Introduction

September 8, 2021 4:57 PM

Optimization

- Objective function $f : \Omega \subset \mathbb{R}^n \to \mathbb{R} \cup \{-\infty, \infty\}$.
- Goal: maximize/minimize $f(x)$.
- Constraints: $x \in C \subset \Omega$,
  - C is called the feasible set.
  - Feasibility: whether the LP is feasible (the feasible set is not empty).
- Constrained optimization: minimize/maximize $f(x)$ such that $x \in C$.

Linear programming (LP)

- Linear functions: $\exists \vec{a} \in \mathbb{R}^n, \forall \vec{x} \in \mathbb{R}^n, f(x) = \vec{a} \cdot \vec{x} = a_1 x_1 + \cdots + a_n x_n$.
- Linear constraints: $\exists \vec{a}, \vec{b} \in \mathbb{R}^n, \vec{a} \cdot \vec{x} >=< \vec{b}$.
- LP is optimization with f a linear function and $x \in C$ all linear (linear constraints).
  - $\vec{x} = (x_1, x_2, \dots, x_n)$ are the decision variables.

Assignment problem:

- N tasks, N people match/assign tasks in an optimal way
- Goal: C_{ij} benefit of assigning to person i the task j per amount of time
- Decision variable: X_{ij}, portion of time of person i spent on task j
- Objective function: $\sum_{i=1}^{N} \sum_{j=1}^{N} X_{ij} C_{ij}$
- Constraints
  - Amount of time spent by person i: $\sum_j X_{ij} = 1$.
  - Non negativity: $0 \leq X_{ij} \leq 1$
  - Each task j must be accomplished fully: $\sum_i X_{ij} = 1$

Standard inequality form:

- Maximization of objective function
- Subject to linear inequalities: less than or equal to some constant
- Each variable is non-negative.
- Matrix notation
  - Max $c \cdot x$.
  - Such that $Ax \leq b$.
  - $x \geq 0$.
- Reduction to standard form
  - $\min f = -\max(-f)$.
  - $\max(f + const) \Leftrightarrow \max f$.
  - $f(x) \geq a \Leftrightarrow -f(x) \leq -a$.
  - $x = a \Leftrightarrow x \leq a$ and $-x \leq -a$.
  - $x \geq a \Leftrightarrow x' = x - a \geq 0$.
  - $x \leq a \Leftrightarrow x' = a - x \geq 0$.
  - No restriction on $x \Leftrightarrow x = x^+ - x^-, x^+, x^- \geq 0$.

Difficulties in solving LP problems

- Finding a solution can be costly
- Geometric method not practical for high dimensions
- Need to find effective algorithms for high dimensional algebraic methods

Geometric intuition

- Feasible set is a polytope
- Objective function is linear
- Maximum occurs at a vertex of the polytope

- A vertex is the intersection of n hyperplanes in $\mathbb{R}^n$

Half space

- Let $\vec{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, $\vec{a} = (a_1, \dots, a_n) \in \mathbb{R}^n$
- A half space is defined as $H_{\vec{a},b} = \{\vec{x} \in \mathbb{R}^n : \vec{a} \cdot \vec{x} \leq b\}$.
  - $\vec{a}$ is an orthogonal vector to the half space
- Intersections of half spaces are polytope (n dimensional polygon)
- Simplex (n dimensional triangles)
  - 1-simplex: line interval.
  - 2-simplex: triangle
  - 3-simplex: tetrahedron

Hyperplanes:

- $P = \{\vec{x} \in \mathbb{R}^n : \vec{a} \cdot \vec{x} = b\}$.
  - 1-d: point
  - 2-d: line
  - 3-d: plane
- They are boundaries of half spaces
- In n-dimension, the hyper plane has dimension $n - 1$
- Intersection of 2 hyper planes has dimension $n - 2$
- Intersection of n hyper planes
  - Let $L_1, L_2, \dots, L_n = \{\vec{x} : \vec{a_l} \cdot \vec{x} = b_l\}$
  - Intersection of m hyperplanes of dimension $n - 1$ has dimension $n - m$.
    - This holds when all $\vec{a_l}$ are linearly independent
  - In n dimensions, need n hyperplanes to get a single intersection point.

