos(2*\pi*x_2))$ ;

the **Rastrigin function** is a non-convex function used as a performance test problem for optimization algorithms. As you can see in the graph, finding the minimum of this function is difficult problem due to its large search space and its large number of local minima.

Function is like below:

$$f(\mathbf{x}) = An + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)]$$

Where here A=10 and n=1.



Here at the first we defined our function.

```
problem2-3.mod
var x1 >= 0;
var x2 >= 0;
param pi := 3.14159;

minimize f: (10 + x1^2 + x2^2) - 10*(cos(2*pi*x1) + cos(2*pi*x2));
```

In this step we run the solver without choosing initial value for x1 and x2.

```
Console
AMPL
ampl: reset;
ampl: option solve LOQO;
ampl: model problem2-3.mod;
ampl: solve;
LOQO 7.03: optimal solution (1 iterations, 1 evaluations)
primal objective -10
dual objective -10
ampl: display x1;
x1 = 0

ampl: display x2;
x2 = 0
```

In the second step we define different values for x1 and x2.

Initial x1	Initial x2	#iterations	Optimal x1	Optimal x2
By solver(0)	By solve (0)	1	0	0
-5	-5	500	-2.5331	-2.5331
-4	4	17	1.98991	3.97979
3	4	8	2.98486	3.97979

The image shows three screenshots of the AMPL console window, each displaying the results of solving a problem with different initial values for x1 and x2.

**Top Left Screenshot:** Initial x1 = -4, Initial x2 = 4. The console shows the optimal solution (17 iterations, 21 evaluations) with primal objective 9.899108429 and dual objective 9.899108411. The optimal values are x1 = 1.98991 and x2 = 3.97979.

**Top Right Screenshot:** Initial x1 = -5, Initial x2 = -5. The console shows the iteration limit (500 iterations, 501 evaluations) with primal objective 42.4022356 and dual objective 42.4022356. The optimal values are x1 = -2.5331 and x2 = -2.5331.

**Bottom Screenshot:** Initial x1 = 3, Initial x2 = 4. The console shows the optimal solution (8 iterations, 8 evaluations) with primal objective 14.87388684 and dual objective 14.8738868. The optimal values are x1 = 2.98486 and x2 = 3.97979.

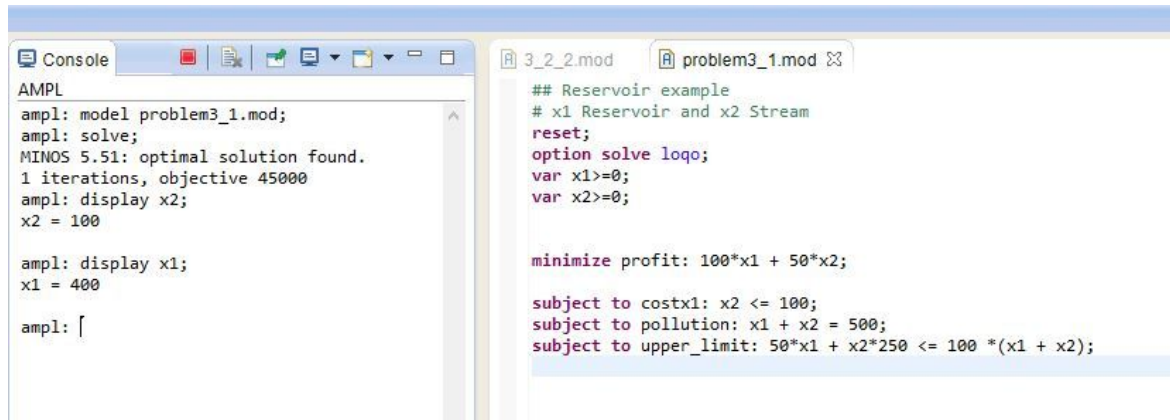
As it is shown and based on graph too, for different initial value values of x1 and x2 are different and here there is no optimal minimum, this function has local minimums.

### 3 Modeling constrained problems

#### 3.1 Water Resources

This city needs 500000 liters water per day and can be achieved by either X1 - Reservoir, X2 - Stream(  $X1+X2 \leq 500$ ). Next condition says that no more than 100,000 liters per day can be drawn from the stream so  $X2 \leq 100$ . Other condition is concentration of pollutants in the water served to the city cannot exceed 100 ppm so based on table upper\_limit is defined to satisfy this constraint.

