EECS 127/227A, Fall 2022

Optimization Models in Engineering

Gireeja Ranade, UC Berkeley
Zhiyu An

Contents

1. Linear Algebra

1.a. Least-Squares Problem Statement

Definition 1.1 (Least Squares)
Assume matrix A and vectors $\vec{x}$ and $\vec{b}$. The problem defined by

$$\min_{\vec{x}} \|A\vec{x} - \vec{b}\|^2$$

is a Least Squares Problem (LSP).

Example 1.2
Assume we have two dimensional data set $\vec{x}$ and $\vec{y}$ and we want to formalize a LSP to find a linear correlation between x and y. We first formalize the goal linear correlation as

$$y = mx + c$$

where we want to find the optimal values for m and c to minimize the squared loss across all data points. Summarizing the above equation for all data points gives us

$$\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \\ x_n & 1 \end{bmatrix} \begin{bmatrix} m \\ c \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Where

$$A = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \\ x_n & 1 \end{bmatrix}, \quad \vec{x} = \begin{bmatrix} m \\ c \end{bmatrix}, \quad \vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

And therefore

$$\min_{\vec{x}} \|A\vec{x} - \vec{b}\|^2 = \min_{m,c} \sum_{i=1}^{n} (y_i - (mx_i + c))^2$$

Theorem 1.3 (Ordinary Least Squares)

Given the column space of the matrix A, for vector $\vec{b}$ not in the said column space, $A\vec{x} - \vec{b} = \vec{e}$ must be orthogonal to the columns of A. (Pythagora's theorem)

Therefore, the dot products of every column of A and $\vec{e}$ must be zero, i.e.

$$\begin{aligned} A^\top(A\vec{x} - \vec{b}) &= 0 \\ A^\top A\vec{x} - A^\top\vec{b} &= 0 \\ A^\top A\vec{x} &= A^\top\vec{b} \\ \vec{x} &= (A^\top A)^{-1}A^\top\vec{b} \end{aligned}$$

We conclude that the solution for Ordinary Least Squares (OLS) is

$$\vec{x}^* = \underset{\vec{x}}{\operatorname{argmin}} \|A\vec{x} - \vec{b}\|^2 = (A^\top A)^{-1}A^\top\vec{b}$$

1.b. Norm

Definition 1.4 (Norm)

A Norm is defined as

$$f : \mathbf{X} \to \mathbb{R}$$

For vector space $\mathbf{X}$.

The norm of x is denoted as $\|x\|$.

For any vector x and y, we have

- $\|x\| \geq 0$ and $\|x\| = 0$ iff $x = \vec{0}$
- $\|x + y\| \leq \|x\| + \|y\|$
- $\|\alpha x\| = |\alpha| * \|x\|$

Definition 1.5 (l-p Norm)

Generally, l-p norm is defined as

$$\|\vec{x}\|_p := \left(\sum |x_i|^p\right)^{\frac{1}{p}}; \quad 1 \leq p < \infty$$

Commonly used norms:

- $\|\vec{x}\|_1 = \sum |x_i|$
- $\|\vec{x}\|_2 = \sqrt{\sum |x_i|^2}$
- $\|\vec{x}\|_\infty = \max |x_i|$

Theorem 1.6 (Cauchy-Schwartz Inequality)

$$< \vec{x}, \vec{y} >= \vec{x}^\top \vec{y} = \|\vec{x}\|_2 \|\vec{y}\|_2 \cos\theta$$

Since $-1 \le \cos\theta \le 1$,

$$< \vec{x}, \vec{y} >= \vec{x}^\top \vec{y} \le \|\vec{x}\|_2 \|\vec{y}\|_2$$

Theorem 1.7 (Holder's Inequality)
For $p, q \ge 1$ s.t. $\frac{1}{p} + \frac{1}{q} = 1$,

$$|\vec{x}^\top \vec{y}| \le \sum_{i=1}^{n} |x_i y_i| \le \|\vec{x}\|_p \|\vec{y}\|_p$$

i.e., Cauchy-Schwartz is a narrowed case of Holder's Inequality.

1.c. Gram-Schimdt

Theorem 1.8 (Gram-Schimdt/QR-decomposition)
Let X be a vector space with basis $\{\vec{a}_1, \vec{a}_2, \dots, \vec{a}_n\}$, which is orthonormal. For any matrix A,

$$A = QR$$

$$\left[\vec{a}_1, \vec{a}_2, \dots, \vec{a}_n\right] = \left[\vec{q}_1, \vec{q}_2, \dots, \vec{q}_n\right] \begin{bmatrix} \vec{r}_{11} & \vec{r}_{12} & \cdots & \vec{r}_{1n} \\ 0 & \vec{r}_{22} & \cdots & \vec{r}_{2n} \\ 0 & 0 & \ddots & \vec{r}_{3n} \\ 0 & 0 & 0 & \vec{r}_{nn} \end{bmatrix}$$

Where Q is orthonormal andR is upper-triangular.

Theorem 1.9 (Fundamental Theorem of Linear Algebra)
For matrix $A \in \mathbb{R}^{m*n}$,

$$Null(A) \bigoplus Range(A^\top) = \mathbb{R}^n$$

Where $\oplus$ denotes "direct sum" and $Range(A^\top)$ is the column space of $A^\top$. With the said equation we can also conclude that

$$Range(A) \bigoplus Null(A^\top) = \mathbb{R}^m$$

Theorem 1.10 (orthogonal decomposition theorem)
X a vector space and S a subspace of X. Then for any $\vec{x}$ in X,

$$\vec{x} = \vec{s} + \vec{r}, \quad \vec{s} \in S, \quad \vec{r} \in S^{\perp}$$

Such that

$$S^{\perp} = \{\vec{r} \mid < \vec{r}, \vec{s} > = 0, \quad \forall \vec{s} \in S\}$$

Therefore,

$$\mathbf{X} = S \bigoplus S^{\perp}$$

Example 1.11 (Minimum Norm Problem)
We want to find

$$\min \|\vec{x}\|_2^2$$

subject to $A\vec{x} = \vec{b}$. From FTLA we know that

$$\vec{x} = \vec{y} + \vec{z} \;\; s.t. \;\; \vec{y} \in N(A; \;\; \vec{z} \in R(A^{\top}).$$

And

$$A(\vec{y} + \vec{z}) = 0 + A\vec{z} = \vec{b}$$

Since $\vec{y} \perp \vec{z}$,

$$\|\vec{x}\|_2^2 = \|y\|_2^2 + \|z\|_2^2$$

Consider $\vec{z} = A^{\top}\vec{w}$,

$$\begin{aligned} A\vec{z} &= \vec{b} \\ AA^{\top}\vec{w} &= \vec{b} \\ \vec{w} &= (AA^{\top})^{-1}\vec{b} \end{aligned}$$

Therefore

$$\vec{z} = \min \|\vec{x}\|_2^2 = A^{\top}(AA^{\top})^{-1}\vec{b}$$

1.d. Symmetric Matrices

Definition 1.12
Matrix A is symmetric if $A = A^{\top}$, i.e. $A_{ij} = A_{ji}$.
Set $\mathbb{S}^n$ means the set of symmetric matrices of dimension n.

Theorem 1.13 (Spectral Theorem)
If matrix A $\in \mathbb{S}^{\ltimes}$, then

- All eigenvalues of A are real numbers
- Eigenspaces are orthogonal
- $dim(N(\lambda_i I - A)) = \mu_i$ where μ_i is the algebraic multiplicity of λ_i

This means that A is always diagonalizable. i.e.:

$$A = U\Lambda U^\top$$

where U orthonormal and Λ diagonal. Orthonormal (or, unitary) means that the columns of U are orthogonal and all columns are normalized, i.e.

$$U^{-1} = U^\top$$

Remark 1.14
For a diagonalizable n*n matrix A that has n linearly independent eigenvectors, A can be factorized as

$$A = U\Lambda U^\top$$

Where U orthonormal and Λ is a diagonal matrix consists of the eigenvalues of A such that

$$\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_i \end{bmatrix}$$

Therefore it is also called an eigenvalue decomposition.