Convexity

- A set $S \subset \mathbb{R}^n$ is convex if $\forall x, y \in S$, $\forall t \in [0,1]$, $(1 - t)x + ty \in S$.
- $\mathbb{R}^n$ are convex
- Half spaces are convex
- Intersection of two convex sets is convex
  - Feasible regions are convex as the intersection of half spaces
- A straight line is convex
- An empty set is convex
- Hyperplanes are convex

Simplex method

2021年9月13日 14:29

Standard equality form:

- Turns inequalities into equalities using slack variables $(x_{n+1}, \dots, x_{n+m})$
- With standard form, we add slack variables to get standard equality form:
  - Maximize: $c_1 x_1 + \cdots c_n x_n$
  - $a_{11}x_1 + \cdots + a_{1n}x_n + x_{n+1} = b_1$.
  - $a_{21}x_1 + \cdots + a_{2n}x_n + x_{n+2} = b_2$.
  - $a_{m1}x_1 + \cdots + a_{m+n}x_n + x_{n+m} = b_m$.
  - $x_1, x_2, \dots, x_{n+1}, \dots, x_{n+m} \geq 0$.
- In matrix vector forms
  - $\vec{x} = (x_1, \dots, x_n, x_{n+1}, \dots, x_{n+m}) \in \mathbb{R}^{m+n}$
  - $\vec{c} = (c_1, \dots, c_n, 0, \dots, 0) \in \mathbb{R}^{m+n}$.
  - $A = \begin{pmatrix} a_{11} & \dots & a_{1n} & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{m1} & \dots & a_{mn} & 0 & \dots & 1 \end{pmatrix}$.
    - Append an identity matrix to the right

Gradients of a function:

- $f: \mathbb{R}^n \to \mathbb{R}, \nabla f(x) = \left(f_{x_1}(x), f_{x_2}(x), \dots, f_{x_n}(x)\right)$.

Dictionary

- Objective function
- Standard equality form with slack variables on one side of the equations
- E.g. The following is a dictionary
  - $z = 4x_1 + 3x_2$.
  - $x_3 = 4 - 2x_1 - x_2$.
  - $x_4 = 3 - x_1 - x_2$.
  - $x_5 = \frac{3}{2} - x_1$.
  - $x_1, x_2, \dots, x_5 \geq 0$.
- The LHS of the dictionaries (Slack variables) are basic variables (Base)
  - A total of m variables
- The RHS of the dictionaries (original variables) are non-basic variables
  - A total of n variables
- Basic solution: when the non-basic variables are all zero.
- Creating new dictionaries by iteration
  - Switch variables to be 0 to move to the next vertex.
  - We stop when the objective function is like $z = c_0 - c_1 x_i - c_2 x_j$ .... (c_0 is optimality)
- A dictionary is feasible if the basic solution is feasible
- A solution is feasible if all variables (basic or non-basic) are non-negative
- If a dictionary is feasible, then the LP problem is feasible
  - The converse is not true
- Entering variable: non-basic to basic
  - Most positive (largest) coefficient in the objective function
- Leaving variable: basic to non-basic
  - Minimum increase of entering variable (so that we keep within our feasible region)
- Anstee's rule:
  - Choose the entering variable with the largest positive coefficient (in objective function)
  - If there is a tie, then choose the one with the smaller subscript
  - If there is a tie in the leaving variable, then choose the one with the smallest subscript

Unboundedness can cause troubles to the simplex method

- If an entering variable causes all leaving variables positive (no leaving variable), then the LP problem is unbounded
- If a dictionary is feasible but there is no leaving variable, then the LP is unbounded