Minimization of function :  $100 \cdot X_1 + 50 \cdot X_2$  to figure out the quantity of water to satisfy the need of city based on constraints with respect to cost, pollution and upper\_limit.



The screenshot shows the AMPL software interface. On the left is the 'Console' window, and on the right is the 'problem3\_1.mod' file editor.

```

Console:
AMPL
ampl: model problem3_1.mod;
ampl: solve;
MINOS 5.51: optimal solution found.
1 iterations, objective 45000
ampl: display x2;
x2 = 100

ampl: display x1;
x1 = 400

ampl: [

problem3_1.mod:
## Reservoir example
# x1 Reservoir and x2 Stream
reset;
option solve loqo;
var x1>=0;
var x2>=0;

minimize profit: 100*x1 + 50*x2;

subject to costx1: x2 <= 100;
subject to pollution: x1 + x2 = 500;
subject to upper_limit: 50*x1 + x2*250 <= 100 *(x1 + x2);
  
```

The result indicates the value of  $x_1$ : 400 and  $x_2$ :100 to attain the minimum profit of the constraint satisfaction problem.

### 3.2 Good-smelling perfume design

The objective of the problem is to develop the new perfume from existing four blends. The minimizing problem to find new one by blending 4 oils based on the constraints.

$$\text{minimize profit: } 55 \cdot x_1 + 65 \cdot x_2 + 35 \cdot x_3 + 85 \cdot x_4;$$

Here we are looking for the least costly way of mixing the 4 blends of essential oils to produce a new perfume, where  $x_1$  is blend 1 and  $x_2$  is blend 2 and  $x_3$  is blend 3 and  $x_4$  is blend 4.

List of constraints are given below.

1-the percentage of blend 2 in the perfume must be at least 5% and cannot exceed 20%,  
 $x_2 \geq 0.05; x_2 \leq 0.2;$

2-the percentage of blend 3 has to be at least 30%,  
 $x_3 \geq 0.3;$

3-the percentage of blend 1 has to be between 10% and 25%,  
 $x_1 \geq 0.1; x_1 \leq 0.25;$

4-the nal percentage of bergamot orange content in the perfume must be at most 50%,  
 $0.35 \cdot x_1 + 0.6 \cdot x_2 + 0.35 \cdot x_3 + 0.4 \cdot x_4 \leq 0.5;$

5-the nal percentage of thymus content has to be between 8% to 13%,  
 $0.15 \cdot x_1 + 0.05 \cdot x_2 + 0.2 \cdot x_3 + 0.1 \cdot x_4 \leq 0.13;$



$$0.15*x1 + 0.05*x2 + 0.2*x3 + 0.1*x4 \geq 0.08;$$

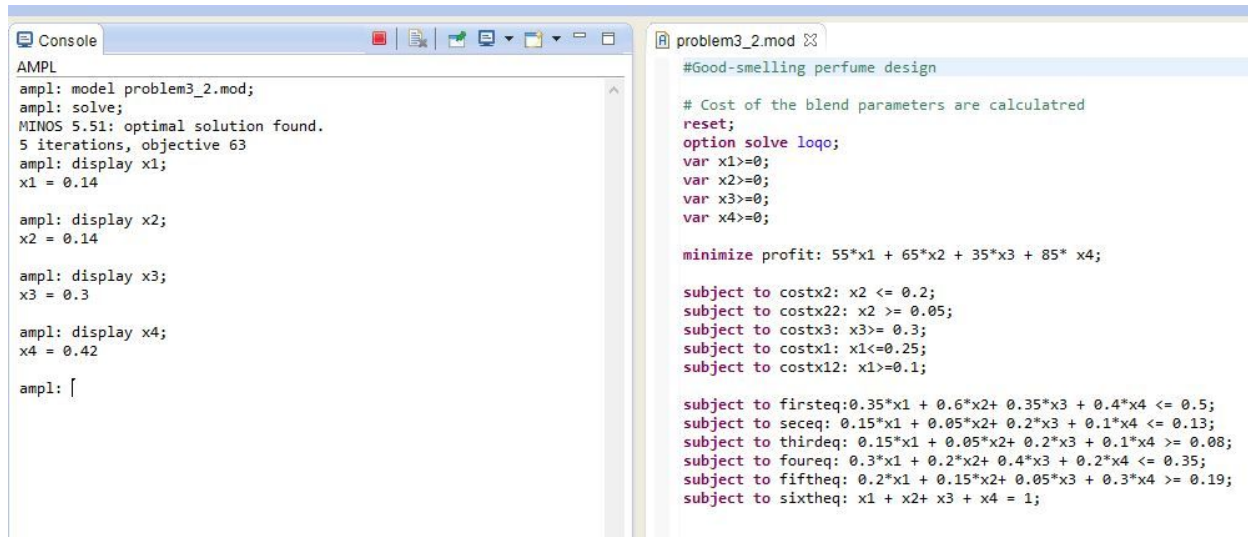
6-the nal percentage of rose content must be at most 35%,