1.e. Principal Component Analysis

Definition 1.15
For $A \in \mathbb{S}$, its Rayleigh coefficient is defined as

$$R = \frac{\vec{x}^\top A \vec{x}}{\vec{x}^\top \vec{x}}$$

The Rayleigh coefficient can bound the eigenvalues of A such that,

$$\lambda_{min}(A) \le \frac{\vec{x}^\top A \vec{x}}{\vec{x}^\top \vec{x}} \le \lambda_{max}(A)$$

PCA is very similar to Singular Value Decomposition (SVD). SVD has more nice properties than PCA.

1.f. Singular Value Decomposition

Theorem 1.16 (SVD)
Let $A \in \mathbb{R}^{m*n}$, the SVD of A is given as

$$A = U\Sigma V^\top$$

Where

$$U \in \mathbb{R}^{m*m},\ \ \Sigma \in \mathbb{R}^{m*n},\ \ V \in \mathbb{R}^{n*n}$$

and Σ has real entries in its diagonal (the singular values) and zero's else where. If $Rank(A) = r$, we can rewrite A as

$$A = \sigma_1 \vec{u}_1 \vec{v}_1^\top + \sigma_1 \vec{u}_1 \vec{v}_1^\top + \cdots + \sigma_r \vec{u}_r \vec{v}_r^\top$$

Proof. For $A \in \mathbb{R}^{m*n}$, consider symmetric matrix $A^\top A$ that has eigenvalues $\lambda_1 \cdots \lambda_r > 0$ with corresponding eigenvectors $v_1 \cdots v_r$ and $\lambda_{r+1} \cdots \lambda_n = 0$. Then we know that

$$A^\top A \vec{v}_i = \lambda_i \vec{v}_i$$

Let

$$V = \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix}$$

Define $\sigma_i = \sqrt{\lambda_i}$, let

$$A\vec{v}_i = \sigma_i \vec{u}_i \ \ i \leq r$$

for some vector $\vec{u}_i$.

Claim. $\vec{u}_i$ are orthonormal.

$$\begin{aligned}
\vec{u}_i^\top \vec{u}_j &= \frac{(A\vec{v}_i)^\top}{\sigma_i} \frac{(A\vec{v}_j)}{\sigma_j} \\
&= \frac{1}{\sigma_i \sigma_j} \vec{v}_i^\top A^\top A \vec{v}_j && A^\top A \vec{v}_j = \lambda_j \vec{v}_j \\
&= \frac{1}{\sigma_i \sigma_j} \vec{v}_i^\top \lambda_j \vec{v}_j \\
&= \frac{\lambda_j}{\sigma_i \sigma_j} \vec{v}_i^\top \vec{v}_j && \vec{v}_i \vec{v}_j \text{ orthonormal} \\
&= \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}
\end{aligned}$$

Therefore $\vec{u}_i$ are orthonormal. Recall that A has rank r, we let

$$V_r = V = \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_r \\ | & & | \end{bmatrix}$$

Hence

$$AV_r = \begin{bmatrix} | & & | \\ \vec{u}_1 & \cdots & \vec{u}_r \\ | & & | \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix} = U_r \Sigma_r$$
$$A = U \Sigma V^\top$$

Since V orthonormal and $V^{-1} = V^\top$ ■

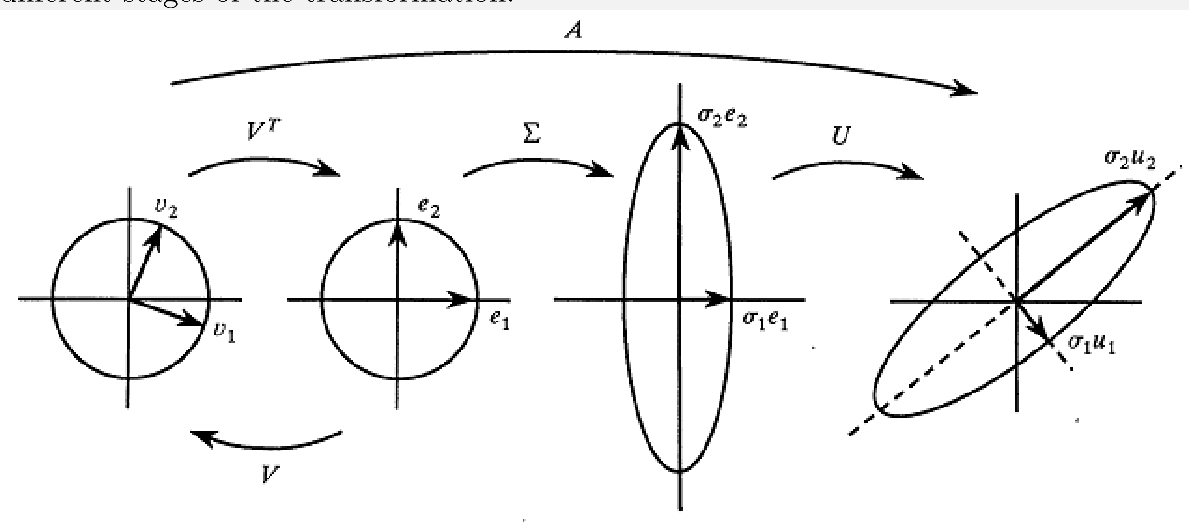
Remark 1.17 (geometric interpretation of SVD)
Consider linear transformation on vector $\vec{x}$ given by matrix A, s.t.

$$A\vec{x} = U\Sigma V^\top \vec{x}$$

SVD helps breaking the transformation into three smaller steps, i.e.

- orthonormal transformation (rotate/reflect) by V,
- scaling by Σ,
- orthonormal transformation by U.

The following illustration is an example of a 2D transformation $A\vec{x}$. It shows the decomposed linear transformation through the unit circles relative to the original unit circle at different stages of the transformation.



1.g. Low-Rank Approximation

Definition 1.18 (matrix norms)
There are two ways to interpret a matrix, either as an operator or as a block of data. Frobenius norm consider the matrix as a block of data.
Frobenius norm of matrix A is defined as

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2} = \sqrt{tr(A^\top A)}$$

Frobenius norm is invariant to orthonormal transformations, i.e. given U an orthonormal matrix,

$$\|UA\|_F = \|AU\|_F = \|A\|_F$$

Spectral norm, or l_2 norm, interpret the matrix as an operator and is defined as

$$\|A\|_2 = \max_{\|\vec{x}\|_2=1} \|A\vec{x}\|_2 = \max_{\|\vec{x}\|=1} \sqrt{\vec{x}^\top A^\top A\vec{x}} = \sqrt{\lambda_{max}(A^\top A)} = \sigma_{max}(A^\top A)$$

Intuitively, the spectral norm of a matrix A is the largest scaling that A can do (recall the Σ matrix that is used to scale the unit circle in the three steps of transformation after SVD).

Theorem 1.19 (Eckart-Young-Mirsky Theorem)
$A \in \mathbb{R}^{m*n}$. Do SVD gives us

$$A = U\Sigma V^\top = \sum_{i=1}^{n} \sigma_i \vec{u}_i \vec{v}_i^\top$$

Define

$$A_k = \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^\top$$

We want to find the best k-rank (lower than r) approximation of A, i.e.

$$\underset{B\in\mathbb{R}^{m*n},\, Rank(B)=k}{\text{argmin}} \|A - B\|_F$$

Suprisingly, Eckart-Young-Mirsky Theorem tells us that

$$\underset{B\in\mathbb{R}^{m*n},\, Rank(B)=k}{\text{argmin}} \|A - B\|_F = A_k$$

Moreover,

$$\underset{B\in\mathbb{R}^{m*n},\, Rank(B)=k}{\text{argmin}} \|A - B\|_2 = A_k$$

This theorem relates two completely different norms and is not obvious at all. It shows how fundamental SVD is, such that in any way of looking at a matrix, the decomposition shows up.