2-phase simplex method

- Phase 1: solve an auxiliary problem, which is an LP for feasibility, and find a feasible dictionary
    - Given the original problem
        - $z = c_1x_1 + \cdots + c_nx_n$.
        - $x_i = b_i - a_{i1}x_1 - \cdots - a_{in}x_n$.
    - Auxiliary problem:
        - Maximize $-x_0$.
        - $a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n - x_0 \leq b_i$ $(i = 1,2, \dots, m)$.
        - $x_j \geq 0, j = 0,1,2, \dots, n$.
        - When swapping, choose the variable that has the smallest constant (most negative)
            - i.e. Choose the variable that increase x_0 the most
        - We swap until all constants are positive, then we got a feasible dictionary
            - We could apply simplex algorithm on it directly
        - The optimal dictionary provides a feasible dictionary for the original problem
            - $x_0 = 0$ means the original problem is feasible, and the final dictionary is feasible for the origianl problem. (not necessarily have an optimal solution)
            - $x_0 > 0, z < 0$ means that the original problem is unfeasible
    - Note:
        - There are LP problems whose auxiliary problem has a non-zero maximum value
        - If the auxiliary problem has a non-zero maximum value, then the original problem does not have a solution
- Phase 2: solve original LP, starting from the result of phase 1
    - Take all constraint equations and remove $x_0 = 0$
    - Use the objective function of the original problem and replace any basic variable by its equation
    - Apply simplex algorithm

Degeneracy

- Degeneracy happens right after we had a tie in the choice of leaving variable
- A dictionary is degenerate if one of the basic variable is equal to zero
- Degeneracy may (not always) make the simplex algorithm stay at the same vertex
    - When leaving a new variable in a degenerate dictionary, we may not change the vertex.
    - Otherwise, we may go out of the feasible region
- Geometrically, degeneracy happens when more than n linearly independent hyperplanes intersect at the same vertex
    - 2D, more than 2 lines
    - 3D, more than 3 planes

The more variables, the more iterations of the simplex method may typically be required
At a degenerate feasible dictionary, which is not the final dictionary, the next dictionary must be different from it.
If all corner points of the feasible region are non-degenerate, then the simplex algorithm must terminate in a finite number of iterations

If a dictionary is feasible, non-degenerate, and has an entering variable with $c_i > 0$, then pivoting will change to a strictly different vertex

- We can increase the optimal value, so we have to change the vertex

In LP, there is a finite number of constraints, so the feasible region has a finite number of vertices.

- Maximum number of vertices = $\binom{n+m}{n}$ with n variables and m constraints.
- Typical LP problems have a complexity of $O(n^3)$.

If an LP problem has no degenerate point, then the simplex method to solve it will terminate in a finite number of steps

Summary
- There are finitely many vertices
- There are finitely many dictionaries per vertex
- If the simplex method runs infinitely it must visit the same dictionary more than once

Cycling
- We reach the same dictionary again after several iterations (pivoting). We might cycle forever
- Pertubation method
  - Untangling a degenerate point
  - Add small constants to the constraints to remove degeneracy
- Bland's rule
  - In case of a tie, the entering variable should be chosen according to the smallest subscript
  - The leaving variable should also be the one with smallest subscript

Fundamental theorem of linear programming
- For an arbitrary LP problem in standard form, the following statements are true
  - If an LP has no optimal solution, then it is either infeasible or unbounded
  - If a standard form LP has a feasible solution, then it has a basic feasible solution
  - If a standard form LP problem has an optimal solution, then it has a basic optimal solution (vertex).
  - There are LP problems that have infinitely many optimal solutions

Uniqueness of optimal solutions
- In 2D, a zero-coefficient for an entering variable gives freedom for that variable to change, as long as all variables stay positive, without changing the objective value
- In 3D, if x_1, x_2, x_3 are all basic optimal solutions, then all convex combinations of them are optimal
  - For $x_1, x_2, \dots, x_k$, the convex combinations are of the form $\sum_{i=1}^{k} \lambda_i x_i$, $\lambda_i \geq 0$ and $\sum_{i=1}^{k} \lambda_i = 1$.
- By doing pivots with zero-coefficient along variables, we can get all basic optimal solutions

Duality

2021年10月13日 14:02

Every feasible solution provides a lower bound on the optimal objective value z^*.

