In this assignment, you will solve the online allocation problem that you have seen in class by using duality. The problem is the following:

- There are m resources, x_1 , ..., x_m with fixed capacities b_1, ..., b_m
- Demand requests arrive sequentially, noted j = 1...N, with requirements a_i, i=1...m and profit p
- We want to make accept/reject decisions without knowing future demands
- We have the historical data of N demands in the file "demand.csv":

$$((a_{ij})\ i=1...m,\ p_j)\ \text{for}\ j=1...N$$

Question 1:

In this question, you will formulate the problem as a linear program. To do so, you will transform the LP to standard form:

$$\max\ c^T x$$

$$Ax <= b$$

$$x >= 0$$

while c^T represents the profit of accepting each demand, A represents the number of required resources, and b represents the upper bound limit for each resources.

You will take m=4, N=100, b_0=125, b_1=130, b_2=118, b_3=137 and use the data in the file *"data_online_allocation.csv"* which simulates the demand for N=100. Return A as a matrix, b as a list, and c as a list.

Note: When you construct the matrix A (lists within list), the first 4 rows will correspond to the capacity constraints and the following 100 rows will correspond to the upper bound of x_i for i=1...100

Question 2:

In this question, you will use the standard form that you have set up in the previous question. Write the dual and solve it using Gurobi. As a reminder, the dual is:

$$\min b^T y$$

$$A^T y >= c$$

$$y >= 0$$

Return the optimal objective value, and the optimal values of the variables in one list.

Note: The list of the optimal values of the variables should contain the dual of the capacity constraints in the first 4 position

Question 3:

Use the shadow prices you fund in the previous questions to create a static policy to make the accept/reject decisions. Apply that policy to the products in the file "new_data.csv" and return the list of the products you reject (this list is called “reject_list”).

The list should be of the form: [2,7,38...] (i.e. it should contain the index of the product, which goes from 1 to 100)

(Hint) Accept if:

$$P_j - sum(lambda_i * a_{ij} , i=1..m) >= 0$$

where $lambda_i$ are the values you calculated when solving the dual in question 2.

Online Allocation (External resource)...