Remark 1.20

Eckart-Young-Mirsky Theorem can be used to **compress images**. For an image, the matrix that represents the pixels of the image can be reduced to a lower rank matrix, and hence a smaller set of data, while remains relatively high resolution. The A_k matrix **captures the key features of the image because it keeps k largest singular values and their corresponding vectors that contribute most to the dataset/transformation.**

Definition 1.21 (trace)

The trace of a matrix is defined as

$$trace := \mathbb{R}^{n*n} \to \mathbb{R}$$

$$trace(A) = \sum_{i=1}^{n} a_{ii}$$

Remark 1.22 (Orthonormal transformation invariance of Frobenius norm)

Proof that $\|UA\|_F = \|AU\|_F = \|A\|_F$

Proof. Recall that $\|A\|_F = \sqrt{tr(A^\top A)}$. By definition, for any matrices A and B, we have $tr(AB) = tr(BA)$ Then,

$$\begin{aligned}\|AU\|_F &= \sqrt{tr((AU)^\top (AU))} \\ &= \sqrt{tr(U^\top A^\top AU)} \\ &= \sqrt{tr(UU^\top A^\top A)} \\ &= \sqrt{tr(A^\top A)} \\ &= \|A\|_F\end{aligned}$$

■

Remark 1.23 (Frobenius norm is the sqrt of the sum of the squares of the singular values)

$$\begin{aligned}\|A\|_F = \|U\Sigma V^\top\|_F &= \|\Sigma\|_F \\ &= \sqrt{\sum_{i=1}^{n} \sigma_i^2}\end{aligned}$$

Proof of Eckart-Young-Mirsky
Goal: B: rank(k), $\|A-B\|_F \geq \|A-A_k\|_F$

Proof.

$$\|A-A_k\|_F = \|\sum_{i=k+1}^{n} \sigma_i \vec{u}_i \vec{v}_i\|_F = \sqrt{\sum_{i=k+1}^{n} \sigma_i^2}$$

Note that the goal is true iff

$$\sum_{i=1}^{n} \sigma_i^2(A-B) \geq \sum_{i=k+1}^{n} \sigma_i^2(A)$$

Further note that the previous statement is true iff:

$$\sigma_i^2(A-B) \geq \sigma_{k+i}^2(A)$$

Let $\sigma_{k+i}(A)$be the k+ith largest singular value of A. Hence

$$\sigma_{k+i}(A) = \sigma_{max}(A-A_k)$$

Denote A-B = C. Then

$$\sigma_i(A-B) = \sigma_i(C) = \|C-C_{i-1}\|_2$$

Since B has rank k,

$$\|B-B_k\|_2 = 0$$

Add it to the previous equation gives us

$$\begin{aligned}\sigma_i(A-B) &= \|C-C_{i-1}\|_2 + \|B-B_k\|_2 \\ &\geq \|C+B-C_{i-1}-B_k\|_2 \\ &\geq \|A-C_{i-1}-B_k\|_2\end{aligned}$$

Let $D = C_{i-1} + B_k$. Rank(D) $\leq$ i-1+k. Then

$$\sigma_i(A-B) \geq \|A-D\|_2$$

Consider the solution to the optimization problem

$$\underset{D,\, rank(D)\leq i+k-1}{\operatorname{argmin}} \|A-D\|_2 = A_k + i - 1$$

$$\min_{rank(D)\leq i+k-1} \|A-D\|_2 = \sigma_{k+1}(A)$$

Finally, bring the above result back to the previous equation gives us

$$\sigma_i(A-B) \geq \sigma_{k+1}(A)$$

as desired. ∎

2. Vector Calculus

Theorem 2.1 (Taylor's Theorem for Vectors)
For $f(\vec{x}) := \mathbb{R}^n \to \mathbb{R}$, the derivative of f is

$$f(\vec{x}_0 + \Delta\vec{x}) = f(\vec{x}_0) + \nabla f|_{\vec{x}=\vec{x}_0}^{\top} \Delta\vec{x} + \frac{1}{2!}(\Delta\vec{x})^{\top}\nabla^2 f|_{\vec{x}=\vec{x}_0}\Delta\vec{x}$$

Where

$$\text{Gradient} = \nabla f|_{\vec{x}=\vec{x}_0}^{\top}$$
$$\text{Hessian} = \nabla^2 f|_{\vec{x}=\vec{x}_0}$$

And

$$f(\vec{x}_0) + \nabla f|_{\vec{x}=\vec{x}_0}^{\top}\Delta\vec{x}$$

is the first-order approximation (a hyperplane).

Definition 2.2 (Gradient)
The gradiant $\nabla f(\vec{x})$ captures change according to all components of $\vec{x}$. It is defined as

$$\nabla f(\vec{x}) = \begin{bmatrix} \frac{\partial}{\partial x_1} f & \frac{\partial}{\partial x_2} f & \cdots & \frac{\partial}{\partial x_n} f \end{bmatrix}$$

The gradient always has the same dimension as the input vector.

Definition 2.3 (Hessian)
The hessian is a matrix that captures the change according to all gradients. It is defined as

$$\nabla^2 f(\vec{x})_{ij} = \frac{\partial f}{\partial x_i \partial x_j}$$

Hessian is **often** symmetric.

Example 2.4

Let

$$f(\vec{x}) = \|x\|_2^2, \quad f := \mathbb{R}^2 \to \mathbb{R}$$

Then the gradient of this function f is

$$\nabla f(\vec{x}) = \begin{bmatrix} 2x_1 \\ 2x_2 \end{bmatrix} = 2\vec{x}$$

And the hessian is

$$\nabla^2 f(\vec{x}) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

According to taylor theorem,

$$\begin{aligned} f(\vec{x} + \Delta\vec{x}) &= (x_1^2 + x_2^2) + \begin{bmatrix} 2x_1 & 2x_2 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \Delta x_1 & \Delta x_2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix} \\ &= x_1^2 + x_2^2 + 2x_1\Delta x_1 + 2x_2\Delta x_2 + \Delta x_1^2 + \Delta x_2^2 \\ &= (x_1 + \Delta x_1)^2 + (x_2 + \Delta x_2)^2 \end{aligned}$$

Example 2.5

Let

$$f(\vec{x}) = \vec{x}^\top \vec{a} = \sum_{i=1}^{n} x_i a_i$$

Then the gradient of this function f is

$$\nabla f(\vec{x}) = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \vec{a}$$

And the hessian is

$$\nabla^2 f(\vec{x}) = 0$$

Example 2.6

Let

$$f(\vec{x}) = \vec{x}^\top A\vec{x}$$

We can see that

$$\begin{aligned} f(\vec{x}) &= \vec{x}^\top A\vec{x} \\ &= \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix} \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \\ &= \sum_i \sum_j x_i a_{ij} x_j \end{aligned}$$

Since all terms that contain x_i is

$$\sum_{j \neq i} x_i a_{ij} x_j + \sum_{j \neq i} x_j a_{ji} x_i + x_i^2 a_i i$$

We know that

$$\frac{\partial f}{\partial x_i} = \sum_j (a_{ij} + aji) x_j$$

Therefore the gradient of this function f is

$$\nabla f(\vec{x}) = (A + A^\top) \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = (A + A^\top)\vec{x}$$

The hessian is

$$\nabla^2 f(\vec{x}) = A + A^\top$$

Theorem 2.7 (The Main Theorem)

Let $f : \mathbb{R}^n \to \mathbb{R}$ and f is differentiable everywhere. Consider the optimization problem subject to

$$\underset{\vec{x},\, \vec{x} \in \Omega}{\operatorname{argmin}} f(\vec{x})$$

Where Ω is an open set in $\mathbb{R}^n$

Then if $\vec{x}^*$ is an optimal solution, then

$$\frac{df}{dx}(x^*) = 0$$

Note that the converse is not necessarily true.

3. Regression

3.a. Sensitivity

Definition 3.1 (problem statement)
Consider optimization problem

$$A\vec{x} = \vec{y}$$

Under the special case that $A \in \mathbb{R}^{n*n}$ and is invertible. Now we apply a change to y such that $\vec{y} \to \vec{y} + \delta\vec{y}$. Because of this, $\vec{x} \to \vec{x} + \delta\vec{x}$. How big is $\delta\vec{x}$?