We then want to find an upper bound on the optimal value

- Suppose we have k constraints $C_1, \dots, C_k$
- Objective function $z = a_1 x_1 + \cdots a_n x_n$.
- Let $y_1, y_2, \dots, y_k \geq 0$, consider $y_1 C_1 + \cdots y_k C_k$.
  - We can simplify it to $f_1(y_1, \dots, y_k)x_1 + \cdots + f_n(y_1, \dots, y_k)x_n \leq f(y_1, \dots, y_k)$.
- Then, to get the tightest upper bound, we have
  - $f_1(y_1, \dots, y_k) \geq a_1$.
  - $f_n(y_1, \dots, y_k) \geq a_n$.
  - $f(y_1, \dots, y_k)$ is the tightest upper bound we can have. We want to minimize it.
  - This is the dual problem.
- The original problem is called primal problem.

Primal and dual problems

- Both linear
- Flipped role of the constraints and decision variables
  - Primal: n variables, k constraints.
  - Dual: k variables, n constraints.
- The coefficient matrices are transpose of each other
  - If the primal has $A\vec{x} = \vec{b}$, the dual has $A^T\vec{x} = \vec{b}'$.
- Objective function coefficients and constants of constraints are exchanged
  - If the primal has $\vec{b}$ as the constants of constraints and $\vec{c}$ as the objective function coefficients, then the dual problem has $\vec{b}$ as the objective function coefficients and $\vec{c}$ as the constants of constraints.
- Primal: max with $\leq$; Dual: min with $\geq$.

Matrix form

Primal	Dual
Max $c^T x$	Min $b^T y$
s.t. $Ax \leq b, x \geq 0$	s.t. $A^T y \geq c, y \geq 0$

Duality

- Not specific to linear programming
- 1-1 correspondence: to every primal, we have a dual
- Involution (Symmetric relation):
  - $dual(A) = B$, then $dual(B) = A$.
  - Dual of the dual is the primal

Weak duality theorem

- If $(x_1, \dots, x_n)$ is feasible for the primal and $(y_1, \dots, y_n)$ is feasible for the dual, then $\sum c_j x_j \leq \sum b_i y_i$.
- If x is primal feasible and y is dual feasible, then $c \cdot x \leq b \cdot y$.
- Max primal $\leq$ min dual.
  - Duality gap: $\max primal < \min dual$.
  - There is no gap in linear programming
- Quick proof: $c \cdot x = c^T x \leq (A^T y)^T x = y^T A x \leq y^T b = b \cdot y$.

Consider a primal and dual problem

- Minimize $c \cdot x$, subject to $Ax \geq b, x \geq 0$.
- Let x' be a feasible solution to the primal, y' be a feasible solution to the dual

- Then both the primal and dual must be bounded
  - Dual gives upper bound for the primal.
  - Primal gives lower bound for the dual.
  - If primal is unbounded, then dual is unfeasible
- It is possible that $c \cdot x' < b \cdot y'$ when we are not at optimal
- If $c \cdot x' = b \cdot y'$, then x' is an optimal solution to the primal and y' is an optimal solution to the dual

Strong duality
- If the primal problem has an optimal solution x^*, then the dual also has an optimal solution y^* such that $c^T x^* = b^T y^*$
- Moreover, from the objective function in the final dictionary of the primal, we find a dual optimal basic solution
- We can read a dual optimal solution as follows (read the solution to the dual from coefficients of the slack variables)
  - $y_i^* = -c_{n+i}^*$.

Remark: An optimal dictionary is not necessarily final
- Optimal: at a basic optimal solution
- Final: all coefficients in objective function are non-positive
  - $c_{n+i}^* \leq 0 \Rightarrow y_i^* \geq 0$.
- This is due to degeneracy

Since primal and dual are dual of each other
Sometimes it is easier to solve one or the other
- When # constraints > # variables, it is faster to solve the dual
- # simplex iterations is proportional to # rows in the dictionary and is relatively insensitive to the number of variables

Primal/dual	Optimal	Unbounded	Unfeasible
Optimal	Possible (strong duality)	Not possible	Not possible
Unbounded	Not possible	Not possible (weak duality)	Possible (weak duality)
unfeasible	Not possible	Possible (weak duality)	possible

Theorem of the alternative
- Let A and b be given, then exactly one of the following two must occur, but not both
- There exists x such that $x \geq 0$ and $Ax \leq b$.
- There exists y such that $y \geq 0$, $Ay \geq 0$ and $b^T y < 0$.

