Project proposal

Team member: Zeren Li, Bomin Zhang

Objective: We want to design a **dietary planning** system that output a couple of dishes to make subject to the constraint of time, cost, nutritional requirements.

Motivation: often time, we have trouble to think about what exactly we should eat for today's lunch or dinner. Such a dietary planning system can greatly assist us in dealing such a problem.

Detailed problem: specifically, we have recipe table, nutrition table, cost table as below:

	x1	x2	…	xi	…	xn
R1	Q11	Q12				Q1n
R2						
…						
Rk						
…						
Rd	Qd1					Qdn

Preference	time
p1	t1
pd	td

Recipe table, denoted as Matrix Q. where {x1,...,xn} is set of raw ingredient, {R1,...,Rd} is set of recipe. Each recipe has its own time to make, the preference is rated from 1-5 and should be different according to one's preference.

	x1	x2	…	xi	…	xn
v1	M11					M1n
v2						
…						
vj						
…						
Vm						Mmn

	Cost
x1	c1
x1	c2
…	…
xi	ci
…	…
xn	cn

Cost table. each ingredient has its cost.

Nutrition table. {V1,...,Vm} is set of nutrition to consider.

Input: preference, cost I, time T, nutrition table, cost table.

Output: a vector α, denotes the quantity of every recipe to make.

Optimization problem

obj:

$$\text{Max} \sum_{k}^{d} \alpha k \cdot pk$$

where $\alpha=[\alpha 1, \alpha 2, \ldots, \alpha k, \ldots, \alpha n]^T$ is a vector denotes the quantity of the recipe to make. So, we want to maximize total preferences.

subject to:

Cost:

$$\sum_{k}^{d} \alpha k \cdot Ck <= I$$

, where

$$Ck = \sum_{i}^{n} Qki \cdot ci$$

Explanation: let capital letter Ck denotes cost of recipe k, and ci denotes cost of ingredient i. So, each recipe costs Ck and total cost for all recipe has to be less than letter I(investment).

Nutrition:

$$\sum_{k}^{d} \sum_{i}^{n} Mji \cdot Qik \cdot \alpha k >= wj$$

above constraint equivalent to

$$M \cdot Q^T \cdot \alpha >= W$$

where $W=[w1, w2, \ldots, wj, \ldots, wm]^T$ is mX1 vector, denoting the minimum nutritional requirement. M is a mXn matrix, Q is nXd matrix, and α is dX1 vector.

Explanation: the nutrition table gives us the nutrition of every ingredient. We have to multiple that with the Quantity to get the nutrition for each recipe, just like what we did for the Cost. Finally, we multiple again with the output vector α to yield the result for every nutrition. Make sure that value is greater than a threshold wj.

time:

$$\sum_{k}^{d} \alpha k \cdot tk <= T$$

let T denotes total time. we have to make all dishes within appropriate time. why only one constraints? Are you cooking all dishes all at once or one after another?

Evaluation:

The problem is **convex** because the objective function is affine. The constraints are affine as well.

The problem is non-trivial because

i) the objective is to make a diet planning system, which is very useful for our daily needs.

ii) there are many variables in the problems. For example, we have variables like cost, time, preference, quantity of ingredient for a recipe. These are not variables.

iii) systems can be more interesting if we try to make recommendation not for just one meal,but for a day(three meals), or a week. For a week, we can explicitly state that we can not have a meal that's the same as we had from last three meals. Or, we can do something like a exponential decay for the preference, saying that if we had the meal the same as last time, the preference is

↑ In the formulation, you need to specify which dish is for which meal.

going to drop to a certain level.

Development phases:

Phase I: do diet planning just for one meal

Phase II: do it for a day (three meals). So we output three vectors, each denotes what we should have each meal. Notice that there will be changes to the constraints, as we set time limit for each meal (not a day), but set cost limit fora day.

Phase III: do it for a week. We should consider some rules that make meals had consistently less tasty by adding some factors. For example, we can state that we can not have the same meal from last three. we will do this if have time.

Next step:

- Find data: Recommend nutritional intake table & nutritional table and ingredients for recipes
- Write code
- delve into the problem algebraically. i.e. Find according dual problem, etc.

What do you plan to do w/ the dual problem?