$$0.3*x1 + 0.2*x2 + 0.4*x3 + 0.2*x4 \leq 0.35;$$

7-the percentage of lily of the valley content has to be at least 19%.

$$0.2*x1 + 0.15*x2 + 0.05*x3 + 0.3*x4 \geq 0.19;$$

8-  $x1 + x2 + x3 + x4 = 1$



The screenshot shows the AMPL solver interface. The left pane, titled 'Console', displays the following output:

```

AMPL
ampl: model problem3_2.mod;
ampl: solve;
MINOS 5.51: optimal solution found.
5 iterations, objective 63
ampl: display x1;
x1 = 0.14

ampl: display x2;
x2 = 0.14

ampl: display x3;
x3 = 0.3

ampl: display x4;
x4 = 0.42

ampl: [

```

The right pane, titled 'problem3\_2.mod', shows the model code:

```

#Good-smelling perfume design

# Cost of the blend parameters are calculated
reset;
option solve loqo;
var x1>=0;
var x2>=0;
var x3>=0;
var x4>=0;

minimize profit: 55*x1 + 65*x2 + 35*x3 + 85* x4;

subject to costx2: x2 <= 0.2;
subject to costx22: x2 >= 0.05;
subject to costx3: x3>= 0.3;
subject to costx1: x1<=0.25;
subject to costx12: x1>=0.1;

subject to firsteq:0.35*x1 + 0.6*x2+ 0.35*x3 + 0.4*x4 <= 0.5;
subject to seceq: 0.15*x1 + 0.05*x2+ 0.2*x3 + 0.1*x4 <= 0.13;
subject to thirdeq: 0.15*x1 + 0.05*x2+ 0.2*x3 + 0.1*x4 >= 0.08;
subject to foureq: 0.3*x1 + 0.2*x2+ 0.4*x3 + 0.2*x4 <= 0.35;
subject to fiftreq: 0.2*x1 + 0.15*x2+ 0.05*x3 + 0.3*x4 >= 0.19;
subject to sixtheq: x1 + x2+ x3 + x4 = 1;

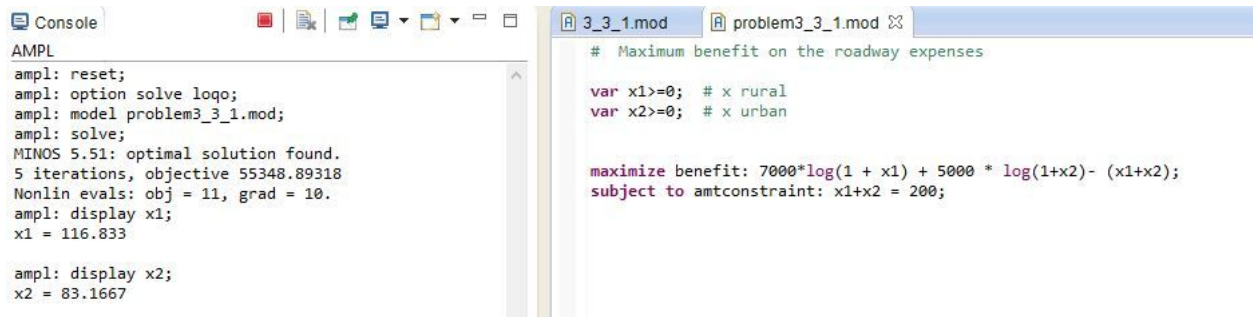
```