Certificate of optimality
- Consequence of strong duality
- When you think you have an optimal solution
  - Check the feasibility of the primal solution
  - Look for the supposed dual optimal solution, check its feasibility
  - Check whether primal and dual objective values agree $c^T x = b^T y$.

Complementary slackness
- Let $x^* = (x_1^*, \dots, x_n^*)$, $y^* = (y_1^*, \dots, y_n^*)$ be primal and dual feasible. Then (x^*, y^*) is optimal for primal and dual problems is equivalent to $\begin{cases} y_i^* \left(b_i - \sum_{j=1}^n a_{ij} x_j^*\right) = 0, \forall i \\ x_j^* \left(\sum_{i=1}^m a_{ij} y_i^* - c_j\right) = 0, \forall j \end{cases}$
  - Note $w_i = x_{n+i} = b_i - \sum_{j=1}^n a_{ij} x_j^*$ are the primal slack variables and $z_i = y_{m+j} = \sum_{i=1}^m a_{ij} y_i^* - c_j$ are the dual slack variables.
  - So $y_i^* x_{n+i}^* = 0$, $x_j^* y_{n+j}^* = 0$ for all i and j
- Complementary slackness cannot guarantee the solution is feasible. It only checks that $c \cdot x = b \cdot y$

- If we get more than n zeros in the optimal solution, we have a degeneration, and we will get inequalities when using complementary slackness
- If the point we are working on is not a vertex, we only get 1 zero, and we get too many equations.

Geometric interpretations

- $\forall j, y_{m+j} = \sum_{i=1}^{n} a_{ij} y_j - c_j$ gives n equations with $m+n$ unknowns
- At non-degenerate vertex for the primal
  - We get n complimentary slackness equations
  - We have a total $n+m$ equations with $n+m$ unknowns, unique solution.

For a primal problem

- If c changes, the optimal solution x^* may or may not change
  - Keep the ratio of c, x^* does not change.
- By duality, if b changes, the optimal solution y^* may or may not change

Penalty method/Lagrange multiplier

- Constrained problem: max $f(x)$ subject to $x \in C$.
- Unconstrained problem:
  - Max $f(x) + p(x)$.
  - Here $p(x) = \begin{cases} 0, x \in C \\ -\infty, x \notin C \end{cases}$ is hard penalty.
- Soft penalty:
  - Max $f(x) + g(x)$.
  - Here, $\begin{cases} g(x) \geq 0, x \in C \\ g(x) < 0, x \notin C \end{cases}$, $g(x)$ gets more negtive as x goes away from C.
- Using Lagrange multiplier, there exists λ such that max $f(x)$ = max $f(x) + \lambda g(x)$.
- Define $\pi(x, y) = c^T x + \sum_{i=1}^{m} y_i \left(b_i - \sum_{j=1}^{n} a_{ij} x_j\right) = c^T x + y^T (b - Ax) = b^T y + x^T (c - A^T y)$.
- Consider $\min_{y \geq 0} \max_{x \geq 0} \pi(x, y)$.
  - Firstly, $\max_{x \geq 0} b^T y + x^T (c - A^T y)$.
    - If $c - A^T y > 0$, then we get ∞.
    - If $c - A^T y \leq 0$, then we get $b^T y$ when $x = 0$.
  - Now, we get $\min_{y \geq 0, A^T y \geq c} b^T y$ (dual problem).
- Consider $\max_{x \geq 0} \min_{y \geq 0} \pi(x, y)$.
  - Firstly, $\min_{y \geq 0} c^T x + y^T (b - Ax)$.
    - If $b - Ax < 0$, then we get $-\infty$.
    - If $b - Ax \geq 0$, then we get $c^T x$ when $y = 0$.
  - Now, we get $\max_{x \geq 0, Ax \leq b} c^T x$ (primal problem).