Theorem 3.2 (condition number)
The value we are interested in is $\frac{\|\delta\vec{x}\|_2}{\|\vec{x}\|_2}$. To investigate this value, we transform the equation such that

$$\begin{aligned} A(\vec{x} + \delta\vec{x}) &= \vec{y} + \delta\vec{y} \\ A\delta\vec{x} &= \delta\vec{y} \\ \delta\vec{x} &= A^{-1}\delta\vec{y} \\ \|\delta\vec{x}\|_2 &= \|A^{-1}\delta\vec{y}\|_2 \end{aligned}$$

Recall that

$$\|A\|_2 = \max_{\|y\|_2=1} \|A\vec{y}\|_2 = \max_y \frac{\|A\vec{y}\|_2}{\|y\|_2} = \sigma_{max}$$

Therefore by the definition of the spectral norm,

$$\|\delta\vec{x}\|_2 = \|A^{-1}\delta\vec{y}\|_2 \leq \|A^{-1}\|_2\|\delta\vec{y}\|_2$$

This gives us an upperbound of the solution. To find the lowerbound,

$$\begin{aligned} &A\vec{x} = \vec{y} \\ &\|\vec{y}\|_2 = \|A\vec{x}\|_2 \leq \|A\|_2\|\vec{x}\|_2 \\ &\|\vec{x}\|_2 \geq \frac{\|\vec{y}\|_2}{\|A\|_2} \end{aligned}$$

Combining these two inequalities gives

$$\begin{aligned} \frac{\|\delta\vec{x}\|_2}{\|\vec{x}\|_2} &\leq \frac{\|A^{-1}\|_2\|\delta\vec{y}\|_2}{\|\vec{y}\|_2/\|A\|_2} \\ &\leq \|A\|_2\|A^{-1}\|_2\frac{\|\delta\vec{y}\|_2}{\|\vec{y}\|_2} \\ &\leq \left(\frac{\sigma_{max}}{\sigma_{min}}\right)\frac{\|\delta\vec{y}\|_2}{\|\vec{y}\|_2} \end{aligned}$$

The term $\frac{\sigma_{max}}{\sigma_{min}}$ is called the condition number of a matrix. If the condition number is large, a small change in y would cause a large change in x.

3.b. Shift property of eigenvalues

Theorem 3.3 (Shift property of eigenvalues)
Consider matrix A. We add a diagonal matrix to A and change it to $A + \lambda I$. Then for λ_1 and $\vec{v}_1$ be the first eigenpair of A,

$$(A + \lambda I)\vec{v}_1 = A\vec{v}_1 + \lambda\vec{v} = \lambda_1\vec{v}_1 + \lambda\vec{v}_1 = (\lambda_1 + \lambda)\vec{v}$$

The eigenvalue of the new matrix $A + \lambda I$ is shifted by λ, but its eigenvector remain unchanged.

3.c. Ridge Regression

Theorem 3.4 (Ridge regression)
Consider the optimization problem

$$\min_{\vec{x}} \|A\vec{x} - \vec{b}\|^2 + \lambda^2\|\vec{x}\|_2^2$$

Where $\lambda^2\|\vec{x}\|_2^2$ is called the **regularizer**. We have

$$\begin{aligned} f(\vec{x}) &= (A\vec{x} - \vec{b})^\top(A\vec{x} - \vec{b}) + \lambda^2\vec{x}^\top\vec{x} \\ &= \vec{x}^\top A^\top A\vec{x} - \vec{x}^\top A^\top\vec{b} - \vec{b}^\top A\vec{x} + \lambda^2\vec{x}^\top\vec{x} + \vec{b}^\top\vec{b} \end{aligned}$$

The gradient of f is

$$\nabla f(\vec{x}) = 2A^\top A\vec{x} - 2(\vec{b}^\top A)^\top + 2\lambda^2\vec{x}$$

Setting the gradient to zero gives us

$$\begin{aligned} (A^\top A + \lambda^2 I)\vec{x}^* &= A^\top\vec{b} \\ \vec{x}^* &= (A^\top A + \lambda^2 I)^{-1}A^\top\vec{b} \end{aligned}$$

Ridge regression has two interpretations.

- We want to shift the eigenvalues of A to limit the condition number so it is not too large.
- Without the regularizer, the predicted coefficient of the polynomial tend to be really large (10^6-level large). The regularizer integrated the size of x into the minimizing terms and controls the size of the predicted value so that it is not insanely large.

Note: the solution to the ridge regression is **not** the same as the solution to OLS. In general, these two solutions are distinct.

3.d. Tikhonov regularization

Definition 3.5 (Tikhonov regularization)
Consider data $A\vec{x} = \vec{b}$. We decide to add weights W_1 to the data points such that the weights represents the "importance" or "confidence." We then add some new data W_2 to A and a corresponding $\vec{x}_0$ to $\vec{b}$. With the additional information, the original data becomes:

$$W_1 \begin{bmatrix} A \\ W_2 \end{bmatrix} \vec{x} = \begin{bmatrix} \vec{b} \\ \vec{x}_0 \end{bmatrix}$$

where W_1 and W_2 are matrices. The optimization problem becomes:

$$\min_{\vec{x}} \|W_1(A\vec{x} - \vec{b})\|_2^2 + \|W_2(\vec{x} - \vec{x}_0)\|_2^2$$

Such problem is called Tikhonov regression.

3.e. Probablistic perspective

Definition 3.6 (Problem statement)
Consider model

$$y_i = g(x_i) + z_i$$

Where z_i is noise. We have some information about the noise such that

$$z_i \sim N(0, \sigma_i^2) \to f(z_i) = \frac{e^{-z_i^2/2\sigma_i^2}}{\sqrt{2\pi}\sigma_i}$$

This model is our data points. **Assume** the model is linear, i.e. $g(\vec{x}_i) = \vec{x}_i^\top \vec{w}$. In this context, we can call $\vec{w}$ as our "model". We can rewrite the original equation to

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \cdots & \vec{x}_1^\top & \cdots \\ & \vdots & \\ \cdots & \vec{x}_n^\top & \cdots \end{bmatrix} \vec{w} + \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}$$

such that $\vec{y} \approx X\vec{w}$. We could solve this problem by OLS, but OLS does not count into consideration the information we know about the noise and thus gives suboptimal solution. Is there a better way to choose $\vec{w}$?

Theorem 3.7 (Maximum Likelihood estimation)

Goal: find $\vec{w}$ that makes observed data most likely, i.e.

$$\underset{\vec{w}_0}{\operatorname{argmax}}\, f(Y_1 = y_1, \cdots, Y_n = y_n | \vec{w} = \vec{w}_0)$$

Assume z_i i.i.d. Then we can rewrite the original problem into

$$\underset{\vec{w}_0}{\operatorname{argmax}} \prod_{i=1}^{n} f(Y_i = y_i | \vec{w} = \vec{w}_0)$$

Note that

$$\begin{aligned} f(Y_i = y_i | \vec{w} = \vec{w}_0) &= f(\vec{x}_i^\top \vec{w}_0 + z_i = y_i | \vec{w} = \vec{w}_0 \\ &= f(z_i = y_i - \vec{x}_i^\top \vec{w}_0 | \vec{w} = \vec{w}_0) \\ &= \frac{e^{\frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2}}}{\sqrt{2\pi}\sigma_i} \end{aligned}$$

Therefore

$$\begin{aligned} \underset{\vec{w}_0}{\operatorname{argmax}} \prod_{i=1}^{n} f(Y_i = y_i | \vec{w} = \vec{w}_0) &= \underset{\vec{w}_0}{\operatorname{argmax}} \prod_{i=1}^{n} \frac{e^{\frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2}}}{\sqrt{2\pi}\sigma_i} \\ &= \underset{\vec{w}_0}{\operatorname{argmax}} \frac{1}{(\sqrt{2\pi})^n \prod_{i=1}^{n} \sigma_i} \prod_{i=1}^{n} e^{\frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2}} \\ &= \underset{\vec{w}_0}{\operatorname{argmax}} \frac{1}{(\sqrt{2\pi})^n \prod_{i=1}^{n} \sigma_i} \exp\left\{ - \sum_{i=1}^{n} \frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2} \right\} \\ &= \underset{\vec{w}_0}{\operatorname{argmax}} - \sum_{i=1}^{n} \frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2} \\ &= \underset{\vec{w}_0}{\operatorname{argmin}} \sum_{i=1}^{n} \frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2} \\ &= \underset{\vec{w}_0}{\operatorname{argmin}} \| S(X\vec{w}_0 - \vec{y}) \|_2^2 \end{aligned}$$