After solving this optimization problem in the AMPL, the result are as shown in the figure, where  $x1 = 0.14$ ,  $x2 = 0.14$ ,  $x3 = 0.3$ ,  $x4 = 0.42$ . This means to prepare the product with 14%, 14%, 30% and 42% in order for the four blends. The result does not make sense cause we are using expensive blends more than cheaper one and with respect to that we are minimizing. Based on this solution  $x4$  has 85 cost and the amount of this blend is more than others that they are cheaper and cheapest blends- $x3$ - is used fewer that others.

### 3.3 Road Expenses

#### 3.3.1 Rural / Urban cases:

In order to develop the roadway of France, to identify the maximum benefit to by spending on rural or urban project based on the constraints.



The screenshot shows an AMPL IDE with two windows. The left window is the 'Console' showing the execution of an AMPL model. The right window is the 'problem3\_3\_1.mod' file, which contains the mathematical model.

```
Console
AMPL
ampl: reset;
ampl: option solve loqo;
ampl: model problem3_3_1.mod;
ampl: solve;
MINOS 5.51: optimal solution found.
5 iterations, objective 55348.89318
Nonlin evals: obj = 11, grad = 10.
ampl: display x1;
x1 = 116.833

ampl: display x2;
x2 = 83.1667

3_3_1.mod
# Maximum benefit on the roadway expenses

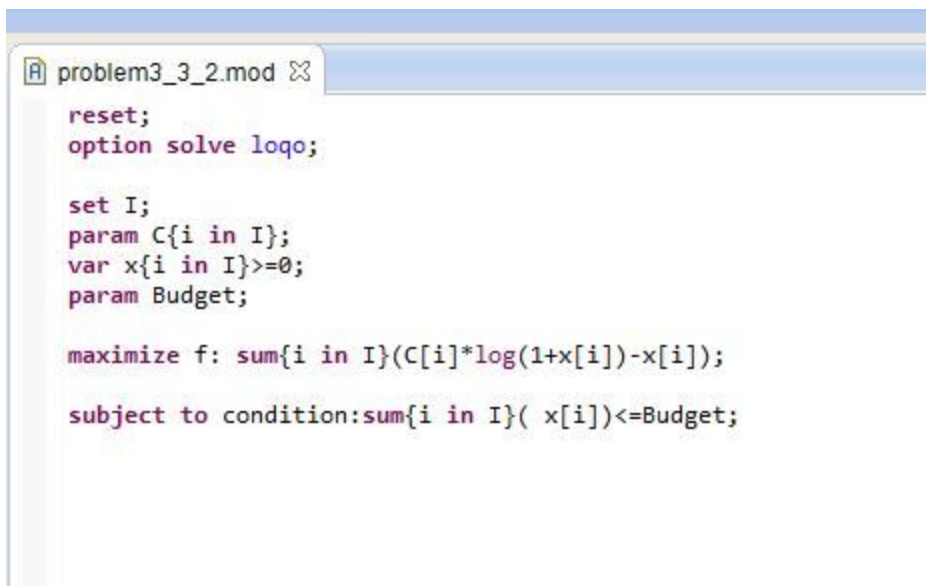
var x1>=0; # x rural
var x2>=0; # x urban

maximize benefit: 7000*log(1 + x1) + 5000 * log(1+x2)- (x1+x2);
subject to amtconstraint: x1+x2 = 200;
```

The benefit of spending money on rural road is more than urban road and in the optimal solution  $x_1=116.833$  and  $x_2=83.1667$  as it makes sense cause value of  $x$  rural is bigger than  $x$  urban and optimal benefit is 55348.89318.

### 3.3.2.1 General case:

In order to generalize more about the previous problem, more generic method to compute the efficient way of investment is figured out. This is the general case using an AMPL model where no numbers appear.