Revised Simplex method

2021年11月19日 14:05

Let $A' = [A, I]$, b, c with A, b, c defined in the standard LP form
Goal: separate basic and non-basic variables
- Define x_B =basic variables, x_N = non basic variable.
- Key observation: a dictionary is determined by the choice of basic and non-basic variables
- Note: we can decompose A' into basic and non-basic parts $A' = [N, B]$ by reordering columns into basic/non-basic parts.

Now, $A'x = b \Rightarrow [N, B]\begin{pmatrix} x_N \\ x_B \end{pmatrix} = b \Rightarrow Nx_N + Bx_B = b$.
Basic variables can be expressed in terms of non-basic variables $\Leftrightarrow$ B is invertible
- $x_B = B^{-1}b - B^{-1}Nx_N$.
- The column vectors of B are linearly independent
  - They form a basis for $\mathbb{R}^n$.
- $y = Bx$ has a unique solution $x = B^{-1}y$

Coefficients of objective function $c = [c_N, c_B]$.
- $z = c^Tx = [c_N^T, c_B^T]\begin{pmatrix} x_N \\ x_B \end{pmatrix} = c_N^Tx_N + c_B^Tx_B = c_B^TB^{-1}b + [c_N^T - c_B^TB^{-1}N]x_N$.

So a dictionary can be written in general as
- $z = c_B^TB^{-1}b + [c_N^T - c_B^TB^{-1}N]x_N$ (scalar objective function).
- $x_B = B^{-1}b - B^{-1}Nx_N$ (vector constraints).
- Basic solution when $x_N = 0$ is $x_B' = B^{-1}b$.

Steps of revised simplex method
- Start with $x_N, x_B, [N, B] = [A, I], c_N, c_B, x_B'$
- For entering column
  - Solve $y = (B^T)^{-1}c_B$, so $y^T = c_B^TB^{-1}$.
    - Equivalent to solving $B^Ty = c_B$
  - Choose the entering column a in N which corresponds to a positive component of $c_N^T - y^TN$
- For leaving column
  - Solve $Bd = a$, so that $d = B^{-1}a$.
  - Find the largest $t \geq 0$ such that $x_B' - td \geq 0$
  - Choose the leaving column corresponding to a zero component of $x_B' - td \geq 0$ for the largest t.
- Update $x_N, x_B, N, B, c_N, c_B, x_B'$
- Repeat

Eta factorization
- Let E_i be the identity matrix with the leaving column of the ith dictionary replaced with $d_i = B^{-1}a_i$.
  - $d_1 = a_1$, $d_2 = E_1^{-1}a_2$, ..., $d_i = E_{i-1}^{-1}E_{i-2}^{-1} \dots E_1^{-1}a_i$.
  - Then $B_0 = I$, $B_1 = E_1$, $B_2 = E_1E_2$,...
  - $y^TB_k = y^TE_1E_2 \dots E_k$.
- It may not seem fast, but when there are a lot of variables and few constraints, it can get long to solve linear systems with B
  - Gaussian elimination has a complexity of $O(m^3)$ for an $m \times m$ matrix.
- Each solution of E_i requires
  - 1 operation for the row with a single non-zero
  - 2 operation for each other row
- Total $2(m-1)+1$ opetations, complexity $O(m)$.
- At pivot k, solving a linear system with B using eta factorizations takes $O(km)$ operations. Therefore, we need to do $O(m^2)$ simplex iterations for it to be more expensive than Gaussian eliminations
- In practice, the average number of iteratio required by the simplex algorithm to reach optimality is roughly m, the number of constraints.