Where

$$S = \begin{bmatrix} \sqrt{\frac{1}{2\sigma_1^2}} & & \\ & \ddots & \\ & & \sqrt{\frac{1}{2\sigma_n^2}} \end{bmatrix}$$

Theorem 3.8 (Maximum a posteriori estimation (MAP))

Based on the problem stated in MLE, what if we have a prior on $\vec{w}$? Again, we have

$$y_i = g(x_i) + z_i$$

$$z_i \sim N(0, \sigma_i^2) \to f(z_i) = \frac{e^{-z_i^2/2\sigma_i^2}}{\sqrt{2\pi}\sigma_i}$$

In addition,

$$w_i \sim N(\mu_i, \rho_i^2)$$

i.e.

$$\vec{w} \sim N(\vec{\mu}, \Sigma_{\vec{w}}) \;\; s.t. \;\; \Sigma_{\vec{w}} = \begin{bmatrix} \rho_1^2 & & \\ & \ddots & \\ & & \rho_n^2 \end{bmatrix}$$

Goal: find the most likely $\vec{w}$ given data $y_1, \cdots, y_n$, i.e.

$$\underset{\vec{w}}{\operatorname{argmax}} f(\vec{w}|\vec{Y} = \vec{y})$$

By the Bayes theorem,

$$f(\vec{w}|\vec{Y} = \vec{y}) = \frac{f(\vec{Y} = \vec{y}|\vec{w}) f\vec{w}}{f\vec{Y}}$$

Hence

$$\begin{aligned}
\underset{\vec{w}}{\operatorname{argmax}} f(\vec{w}|\vec{Y} = \vec{y}) &= \underset{\vec{w}}{\operatorname{argmax}} f(\vec{Y} = \vec{y}|\vec{w}) f(\vec{w}) \\
&= \underset{\vec{w}}{\operatorname{argmax}} \left(\prod_{i=1}^{n} f(Y = y_i|\vec{w}) \right) f(\vec{w})
\end{aligned}$$

Borrowing the calculation we did in MLE,

$$\begin{aligned}
\underset{\vec{w}}{\operatorname{argmax}} f(\vec{w}|\vec{Y} = \vec{y}) &= \underset{\vec{w}}{\operatorname{argmax}} \prod_{i=1}^{n} \frac{\exp\left\{\frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2}\right\}}{\sqrt{2\pi}\sigma_i} \frac{\exp\left\{-(\vec{w} - \vec{\mu})^\top \Sigma_W^{-1} (\vec{w} - \vec{\mu})\right\}}{(\sqrt{2\pi})^n (\prod \rho_i)} \\
&= \underset{\vec{w}}{\operatorname{argmax}} \exp\left\{ \sum_{i=1}^{n} \frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2} - (\vec{w} - \vec{\mu})^\top \Sigma_W^{-1} (\vec{w} - \vec{\mu}) \right\} \\
&= \underset{\vec{w}}{\operatorname{argmax}} \sum_{i=1}^{n} \frac{-(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2} - (\vec{w} - \vec{\mu})^\top \Sigma_W^{-1} (\vec{w} - \vec{\mu}) \\
&= \underset{\vec{w}}{\operatorname{argmin}} \sum_{i=1}^{n} \frac{(y_i - \vec{x}_i^\top \vec{w}_0)^2}{2\sigma_i^2} + (\vec{w} - \vec{\mu})^\top \Sigma_W^{-1} (\vec{w} - \vec{\mu}) \\
&= \underset{\vec{w}}{\operatorname{argmin}} \|S(X\vec{w}_0 - \vec{y})\|_2^2 + \|\sqrt{\Sigma_W^{-1}} (\vec{w} - \vec{\mu})\|_2^2
\end{aligned}$$

For example, if some ρ's are large (note that ρ's are the variances of the w's), you do not need to care too much about keeping w and μ close in their values. But if ρ's are small, than differences in values of w and μ are going to have a large impact (Therefore you should put a high weight on keeping w and μ similar).

4. Convexity

4.a. Convex Sets

Definition 4.1 (convex combination)
Consider $\vec{x}_i$,

$$\sum_{i=1}^{n} \lambda_i \vec{x}$$

is a convex combination of $\vec{x}$ if

$$\lambda_i \geq 0 \text{ and } \sum_{i=1}^{n} \lambda_i = 1$$

Definition 4.2 (convex set)
A set C is convex if the line segment joining any two points in the set is contained in the set.

Example 4.3
Consider C a vector space. If C is convex then

$$\theta \vec{x}_i + (1 - \theta)\vec{x}_2 \in C \quad \forall \theta$$

if $\vec{x}_1, \vec{x}_2 \in C$ and $\theta \in [0, 1]$.

Example 4.4
Let

$$C = \{\vec{x} \mid \vec{a}^\top \vec{x} = b\}$$

Note that C is a hyperplane. It can be rewritten into

$$\vec{a}(\vec{x} - \vec{x}_0) = 0$$
$$\vec{a}^\top \vec{x} = \vec{a}^\top \vec{x}_0 = b$$

To check wheter C is convex, consider $\vec{x}_1, \vec{x}_2 \in C$ and let

$$\vec{x}_3 = \theta \vec{x}_1 + (1 - \theta)\vec{x}_2$$

We know that

$$\vec{a}^\top \vec{x}_3 = \theta \vec{a}^\top \vec{x}_1 + (1 - \theta)\vec{a}^\top \vec{x}_2 = b$$

Therefore $\vec{x}_3$ belongs to C and C is convex.

Remark 4.5

A hyperplane (a plane which's dimension is 1 less than the dimension of its ambient space) divides the space into two half spaces. The set

$$\{\vec{x} \mid \vec{a}^\top \vec{x} \geq b\}$$

defines a hyperplane, where $\vec{a}$ is perpendicular to all vectors on this plane. This hyperplane naturally generates a counter part

$$\{\vec{x} \mid \vec{a}^\top \vec{x} \leq b\}$$

Example:

$$P = \{\vec{x} \mid \vec{a}^\top(\vec{x} - \vec{x}_0) \geq 0\} \;\; N = \{\vec{x} \mid \vec{a}^\top(\vec{x} - \vec{x}_0) \leq 0\}$$

devides the space into two parts (P for positive and N for negative).

Example 4.6

Consider

$$P = \{A \mid A \in \mathbb{S}^n, \;\; A \text{ is PSD}\}$$

Recall that A is PSD iff

$$\vec{x}^\top A \vec{x} \geq 0 \;\; \forall \vec{x} \in \mathbb{R}^n$$

Is P convex? Let

$$A_1, \; A_2 \in P \;\; and \;\; A_3 = \theta A_1 + (1 - \theta) A_2$$

Then

$$\begin{aligned} \vec{x}^\top A_3 \vec{x} = \theta(\vec{x}^\top A_1 \vec{x}) + (1 - \theta)\vec{x}^\top A_2 \vec{x} \geq 0 \\ \implies A_3 \in P \end{aligned}$$

Therefore P is convex.

Remark 4.7

Linear transformations always preserve convexity.