The screenshot shows an AMPL IDE with a single window titled 'problem3\_3\_2.mod'. The window contains a general AMPL model without specific numerical values.

```
problem3_3_2.mod
reset;
option solve loqo;

set I;
param C{i in I};
var x{i in I}>=0;
param Budget;

maximize f: sum{i in I}(C[i]*log(1+x[i])-x[i]);

subject to condition:sum{i in I}( x[i])<=Budget;
```

### 3.3.2.2

In this part with defining the data file, we assign the value of the previous section and solve the problem. This result is exactly same as step 3.3.1 .

```

Console
AMPL
ampl: model problem3_3_2.mod;
ampl: data problem3_3_2.dat;
ampl: solve;
MINOS 5.51: optimal solution found.
5 iterations, objective 55348.89318
Nonlin evals: obj = 12, grad = 11.
ampl: display x;
x [*] :=
rural 116.833
urban 83.1666
;

problem3_3_2.mod  problem3_3_2.dat
set I:=rural urban ;
param Budget:= 200;
param C := rural 7000 urban 5000 ;

```

### 3.3.2.3

In this step we are adding the third road category to the problem (suburban) with  $C(\text{suburban}) = 7000$  and solve the problem.

```

Console
AMPL
ampl: model problem3_3_3.mod;
ampl: data problem3_3_3.dat;
ampl: solve;
MINOS 5.51: optimal solution found.
9 iterations, objective 80096.50465
Nonlin evals: obj = 20, grad = 19.
ampl: display x;
x [*] :=
rural 73.7895
suburban 73.7895
urban 52.4211
;

problem3_3_3.mod  problem3_3_3.dat
set I:=rural urban suburban;
param Budget:= 200;
param C := rural 7000 urban 5000 suburban 7000;

```

## 3.4 Design your own optimization problem

### 3.4.1 Problem Definition:

Now this is the end of semester and time for finals. The goal is to increase average as we can validate semester. This semester consists of four courses (machine learning (ECTS 6), data mining (ECTS 4), algorithms (ECTS 5), Artificial intelligent (ECTS 3)). Based on experience we know if we study one hour machine learning, we increase final grade at least 0.25, one hour data mining increases final grade 0.3, one hour artificial intelligent increases final grade 0.45, and one hour algorithms increases final grade 0.27. overall time that we have is just 200 hours. How we can share our time between courses to get the maximum average based on the time that we have.

(\*\*the final grades for courses are at most 20\*\*)

(\*\* the minimum study for each course is 10 hours\*\*)

### Constrained Optimization problem:

The constrained optimization problem is to find the maximum average thus the person could achieve based on the time for study.

(\*\* the minimum study for each course is 10 hours\*\*)

**First constraint:**  $\text{var } x1 \geq 10; \text{var } x2 \geq 10; \text{var } x3 \geq 10; \text{var } x4 \geq 10;$

(\*\*the final grades for courses are at most 20\*\*)

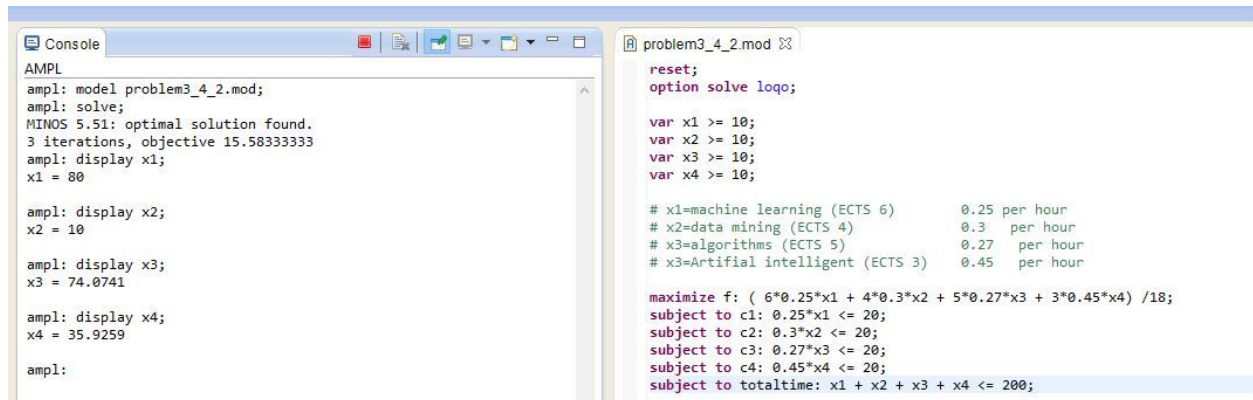
**Second constraints:**

$c1: 0.25 \cdot x1 \leq 20; \quad c2: 0.3 \cdot x2 \leq 20; \quad c3: 0.27 \cdot x3 \leq 20; \quad c4: 0.45 \cdot x4 \leq 20;$