Matrix decomposition (side note)
- ALU: split into lower and upper diagonal matrix

Dual problem in matrix form
- Dictionary
  - $w = -c_B^TB^{-1}b - (B^{-1}b)^Ty_B$.
  - $y_N = -(c_N - c_B^TB^{-1}N)^T + (B^{-1}N)^Ty_B$.
- This is the negative transpose of the primal dictionary
- Comparing with primal
  - Primal:

$c_B^TB^{-1}b$ (scalar)	$c_N^T - c_B^TB^{-1}N$ (vector of $(1, m)$)
$B^{-1}b$ (vector of $(n, 1)$)	$-B^{-1}N$ (matrix of (n, m))

  - Dual:

$-c_B^TB^{-1}b$ (scalar)	$-(B^{-1}b)^T$ (vector of $(1, n)$)
$-(c_N^T - c_B^TB^{-1}N)^T$ (vector of $(m, 1)$)	$(B^{-1}N)^T$ (matrix of (m, n))

- Dictionary meaning

Objective value	Coefficients of variables in objective function
Basic solution	Coefficients of constraints

- Note:
  - y_B is on the right (not as usual)
  - Primal basic x_B corresponds to y_B dual non basic.
    - $x_{n+i} \leftrightarrow y_i$.
  - Primal non basic x_N corresponds to y_N dual basic.
    - $x_j \leftrightarrow y_{m+j}$.
  - Entering variable in the primal ↔ leaving variable in the dual
  - Leaving variable in the primal ↔ entering variable in the dual

Dual dictionaries

- To each vertex in the primal is associated a vertex in the dual
  - Primal: n variables$= 0$.
  - Dual: m variables $= 0$.
  - Without degeneracy, there is the same number of vertices in both
  - The basic dual solution is one of the associated vertex in the dual
- Dual dictionary is unfeasible before optimality
  - By construction, $b \cdot y < c \cdot x^*$.
  - Not in the right range of values
- Link with complementary slackness
  - The basic solution of the dual dictionary always satisfies complementary slackness with the basic solution of the primal
  - Regardless of feasibility, y is feasible if and only if x is optimal
- Degeneracy
  - A dictionary whose basic solution is optimal is not necessarily final
  - The dictionary is final when we can't choose an entering variable anymore
- If we have a basic optimal solution for the primal, the dual is optimal, but the basic solution of the dual dictionary may not be optimal due to degeneracy

Theorem: suppose x^* is a non-degenerate feasible basic solution. If x^* is optimal, then the corresponding dual basic solution y^* is the unique feasible and optimal solution

Recall **primal /dual correspondences**:

- $x_j \longleftrightarrow y_{m+j}, \quad j = 1, \dots, n$
- $x_{n+i} \longleftrightarrow y_i, \quad i = 1, \dots, m$
- $\vec{x}_B \longleftrightarrow \vec{y}_B$
- $\vec{x}_N \longleftrightarrow \vec{y}_N$
- basic $\longleftrightarrow$ non-basic
- non-basic $\longleftrightarrow$ basic
- entering variable $\longleftrightarrow$ leaving variable
- leaving variable $\longleftrightarrow$ entering variable

primal

$$\begin{array}{rcll} z & = & z' & -\vec{y}_N'^{\,T}\vec{x}_N \\ \hline \vec{x}_B & = & \vec{x}_B' & -B^{-1}N\vec{x}_N \end{array}$$

dual

$$\begin{array}{rcll} -w & = & -z' & -\vec{x}_B'^{\,T}\vec{y}_B \\ \hline \vec{y}_N & = & \vec{y}_N' & +\left(B^{-1}N\right)^T\vec{y}_B \end{array}$$

Dual simplex method

- Dual pivot: the operation on the primal dictionary corresponding to the usual pivot on the dual dictionary
- From a feasible dictionary $x_B \geq 0$,
  - Pivot tries to achieve dual feasibility $y_N \geq 0$
  - If not possible, primal is unbounded
  - Pivot keeps feasibility of the dictionary
- From a dual feasible dictionary ($y_N \geq 0$)
  - Dual pivot tries to achieve primal feasibility $x_B \geq 0$
  - If not possible, dual is unbounded
  - Dual pivot keeps dual feasibility of the dictionary
- From a dictionary neither feasible or dual feasible, we can apply two phase method either to the primal or the dual problem

Dual pivot algorithm

- Start from a dual feasible dictionary
- Choose a leaving variable (with negative coefficient)
  - If none, then the dictionary is feasible, thus both primal and dual dictionaries are feasible. Thus, they are optimal
- If the pivot row has no positive coefficient, then the dual problem is unbounded, and the primal problem is unfeasible
- Otherwise, compare the ratio for each of the non-basic variables
  - $\frac{corresponding\ coefficient\ of\ the\ objective\ row}{positive\ coefficient\ of\ the\ pivot\ row}$.
- Choose the entering variable such that the ratio is the least negative

Sensitivity analysis

November 3, 2021 3:15 PM

Say we perturb b with a vector t so that we have the new LP problem
- Max $c \cdot x$
- Subject to $Ax \leq b + t$.