Theorem 4.8 (separating hyperplane theorem)

Let C, D be convex sets and $C \cap D = \varnothing$. Then there exists hyperplane $\vec{a}^\top \vec{x} = b$ separating two sets such that

$$\begin{aligned} \forall \vec{x} \in C \;\; \vec{a}^\top \vec{x} \geq b \\ \forall \vec{x} \in D \;\; \vec{a}^\top \vec{x} \leq b \end{aligned}$$

Proof: TODO

4.b. Convex Functions

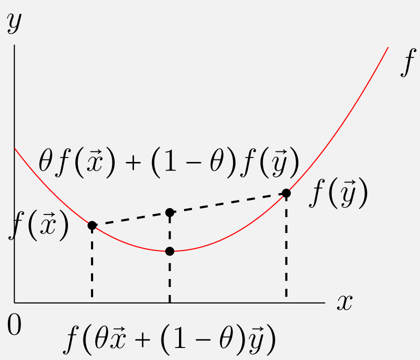
Definition 4.9 (convex functions)
Let

$$f : \mathbb{R}^n \to \mathbb{R}$$

Function f is convex if the domain of f is a convex set and

$$f(\theta\vec{x} + (1-\theta)\vec{y}) \le \theta f(\vec{x}) + (1-\theta)f(\vec{y}) \quad 0 \le \theta \le 1$$

The above inequality is called **Jensen's Inequality**. Here is an example of a convex function that visualizes the Jensen's Inequality.



If the "cord" is always above the function, the function is **convex**. If the "cord" is always below the function, the function is **concave**.

Theorem 4.10
If a function f is convex, any local minimum is the global minimum.

Definition 4.11 (Epigraph)
The epigraph of a function f is defined as

$$Epi f = \{(x,t) \mid x \in dom f \;\; f(x) \le t\}$$

f is a convex function $\iff$ Epi f is a convex set.

Theorem 4.12 (First-order condition)
Define $f : \mathbb{R}^n \to \mathbb{R}$ a differentiable function. Then f is convex iff

$$f(\vec{y}) \ge f(\vec{x}) + \nabla f(\vec{x})^\top(\vec{y} - \vec{x}) \;\; \forall \vec{x}, \vec{y} \in dom f \;\; 0 \le \theta \le 1$$

Remark 4.13 (Implication of the FOC)
If $\nabla f(\vec{x}_*) = 0$ and f is convex, then

$$f(\vec{y}) \geq f(\vec{x}) + 0(\vec{y} - \vec{x})$$
$$f(\vec{y}) \geq f(\vec{x})$$

For all y in the domain, which means that $\vec{x}_*$ is a global minimum!!

Theorem 4.14 (Second-order condition)
Let $f : \mathbb{R}^n \to \mathbb{R}$ who's domain is convex and is twice-differentiable. f is convex iff

$$\nabla^2 f(\vec{x}) \geq 0$$

In another word, $\nabla^2 f(\vec{x})$ is positive semi-definite.

Definition 4.15 (Strict Convexity)
Dom f convex. For all x y in domain, f is strictly convex iff

$$f(\theta\vec{x} + (1-\theta)\vec{y}) < \theta f(\vec{x}) + (1-\theta) f(\vec{y})$$

FOC:

$$f(\vec{y}) > f(\vec{x}) + \nabla f(\vec{x})^\top (\vec{y} - \vec{x}) \ \ \forall \vec{x}, \vec{y} \in dom f \ \ 0 < \theta < 1$$

SOC:

$$\nabla^2 f(\vec{x}) \succ 0$$

Remark 4.16
If f is a stright line, f is both convex and concave, but not strictly convex.

Definition 4.17 (Strong Convexity)
Dom f convex. For all x y in domain, f is μ-strongly convex iff

$$f(\vec{y}) \geq f(\vec{x}) + \nabla f(\vec{x})^\top (\vec{y} - \vec{x}) + \frac{\mu}{2} \|\vec{y} - \vec{x}\|^2$$

Remark 4.18 (implication of strong convexity)
Recall that by Taylor's theorem, for $f(\vec{x}) := \mathbb{R}^n \to \mathbb{R}$, the derivative of f is

$$f(\vec{y}) \approx f(\vec{x}) + \nabla f^\top (\vec{y} - \vec{x}) + \frac{1}{2}(\vec{y} - \vec{x})^\top \nabla^2 f(\vec{y} - \vec{x})$$

If we let $\mu I = \nabla^2 f$, we have

$$\frac{\mu}{2}\|\vec{y} - \vec{x}\|^2 = \frac{1}{2}(\vec{y} - \vec{x})^\top \mu I(\vec{y} - \vec{x})$$

Thus the implication of strong convexity is that the hessian of f is at least μI.

Remark 4.19
Strong convexity $\implies$ strict convexity $\implies$ convexity

Remark 4.20
For matrices A and B,

$$A \geq B \implies A - B \geq 0$$

5. Gradient Descent

5.a. Introduction to Gradient Descent

Definition 5.1 (Gradient Descent)

Gradient Descent is an approach to unconstrained optimization problems. The basic idea is to nudge the function in the right direction by a little bit in every step, and after a lot of steps the function will arrive at a local minimum. Formally, for a step size s and a direction $\vec{v}$,

$$f(\vec{x} + s\vec{v}) \approx f(\vec{x}) + s < \nabla f(\vec{x}), \vec{v} >$$

Recall Cauchy-Schwartz, the magnitude of $< \nabla f(\vec{x}), \vec{v} >$ is maximized if $\vec{v}$ is aligned with $\nabla f(\vec{x})$. We want to minimize the inner product while maximize its magnitude so the function steps towards the minimum at the fastest rate, hence we choose

$$\vec{v} = -\nabla f(\vec{x})$$

The formal algorithm for gradiant descent is defined as follows. Let $\vec{x}$ be the parameter of function f. At step k,

$$\vec{x}_{k+1} = \vec{x}_k - \eta \nabla f(\vec{x}_k)$$

Where $\vec{x}_0$ is the initial point and η is the stepsize.

Example 5.2 (GD on LS)

Let $f(\vec{x}) = \|A\vec{x} - \vec{b}\|_2^2$. It has a direct solution of $\vec{x}^* = (A^\top A)^{-1} A^\top \vec{b}$. If A is a n*n matrix, the runtime of computing the direct solution is at least $O(n^3)$ (taking a matrix inverse is approx. $O(n^3)$). It is computationally cheaper to use gradient descent. Thus,

$$\nabla f(\vec{x}) = 2A^\top (A\vec{x} - \vec{b})$$

$$\begin{aligned} \vec{x}_{k+1} &= \vec{x}_k - \eta \nabla f(\vec{x}_k) \\ &= \vec{x}_k - \eta 2A^\top (A\vec{x} - \vec{b}) \\ \vec{x}_{k+1} &= (I - 2\eta A^\top A)\vec{x}_k + 2\eta A^\top \vec{b} \end{aligned}$$

Next we need to prove that this algorithm will converge. The following is one of the ways to prove convergence. The difference between optimal value and the k-step value is

$$\begin{aligned} \vec{x}_{k+1} - (A^\top A)^{-1} A^\top \vec{b} &= (I - 2\eta A^\top A)\vec{x}_k + 2\eta A^\top \vec{b} - (A^\top A)^{-1} A^\top \vec{b} \\ &= (I - 2\eta A^\top A)\vec{x}_k + 2\eta (A^\top A)(A^\top A)^{-1} A^\top \vec{b} - (A^\top A)^{-1} A^\top \vec{b} \\ &= (I - 2\eta A^\top A)\vec{x}_k + (2\eta A^\top A - I)(A^\top A)^{-1} A^\top \vec{b} \\ &= (I - 2\eta A^\top A)(\vec{x}_k - (A^\top A)^{-1} A^\top \vec{b}) \end{aligned}$$

Hence if the absolute values of the eigenvalues of $I - 2\eta A^\top A$ are strictly less than 1, GD converges for LS.

Example 5.3 (GD on LS continued)
Since we showed in the previous part that

$$\vec{x}_{k+1} - (A^\top A)^{-1}A^\top\vec{b} = (I - 2\eta A^\top A)^{k+1}(\vec{x}_k - (A^\top A)^{-1}A^\top\vec{b})$$

Where η (step size) is a parameter of choice and we want to make sure the absolute values of the eigenvalues of $(I - 2\eta A^\top A)$ is strictly less than 1, we should choose an appropriate η such that the algorithm converges.