$\text{totaltime: } x1 + x2 + x3 + x4 \leq 200$

Now goal is to maximize :  $f: (6 \cdot 0.25 \cdot x1 + 4 \cdot 0.3 \cdot x2 + 5 \cdot 0.27 \cdot x3 + 3 \cdot 0.45 \cdot x4) / 18;$

The result shows that the maximum average is 15.58 with 200 hours study and you have study  $x1=80$  and  $x2=10$  and  $x3=74$  and  $x4=35$  hours.



The screenshot shows a software interface with two panes. The left pane, titled 'Console', displays the output of an AMPL solver. It shows the command 'ampl: model problem3\_4\_2.mod;' followed by 'ampl: solve;' and the message 'MINOS 5.51: optimal solution found. 3 iterations, objective 15.5833333'. It then shows the results of 'ampl: display x1;' (x1 = 80), 'ampl: display x2;' (x2 = 10), 'ampl: display x3;' (x3 = 74.0741), and 'ampl: display x4;' (x4 = 35.9259). The right pane, titled 'problem3\_4\_2.mod', shows the AMPL model code. It starts with 'reset;' and 'option solve loq;'. It then defines variables x1, x2, x3, and x4, all with lower bounds of 10. It includes comments for each variable: x1 is machine learning (ECTS 6) at 0.25 per hour, x2 is data mining (ECTS 4) at 0.3 per hour, x3 is algorithms (ECTS 5) at 0.27 per hour, and x4 is Artificial intelligent (ECTS 3) at 0.45 per hour. The objective function is 'maximize f: ( 6\*0.25\*x1 + 4\*0.3\*x2 + 5\*0.27\*x3 + 3\*0.45\*x4) /18;'. It then lists constraints: 'subject to c1: 0.25\*x1 <= 20;', 'subject to c2: 0.3\*x2 <= 20;', 'subject to c3: 0.27\*x3 <= 20;', 'subject to c4: 0.45\*x4 <= 20;', and 'subject to totaltime: x1 + x2 + x3 + x4 <= 200;'.

(\*\* the minimum study for each course is 10 hours\*\*)

**First constraint:**  $\text{var } x1 \geq 10; \text{var } x2 \geq 10; \text{var } x3 \geq 10; \text{var } x4 \geq 10;$

(\*\*the final grades for courses are at most 20\*\*)

**Second constraints:**

$c1: 0.25 \cdot x1 \leq 20; \quad c2: 0.3 \cdot x2 \leq 20; \quad c3: 0.27 \cdot x3 \leq 20; \quad c4: 0.45 \cdot x4 \leq 20;$

$\text{totaltime: } x1 + x2 + x3 + x4 \leq 200$

Now goal is to maximize :  $f: (6 \cdot 0.25 \cdot x1 + 4 \cdot 0.3 \cdot x2 + 5 \cdot 0.27 \cdot x3 + 3 \cdot 0.45 \cdot x4) / 18;$

The result shows that the maximum average is 15.58 with 200 hours study and you have study  $x1=80$  and  $x2=10$  and  $x3=74$  and  $x4=35$  hours.

## 4 Data Analysis

### Machine Learning:

In Machine Learning, Linear Regression is the method to model the relationship between a scalar dependent variable  $y$  and one or more explanatory variables. The linear regression is composed The regression of the model based on different norms has a varied error rate to its prediction accuracy.

