More LP

Frank

November 27, 2021

1 LP Definitions Again

1.1 Definitions

- **Objective Function**: Function being min/maximized.
- **Solution**: Assignment of values in R to x_j's
- **Feasible Solution**: Solution that satisfy all constraints
- **Feasible set/region**: Set of feasible solutions
- **Basic Feasible Solution**: Vertex of the feasible region
- **Optimal Solution**: Feasible solution that optimizes the objective function
- **Value of Solution**: value of the objective function at that solution.
- **Optimal Value**: Value of objective function at optimal solution
- **Feasible LP**: Has feasible solutions
- **Bounded LP**: has optimal solutions

1.2 Input/Output

- **input**: $\vec{c}, \vec{b}, A$
- **output**: $\vec{x}$

2 Linear Programming Algorithms

2.1 Simplex Algorithm (High Level)

- Danzig 1947, Kantorovich 1939
- Consider the corn problem, simplex algorithm only looks at basic feasible solutions and starts at an arbitrary one.

  Then it compares the value of each solution with its neighbours to figure out the optimal solution.
- Relies on the convex nature of the feasible region. Therefore, if a vertex is found to be a local optimal, then it is a global optimal. This does not apply to non-convex polygons.
- Good in practice, but exponential in the worst case.

2.2 Ellipsoid Algorithm

- Khachian 1979
- Polynomial worst-case, but slow in practice

2.3 Interior Point Algorithm

- Karmarkar 1984
- Polynomial, competitive with simplex algorithm in practice

  Polynomial in

  - Number of variables n
  - Number of constraints m
  - Size of the numbers in $\vec{c}, \vec{b}, A$

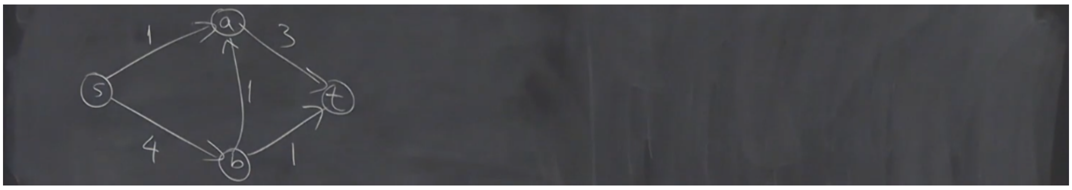
2.4 In class

- We can assume that there are LP algorithms that solves the problem in poly-time.

3 Reduction of Problems to LP

3.1 Max Flow Problem

- Suppose we have following flow network



- **Variables**:
  - x_e $\forall$ edge e. [x_e = traffic on e]

    $x_{sa}, x_{sb}, x_{ba}, x_{at}, x_{bt}$ in this case
- **Objective Function**: maximize $x_{sa} + x_{sb}$ or in general, max of $\sum_{e \in out(s)} x_e$
- **Constraints**:
  - $x_{ba} \geq 0$, $x_{ba} \leq 1$, and similarly for other edges. In general, $\forall e$: $x_e \geq 0$, $x_e \leq c(e)$ [capacity]
  - $\forall u \neq s, t$ $\sum_{e \in in(u)} x_e - \sum_{e \in out(u)} x_e = 0$ [conservation]

3.2 Line Fitting Problem

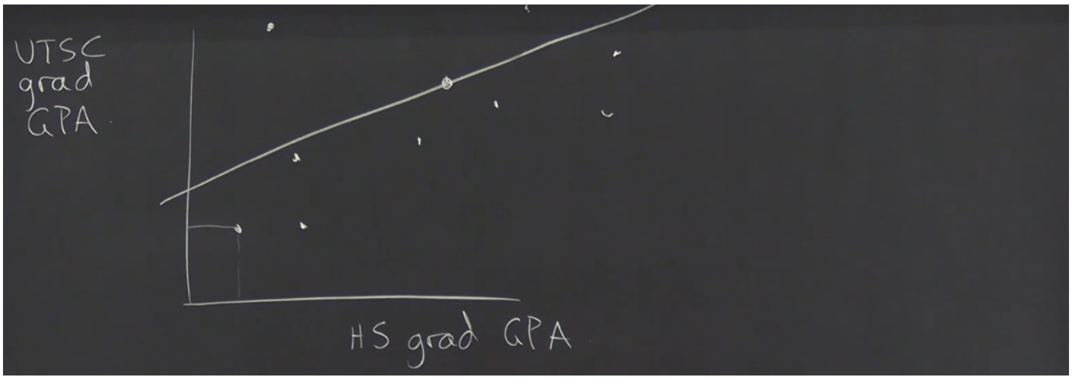
3.2.1 Line Fitting/Linear Regression Problem