5.b. Gradient Descent for μ-strongly convex L-smooth functions

Definition 5.4 (Bounds of convex functions)
Recall the definition of μ-strongly convex: Dom f convex. For all x y in domain, f is μ-strongly convex iff

$$f(\vec{y}) \geq f(\vec{x}) + \nabla f(\vec{x})^\top(\vec{y} - \vec{x}) + \frac{\mu}{2}\|\vec{y} - \vec{x}\|^2$$

Intuitively, this means that there exists a quadratic function under f such that this quadratic function is the lower-bound of f, hence the gradient of f is changing fast enough. On the other hand, **L-smooth** means that there exists a quadratic function such that f is upper-bounded by this function, hence the gradient of f is not changing too fast. Formally,

$$f(\vec{y}) \leq f(\vec{x}) + \nabla f(\vec{x})^\top(\vec{y} - \vec{x}) + \frac{L}{2}\|\vec{y} - \vec{x}\|_2^2$$

Theorem 5.5
Let $f : \mathbb{R}^n \to \mathbb{R}$ and define optimization problem

$$\min_{\vec{x}\in\mathbb{R}^n} f(\vec{x})$$

Let $\vec{x}_*$ be the optimal solution to the above problem. Then for GD approach

$$\vec{x}_{t+1} = \vec{x}_t - \eta\nabla f(\vec{x}_t)$$

I can choose an η such that

$$\|\vec{x}_{t+1} - \vec{x}_*\|_2^2 \leq (C)^{t+1}\|\vec{x}_0 - \vec{x}_*\|_2^2$$

Lemma 5.6
Let f L-smooth, then

$$\|\nabla f(\vec{x})\|_2^2 \leq 2L(f(\vec{x}) - f(\vec{x}_*))$$

Proof. Since $f(\vec{x}_*)$ minimum, we have

$$f(\vec{x}_*) \le f(\vec{x})$$

And

$$f(\vec{x}_*) \le f(\vec{x} - \frac{\nabla f(\vec{x})}{L})$$

Recall the definition of L-smooth:

$$f(\vec{y}) \le f(\vec{x}) + \nabla f(\vec{x})^\top (\vec{y} - \vec{x}) + \frac{L}{2}\|\vec{y} - \vec{x}\|_2^2$$

We choose $f(\vec{y}) = f(\vec{x} - \frac{\nabla f(\vec{x})}{L})$, then

$$\begin{aligned} f(\vec{x} - \frac{\nabla f(\vec{x})}{L}) &\le f(\vec{x}) + \nabla f(\vec{x})^\top (-\frac{\nabla f(\vec{x})}{L}) + \frac{L}{2}\| - \frac{\nabla f(\vec{x})}{L}\|_2^2 \\ &\le f(\vec{x}) - \frac{1}{L}\|\nabla f(\vec{x})\|^2 + \frac{1}{2L}\|\nabla f(\vec{x})\|_2^2 \\ &\le f(\vec{x}) + \frac{1}{2L}\|\nabla f(\vec{x})\|_2^2 \end{aligned}$$

Since $f(\vec{x}_*) \le f(\vec{x} - \frac{\nabla f(\vec{x})}{L})$,

$$f(\vec{x}_*) \le f(\vec{x}) + \frac{1}{2L}\|\nabla f(\vec{x})\|_2^2$$

■

Lemma 5.7

If f μ-strongly convex,

$$\nabla f(\vec{x})^\top (\vec{x}_* - \vec{x}) \ge f(\vec{x}) - f(\vec{x}_*) + \frac{\mu}{2}\|\vec{x}_* - \vec{x}\|_2^2$$

Proof. Recall the definition of μ-strong convexity: Dom f convex. For all x y in domain, f is μ-strongly convex iff

$$f(\vec{y}) \ge f(\vec{x}) + \nabla f(\vec{x})^\top (\vec{y} - \vec{x}) + \frac{\mu}{2}\|\vec{y} - \vec{x}\|^2$$

Let $\vec{y} = \vec{x}_*$. We have

$$\begin{aligned} f(\vec{x}_*) &\ge f(\vec{x}) + \nabla f(\vec{x})^\top (\vec{x}_* - \vec{x}) + \frac{\mu}{2}\|\vec{x}_* - \vec{x}\|^2 \\ f(\vec{x}_*) - f(\vec{x}) - \frac{\mu}{2}\|\vec{x}_* - \vec{x}\|_2^2 &\ge \nabla f(\vec{x})^\top (\vec{x}_* - \vec{x}) \\ -f(\vec{x}_*) + f(\vec{x}) + \frac{\mu}{2}\|\vec{x}_* - \vec{x}\|_2^2 &\le \nabla f(\vec{x})^\top (\vec{x}_* - \vec{x}) \end{aligned}$$

■

Proof of Main Theorem (Theorem 5.5): TODO

5.c. Introduction to Stochastic Gradient Descent

Definition 5.8

Let $f : \mathbb{R}^n \to \mathbb{R}$ and define optimization problem

$$\min_{\vec{x} \in \mathbb{R}^n} f(\vec{x})$$

Assume the function f is of the following form:

$$f(\vec{x}) = \sum_{i=1}^{m} \frac{1}{m} f_i(\vec{x})$$

E.g. for least squares problem

$$\frac{1}{2m} \|A\vec{x} - \vec{b}\|_2^2 = \frac{1}{2m} \sum_{i=1}^{m} (\vec{a}_i^\top \vec{x} - \vec{b}_i)^2$$

For

$$A = \begin{bmatrix} \cdots & \vec{a}_1^\top & \cdots \\ & \vdots & \\ \cdots & \vec{a}_n^\top & \cdots \end{bmatrix}$$

To calculate the solution of such problem using gradient descent would cost too much computation time (order n time complexity to calculate n losses). Thus instead of computing all of the losses, we choose one component of the gradient in each step.

$$\vec{x}_{k+1} = \vec{x}_k - \eta_k \nabla f_i(\vec{x}_k)$$

This works because

$$\begin{aligned} \mathbb{E}[\nabla f_i(\vec{x}_k)] &= \frac{1}{m} \sum_{i=1}^{m} f_i(\vec{x}) \\ &= \nabla f(\vec{x}_k) \end{aligned}$$

We usually let η be time-dependent.

Example 5.9

Consider optimization problem

$$f(\vec{x}) = \frac{1}{m}\sum_{i=1}^{m} \|\vec{x} - \vec{p}_i\|_2^2$$

We use SGD to solve this problem. Let step size $\eta = \frac{1}{t}$ and $\vec{x}_0 = 0$, then

$$\nabla f(\vec{x}) = \frac{1}{2}2(\vec{x} - \vec{p}_i) = \vec{x} - \vec{p}_i$$

The $\vec{x}$ in each step goes like

$$\begin{aligned}
\vec{x}_0 &= 0 \\
\vec{x}_1 &= \vec{x}_0 - \frac{1}{1}(\vec{x}_0 - \vec{p}_1) = \vec{p}_1 \\
\vec{x}_2 &= \vec{x}_1 - \frac{1}{2}(\vec{x}_1 - \vec{p}_2) = \frac{\vec{p}_1 + \vec{p}_2}{2} \\
\vec{x}_3 &= \vec{x}_2 - \frac{1}{3}(\vec{x}_2 - \vec{p}_3) = \frac{\vec{p}_1 + \vec{p}_2 + \vec{p}_3}{3}
\end{aligned}$$

The future terms of GD can be projected to

$$\vec{x}_i = \frac{\sum_{j=1}^{i} p_j}{i}$$

5.d. Projected Gradient Descent

Definition 5.10 (projected gradient descent)

Differ from ordinary gradient descent, projected gradient descent solves for the general GD problem where $\vec{x}$ is constrained. Formally, we want to solve the problem

$$\min_{\vec{x}} f(\vec{x}) \ , \ \ \vec{x} \in C$$

Such that

$$\vec{x}_{k+1} = \prod_{C} \vec{x}_k - \eta \nabla f(\vec{x}_k)$$

And

$$\prod_{C}(\vec{y}) = \operatorname*{argmin}_{\vec{\delta} \in C} \|\vec{y}_{k+1} - \vec{\delta}\|_2^2$$

However, this could still be computationally expensive, therefore is replaced by Conditional Gradient Descent in the next subsection.