The computation of equation of line over the regression model to compute the next element in a sequence:

Function	L1	L2
a	0.132075	0.288549
b	4.86792	26.3217
Number of elements	20	20
Number of iterations	11	9
Objective Function	547.0566042	102.510

The screenshot displays the AMPL (Algebraic Modeling Language) interface. On the left, the 'Console' window shows the execution results for two models: L1\_linear\_regression.mod and L2\_linear\_regression.mod. The L1 model results are: primal objective 547.0566042, dual objective 547.056601, a = 0.132075, and b = 4.86792. The L2 model results are: primal objective 102.5105342, dual objective 102.5105342, a = 0.288549, and b = 26.3217. On the right, the 'L2\_linear\_re...' window shows the AMPL code for the L2 regression model. The code includes comments for 'Machine Learning Basic Model L2', a 'reset;' command, parameter declarations for 'num\_of\_elements', 'a', and 'b', variable declarations for 'x' and 'y', a computation for 'z', and a minimization objective function for 'l2\_regression'.

```

AMPL
ampl: reset;
ampl: model D:\Vinoth\Study_materials\Optimization_lab\L1_linear_regression.mod;
ampl: data D:\Vinoth\Study_materials\Optimization_lab\Linear_regression_dataset.dat;
LOGO 7.03: optimal solution (11 iterations, 11 evaluations)
primal objective 547.0566042
dual objective 547.056601
a = 0.132075
b = 4.86792

ampl: model D:\Vinoth\Study_materials\Optimization_lab\L2_linear_regression.mod;
ampl: data D:\Vinoth\Study_materials\Optimization_lab\Linear_regression_dataset.dat;
LOGO 7.03: optimal solution (9 iterations, 9 evaluations)
primal objective 102.5105342
dual objective 102.5105342
a = 0.288549
b = 26.3217
ampl:

# Machine Learning Basic Model L2
## The Analysis of regression model

reset; # Reset to remove all previous values

param num_of_elements; # Defining the number of elements parameters

# Declaration of variables a and b where the constants
var a;
var b;

# Initializing the number of parameters with respect to x and z
param x{i in 1..num_of_elements};
param y{i in 1..num_of_elements};

# Computation of z = ax+b
var z{i in 1..num_of_elements}=y[i]-(a*x[i]+b);

# Minimizing the regression
minimize l2_regression : sqrt(sum{i in 1..num_of_elements} ((z[i])^2));

```

The expression of L1 and L2 model where the L2 regression has shown the better efficiency over the set of models with standard stable solution.

## SVM :

As we saw in the course one of the supervised learning algorithm is SVM which has binary classification. Here is defined the general program of SVM dual problem in AMPL.

Optimization problem (linear kernel)

$$\sum_{i=1}^l a_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l a_i a_j y_i y_j (x_i, x_j)$$

$$\sum_{i=1}^l a_i y_i = 0$$

Optimization problem (RBF kernel)

$$\sum_{i=1}^l a_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l a_i a_j y_i y_j \exp(-\gamma (x_i, x_j)^2)$$

$$\sum_{i=1}^l a_i y_i = 0$$

```

SVM.mod
reset;
option solve lqoq;
# Number of training samples
param l;
# Number of independent samples
param l2;
# Number of variables
param n;
# C parameter of SVM
param C;
# Parameters of RBF kernel
param gamma;
# Label vectors
param y {1..l};
param y2 {1..l2};
# Data (variables) matrices
param x {1..l, 1..n};
param x2 {1..l2, 1..n};
# Dual problem variables a (lambda) and C constraint
var a {1..l} >= 0, <= C;
# Optimization problem (linear kernel)
maximize svm1: sum {i in 1..l} a[i] - 0.5 *
sum {i in 1..l, j in 1..l} a[i] * a[j] * y[i] * y[j] *
(sum {k in 1..n} x[i, k] * x[j, k]);

# Optimization problem (RBF kernel)
maximize svmrad: sum {i in 1..l} a[i] - 0.5 *
sum {i in 1..l, j in 1..l} a[i] * a[j] * y[i] * y[j] *
exp(-gamma * (sum {k in 1..n} (x[i, k] - x[j, k])^2));
# Constraints
s.t. rest1: sum {i in 1..l} a[i] * y[i] = 0;

```

**Conclusion :**

Thus, Optimization practical session with the help of AMPL has given a new exposure to wide variety of real world problems. It strongly helps to map the theoretical concepts in practical problems to solve efficiently. Thus optimization helps to have a very good understanding of operations and optimization methods.