- How "outcomes" depend on "attributes" of "population"

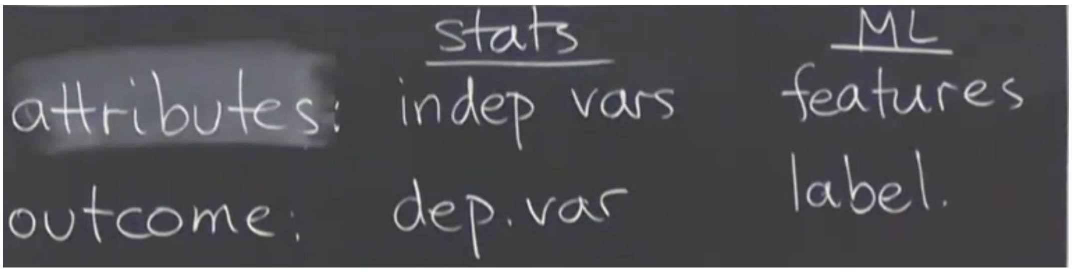
  For example, for admission to the CS program, the outcome might be graduation with a certain gpa. The attributes might be highschool grades plus A48, A67, A37 grades, and the population might be the students.

  Now we want a model to predict the outcome of an instance of a population based on the attributes

  Let do this in 2-D, with x-axis being highschool grades and y-axis being UTSC grad GPA. Then we are looking for a best fitting line (linear function) that best fits the data



- **input**: points on the graph (attribute vs. outcome)
- **output**: the best fitting line ($ax + b$)
- Formal definition of terms:



3.2.2 Line Fitting Reduction into LP

- **Input**:
  1. m points $P_i = (P_{i1}, \dots, P_{id})$ where $P_{ij} \in \mathbb{R}$ is value of attribute j in individual i
  2. m labels $l_i \in R$ is outcome for individual i

- **Output**: Linear function that "best fits" input

  We want $h : \mathbb{R}^d \to R$, maps d attributes to an outcome

  For example, $h(x_1, \dots, x_d) = a_1 x_1 + \dots + a_d x_d + b$, where $a_1, \dots, a_d$ and b are unknowns

  Then we want $\vec{a}$ and b that best fits the inputs $x_1, \dots, x_d$

  So we want a best fitting line of a set of points that has d dimensions for attributes and one more dimension for the outcome

- Best fit:
  - Define an error function $E_i(\vec{a}, b) = |h(P_i) - l_i| = |(b + \sum_{j=1}^{d} a_j P_{ij}) - l_i|$, where l_i is the actual outcome and $h(P_i)$ is the predicted outcome w.r.t. individual i

    Then $E_i(\vec{a}, b)$ is the error w.r.t. individual i

    Then $E(\vec{a}, b) = \sum_{i=1}^{m} E_i(\vec{a}, b)$ is the total error
  - Thus, we are trying to minimize $E(\vec{a}, b)$ total error

- Linear Programming:
  - **Variables**: $a_1, a_2, \dots, a_d, b$
  - **Objective Function**: minimize $\sum_{i=1}^{m} |(b + \sum_{j=1}^{d} a_j P_{ij}) - l_i| = E(\vec{a}, b)$, where we want $\vec{a}, b$ given P_i, l_i
  - **Constraints**: no constraints

  Notice we have a big problem, our objective function is an absolute value function, not a linear function

  Solution:
  - We want to minimize $|x|$

    $|x| = max(x, -x)$

    Then let $y = |x|$
  - The problem now becomes minimizing y with the constraint that $y \geq x$ and $y \geq -x$.

