Constrained optimization - Practical Session

Master DSC/MLDM/CPS2

Assignment: Due date 22/03/2020 12:00 (on claroline - Assignments section)
It is recommended to post the assignment as soon as possible

This work will be done during 2 practical sessions - no more than 2 students can be associated to the same work

The goal of this practical is to learn to formulate (simplified) real-world problems as optimization problems in nice form, and use CVXPY to solve them.

Assignment: You must send a report on your work, detailing your methodology, including the results and the commands used. Personal comments are welcome. The work can be done by at most 2 students and must be posted on claroline (use `.zip` or `.tar.gz`).

1 Using CVXPY

CVXPY is a Python-embedded modeling language for convex optimization problems. It automatically transforms the problem into standard form, calls a solver, and unpacks the results.

An example of a simple optimization problem solved using CVXPY in Python is given below.

```
import cvxpy as cp

# Create two scalar optimization variables.
x = cp.Variable()
y = cp.Variable()

# Create two constraints.
constraints = [x + y == 1,
               x - y >= 1]

# Form objective.
obj = cp.Minimize((x - y)**2)

# Form and solve problem.
prob = cp.Problem(obj, constraints)
prob.solve()  # Returns the optimal value.
print("status:", prob.status)
print("optimal value", prob.value)
print("optimal var", x.value, y.value)
```

To define and solve an optimization problem using CVXPY in Python you need to do the following:

1. **Install** and **import** the cvxpy package

2. Declare **variables** involved in the optimization problem. This can be done using *cp.Variable()* command where *Variable()* method can be used with attributes to define a scalar *cp.Variable()* or vector

variable *cp.Variable(n)* where n is the dimensionality of the variable. For more details on variable definition, follow the link `https://www.cvxpy.org/api_reference/cvxpy.expressions.html` and go to Variable section.

3. Declare the **constraints** involved in the optimization problem. You can use $==, <=, and >=$ to construct constraints in CVXPY but strict inequalities $<$ and $>$ are forbidden. Equality and inequality constraints are element-wise, whether they involve scalars, vectors, or matrices. As shown above, the constraints set is defined as a list. For more details on constraints definition, follow the link `https://www.cvxpy.org/tutorial/intro/index.html` and go to Constraint section.

4. Declare the **objective function** using *cvxpy.Minimize(expression)* and *cvxpy.Maximize(expression)*. The expression should be defined using standard arithmetic operations and built-in functions of CVXPY for logarithms (*cvxpy.log()*) and trigonometric expressions.

5. Define the **optimization problem** with the objective function and constraints defined previously using the *cp.Problem()* method.

6. Solve the problem using *solve()* method.

2 Unconstrained optimization

We start with 3 unconstrained optimization problems that are easy to formulate to learn the basic features of CVXPY.

2.1 Problem 1

Consider the following problem:

$$\min_{x_1,x_2\in\mathbb{R}} \quad (x_1-4)^2+7(x_2-4)^2+4x_2$$

- Formulate the problem in CVXPY and solve it. Check that you get the correct solution by solving on your own.

2.2 Problem 2

$$\min_{x_1,x_2,x_3\in\mathbb{R}} \quad 2x_1^3+2(3x_2-x_3)^2-4x_3+2$$

- Is this problem convex? Why?
- Find the analytical solution to the problem (if needed do not hesitate to add some constraints).
- Formulate the problem in CVXPY and solve it. Check that you get the correct solution by solving the problem on your own.

2.3 Problem 3

Consider the following constrained problem:

$$\min_{x_1,x_2\in\mathbb{R}} \quad (x_1-2)^2+3x_2, \text{ s.t. } -x_1-x_2+4\geq 0, \quad x_1,x_2\leq 0.$$

- Reformulate this problem using the log-barrier function in CVXPY. What option should you pass to the *solve()* command for this case?
- Try different values of the coefficient associated to the log-barrier function by defining a parameter variable with the *cvxpy.Parameter()* command. What do you observe?

3 Modeling constrained problems

We now focus on modeling some kind of "real" problems.

3.1 Water resources

A city needs 500,000 liters of water per day, which can be drawn from either a reservoir or a stream. Characteristics of these two sources are found in Table 1. No more than 100,000 liters per day can be drawn from the stream, and the concentration of pollutants in the water served to the city cannot exceed 100 ppm [1]. The problem is to find how much water the city should draw from each source (subject to the constraints) so as to minimize the cost.

	Reservoir	**Stream**
Cost (€ per 1,000L)	100	50
Upper limit (× 1,000L)	∞	100
Pollution (ppm)	50	250

Table 1: Reservoir and stream information

3.2 Good-smelling perfume design

A perfume company located in the south of France is trying to develop new perfumes based on mixes of 4 blends of essential oils developed in their lab. Each of these 4 blends embeds some essences of real flowers (rose, bergamot orange, lily of the valley, thymus). The table below describes the composition of each blend and the cost to produce each of them. The company wants to develop its own trademark and imposes some contraints on the possible mixes to design a new perfume:

	blend 1	blend 2	blend 3	blend 4
rose	30	20	40	20
bergamot orange	35	60	35	40
lily of the valley	20	15	5	30
thymus	15	5	20	10
Cost (€/liter)	55	65	35	85

- the percentage of blend 2 in the perfume must be at least 5% and cannot exceed 20%,
- the percentage of blend 3 has to be at least 30%,
- the percentage of blend 1 has to be between 10% and 25%,
- the final percentage of bergamot orange content in the perfume must be at most 50%,
- the final percentage of thymus content has to be between 8% to 13%,
- the final percentage of rose content must be at most 35%,
- finally, the percentage of lily of the valley content has to be at least 19%.

We are looking for the least costly way of mixing the 4 blends of essential oils to produce a new perfume subject to the constraints given above.

Questions:

1. Model this problem as an constrained optimization problem.
2. Formulate it in CVXPY and solve it. Check that the optimal solution makes sense. What is the optimal cost?

[1] The proportion of pollutants is measured by the quantity of pollutants coming from the reservoir and from the stream divided by the quantity in stream and reservoir.

3.3 Roadway expenses

France has 200M€ to spend on roadway improvements this coming year, and the government has to decide how much to spend on rural projects, and how much to spend on urban projects. Let x_{rural} and x_{urban} represent the amount of money spent on these two categories, in millions of euros. The benefit from spending x_{rural} on rural projects is:

$$B_{rural} = 7000 \log(1 + x_{rural}),$$

and the benefit from spending x_{urban} on urban projects is

$$B_{urban} = 5000 \log(1 + x_{urban}).$$

We want to maximize the net benefit to the state, that is, $B_{rural} + B_{urban} - x_{rural} - x_{urban}$.

Questions:

1. Model this problem as a constrained optimization problem.
2. Formulate it in CVXPY and solve it. Check that the optimal solution makes sense. What is the optimal benefit?

3.4 Design you own optimization problem

Now it's your turn! Propose a problem that could be realistic. It can be related to your personal interests, hobbies (or not). To do:

1. You must clearly define the problem to solve and the data associated with it.
2. Write the optimization problem that can be used to find the solution.
3. Provide the Python code associated with it. If relevant, do not hesitate to precise how general and smart your implementation is.

4 Data Analysis

Choose a method you saw in the context of the data analysis or the machine learning classes that can be expressed as an optimization problem.

1. Propose an implementation of this method with Python and CVXPY.
2. Propose several variants of this problem by using different norms, constraints, formulations when appropriate. The diversity of the formulations will be taken into account.