Can we find the new objective value?
- Consider the dual
  - Min $(b + t) \cdot y$.
  - Subject to $A^T y \geq c, y \geq 0$.
  - $z^{**} = (b + t) \cdot y^{**} = c \cdot x^{**}$
- If we don't change b too much, then $y^{**} = y^*$ and $z^{**} = z^* + t \cdot y^*$.
- The dual optimal y^* gives the rate of change of z^* under a small perturbation of b.
  - $z^{**} - z^* = t \cdot y^*$.

Theorem: for a primal problem, assume x^* is a non-degenerate basic primal optimal solution, y^* is a basic dual optimal solution, there exists $\epsilon > 0$ such that $|t_i| < \epsilon$ for $i \in [1, m]$, then the new LP problem (max $c \cdot x$, $Ax \leq b + t$) has optimal value $z^{**} = z^* + t \cdot y^*$.
- Degeneracy may cause problem

Note: the normal vectors of the boundaries are always outward
- If $A = \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_m \end{pmatrix}$, then the normal vectors should be a_i or $-e_i$ (to account for $x \geq 0$).
- Then $c = \sum_{i:w_i = x_{n+i} = 0} y_i \vec{a_i} + \sum_{j:x_j^* = 0} v_j^* \left(-e_j\right)$.
  - $c = A^T y - v$, i.e. v is the slack variables for the dual problem.
- For the dual: $b = \sum_{i\ s.t. v_i^* = 0} x_i^* \vec{a_i} + \sum_{j\ s.t. y_j^* = 0} w_j^* e_j$.

For a non-degenerate case, we have $n + m$ equations (x^*), n of them are zero, m of them are non-zero
- Primal
  - Find the set (size n) of hyperplanes where x^* lies
  - Value of x^* (intersection) is easy to find by solving a system of linear equations
- Dual
  - Find the set of n vectors combine to get c as a linear combination of a_i and e_j.
  - Weights y_i^* and v_i^* are easy to find once we have the right set.
    - As long as we have n linearly-independent vectors, we have a unique decomposition
    - The y_is that do not influence z^* are the hyperplanes that x^* does not lie on
      - $x_{n+i} > 0$.

If x^* is a basic primal optimal solution, non-degenerate, then the dual optimal solution y^* is unique

Economic analysis for dual problem in resource allocation
- Let y_i be the price makeup for company B to buy resources from company A
  - Makeup: how much more you should pay than the base price
- Non-negativity: otherwise, there is no profit for A, A will not sell.
- Constraints: How to ensure that y_i are high enough that company A is willing to sell at that price
  - Company A must make profit selling its resources than using them to make and sell each product

- Objective function: goal of company B
  - Maximize profit. i.e. minimize the cost
- At dual optimality, the prices y_i for each resources are the fair makeup prices since company A should be willing to sell them
  - If $y_i^* = 0$, B may not want to pay more.
  - If $y_i^* > 0$, B wants to pay more to get more.
- The y_i are also called marginal prices/values
  - Net/marginal value on top of the cost the company estimates resource i to be worth
- Strong duality:
  - Company A's maximum revenue from making and selling products = company B's minimum cost of purchasing the resources
  - Equilibrium under perfect competition: companies make no excess profits
  - In practice, it rarely happens. The market always tends to equilibrium, but it is always perturbed by factors
    - Changes in demand and supply, innovation, etc.
- Complementary slackness: If $x_{n+i}^* > 0$, the resource i is not utilized entirely, then the marginal price is 0.
  - It is exactly determined by y_i^*: $z^{**} - z^* = \sum_j y_j^* t_j$ for an increase in supply $b = b + t$.