    Note that y is unbounded, however we are minimizing y, so then minimum of y is equal to the minimum of $|x|$, since $y = |x|$

  Then we introduce new variables $y_1, \dots, y_m$ with the intention of $y_i = E_i(\vec{a}, b)$, then

  Variables: $\vec{a}, \vec{y}, b$ but still want $\vec{a}, b$, since $\vec{y}$ is placeholder

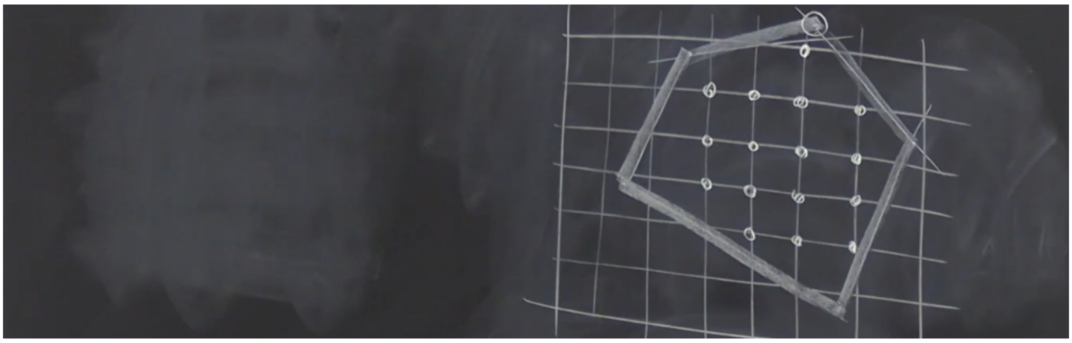
  Objective Function: minimize $\sum_{i=1}^{m} y_i$

  Constraints: where $y_i \geq (b + \sum_{j=1}^{d} a_j P_{ij}) - l_i$ and $y_i \geq l_i - (b + \sum_{j=1}^{d} a_j P_{ij})$

4 Integer LP & 0-1 LP

4.1 Problem

- We can still consider a feasible region within a set of constraints, however, our solutions would be a finite set of points within this region



Note that vertices no longer has to have the optimal solution

4.2 ILP: Integer LP

- LP with additional constraint where $x_j \in \mathbb{Z}$ which is non-linear

4.3 0-1 LP

- $x_j \in \{0, 1\}$
- Note that both ILP and 0-1 LP are NP-hard

4.4 Knapsack Problem

- **Input**:
  - items $1, \ldots, n$
  - item i has value v_i and weight w_i
  - Knapsack has capacity C
- **Output**: Find which items to take to maximize the value of the knapsack (discrete knapsack: 0 means do not take, 1 means take)
- Can be reduced to 0-1 LP
  - **Variables**: x_i for each item i, where $x_i = 1$ if item i is taken; 0 o/w.
  - **Objective Function**: maximize $\sum_{i=1}^{n} v_i x_i$
  - **Constraints**: $\sum_{i=1}^{n} w_i x_i \leq C$ [linear], and $x_i \in \{0, 1\}$ [non-linear]
- Note that replacing the constraint with $x_i \geq 0, x_i \leq 1$ gives the fractional knapsack problem

4.5 Vertex Cover Problem

- **Input**:
  - $G = (V, E)$ undirected
- **Output**: $V' \subseteq V$ s.t. $\forall\{u, v\} \in E$, $u \in V'$ or $v \in V'$
- Consider the graph, and a minimum vc

- Can be reduced to 0-1 LP
  - **Variables**: x_u for all $u \in V$ where $x_u = 1$ if u is in vc; 0 o/w
  - **Objective Function**: minimize $\sum_{u \in V} x_u$
  - **Constraints**: $\forall\{u, v\} \in E$, $x_u + x_v \geq 1$ and $\forall u \in V$, $x_u \in \{0, 1\}$ [non-linear]

    Note that we can also express the first constraint with $\forall\{u, v\} \in E$, either $x_u = 1$ or $x_v = 1$, which is a boolean constraint. But we turned this constraint into a linear constraint.