5.e. Conditional Gradient Descent

Definition 5.11 (conditional gradient descent/Frank–Wolfe algorithm)
Let γ_k be a predetermined sequence. Define

$$\vec{y}_k = \operatorname*{argmin}_{\vec{y} \in C} \nabla f(\vec{x}_k)^\top \vec{y}$$

And

$$\begin{aligned} \vec{x}_{k+1} &= (1 - \gamma_k)\vec{x}_k + \gamma_k \vec{y}_k \\ &= \vec{x}_k + \gamma_k(\vec{y}_k - \vec{x}_k) \end{aligned}$$

Both Frank-Wolfe and PGD are generally computationally expensive, but Frank-Wolfe is sometimes cheaper. Frank-Wolfe also has a nice sparse property but it is out of scope for this class.

6. Duality

6.a. Weak Duality

Remark 6.1

Gradient Descent works for unconstrained optimization problem on convex functions. It has challenges that the function should be differentiable and convex, and at the same time GD is computationally expensive.

Duality is a technique that transforms every problem to a convex form (the dual form). It does not necessarily give us the solution to the original problem, but it will always give a bound.

Theorem 6.2 (Lagrangian)

Optimization problem

$$p^* = \min f_0(\vec{x})$$

Under the constrains

$$f_i(\vec{x}) \le 0, \quad \forall i \;\; 1 \le i \le m$$
$$h_i(\vec{x}) = 0, \quad \forall i \;\; 1 \le i \le p$$

The problem defined above is called a **primal problem**. Define **Lagrangian**:

$$L(\vec{x}, \vec{\lambda}, \vec{\nu}) = f_0(\vec{x}) + \sum_{i=1}^{m} \lambda_i f_i(\vec{x}) + \sum_{i=1}^{p} \nu_i h_i(\vec{x}) \quad \forall \lambda_i \ge 0$$

Note that it is an **affine** function of λ and ν, which means that it is **both convex and concave**. Now, define new problem

$$\min_{\lambda \ge 0} L(\vec{x}, \vec{\lambda}, \vec{\nu}) := g(\vec{\lambda}, \vec{\nu})$$

Over all $\vec{x}$. Examine g. Properties of g include

1. g is a function of only $\vec{\lambda}, \vec{\nu}$.
2. L is an affine function of $\vec{\lambda}, \vec{\nu}$.
3. By 1 and 2, g is a concave function of $\vec{\lambda}, \vec{\nu}$.
4. It turns out, **g($\vec{\lambda}, \vec{\nu}$) is a lower bound on the primal optimal** p^*.

Proof of property No. 4: TODO

Example 6.3 (Duality on LS)

Let $A \in \mathbb{R}^{m*n}$, $m < n$. Problem

$$\min \vec{x}^\top \vec{x} = \vec{p}^*$$

Under the constrain

$$A\vec{x} = \vec{b}$$

Since there is no inequality constains, the lagrangian of this problem does not have λ.

$$L(\vec{x}, \vec{\nu}) = \vec{x}^\top \vec{x} + \nu^\top (A\vec{x} - \vec{b})$$

$$g(\vec{\nu}) = \min_{\vec{x}} L(\vec{x}, \vec{\nu})$$

To solve for g, we do

$$\nabla_{\vec{x}} L(\vec{x}, \vec{\nu}) = 2\vec{x} + A^\top \vec{\nu}$$

Set gradient to zero,

$$\vec{x} = -\frac{1}{2} A^\top \vec{\nu}$$

Is the point where L is minimized. g at this point is

$$\begin{aligned} g(\vec{\nu}) = L(-\frac{1}{2}A^\top \vec{\nu}, \vec{\nu}) &= (-\frac{1}{2}A^\top \vec{\nu})^\top (-\frac{1}{2}A^\top \vec{\nu}) + \vec{\nu}^\top (-\frac{1}{2}AA^\top \vec{\nu} - \vec{b}) \\ &= \frac{1}{4}\vec{\nu}^\top AA^\top \vec{\nu} + \vec{\nu}^\top (-AA^\top \vec{\nu} - \vec{b}) \\ &= -\frac{1}{4}\vec{\nu}^\top AA^\top \vec{\nu} - \vec{\nu}^\top \vec{b} \end{aligned}$$

Which is the lower bound of p^*. What is the max lower bound? In another word, we want to find

$$\max_{\vec{\nu}} g(\vec{\nu})$$

Since g concave, we take its gradient and set to zero:

$$\nabla_{\vec{\nu}} g(\vec{\nu}) = -\frac{1}{4} 2AA^\top \vec{\nu} - \vec{b} = 0$$

$$\vec{\nu}^* = -2(AA^\top)^{-1}\vec{b}$$

Thus

$$\vec{x}^* = -\frac{1}{2}A^\top (-2(AA^\top)^{-1}\vec{b}) = A^\top (AA^\top)^{-1}\vec{b}$$

Definition 6.4 (Dual)
Continuing the definition of lagrangian, we want to find the tightest lower bound of p^*, formally

$$\max_{\vec{\lambda} \geq 0} g(\vec{\lambda}, \vec{\nu}) = d^*$$

Over $\vec{\nu}$, this problem is the **DUAL Problem**. It is a maximization problem of a concave function under linear constrains, thus a **convex program**.

Remark 6.5
Properties of the Dual problem:

1. # of variables = # of constraints of the primal.
2. **Always convex problem even if primal is not! :)**

By 1, if the data is large but the constaints are few, the Dual problem is an easier problem to solve.

Definition 6.6 (Weak Duality)
Continuing from the definition of lagrangian and duality, if

$$d^* \leq p^*$$

Then it is **WEAK DUALITY**.

6.b. Strong Duality

Definition 6.7 (Strong Duality)
If

$$d^* = p^*$$

Like we seen in the Duality on LS example, then it is **STRONG DUALITY**.

Remark 6.8
Strong Duality $\implies$ Weak Duality. **In general, weak duality always holds**, while strong duality only holds under certain circumstances.

Definition 6.9 (duality gap)

$$p^* - d^*$$

is called the duality gap. Duality gap = 0 iff strong duality.

Theorem 6.10 (Minmax Inequality)
Sets X, Y. F is any function. Then,

$$\min_{x\in X}\max_{y\in Y} F(x,y) \geq \max_{y\in Y}\min_{x\in X} F(x,y)$$

Proof of Minmax Inequality:

Proof. Fix $x_0 \in X$, $y_0 \in Y$. Define

$$h(y_0) := \min_{x\in X} F(x, y_0)$$

and

$$g(x_0) := \max_{y\in Y} F(x_0, y)$$

See that

$$h(y_0) = \min_{x\in X} F(x,y_0) \leq F(x_0,y_0) \leq \max_{y\in Y} F(x_0,y) = g(x_0)$$

Thus

$$\begin{aligned} h(y_0) &\leq g(x_0) \\ \max_{y\in Y} h(y_0) &\leq \min_{x\in X} g(x_0) \\ \max_{y\in Y}\min_{x\in X} F(x,y) &\leq \min_{x\in X}\max_{y\in Y} F(x,y) \end{aligned}$$

■

Remark 6.11 (Implication of Minmax Inequality on Duality)
By definition,

$$\begin{aligned} d^* &= \max_{\vec{\lambda}\geq 0} g(\vec{\lambda},\vec{\nu}) \\ &= \max_{\vec{\lambda}\geq 0}\min_{\vec{x}} L(\vec{x},\vec{\lambda},\vec{\nu}) \end{aligned}$$

How can we connect this to the primal? Consider

$$\max_{\vec{\lambda}\geq 0,\,\vec{\nu}} \left(f_0(\vec{x}) + \sum_{i=1}^{m} \lambda_i f_i(\vec{x}) + \sum_{i=1}^{p} \nu_i h_i(\vec{x}) \right)$$
$$= \begin{cases} f_0(\vec{x}) & \text{if } \vec{x} \text{ is feasible} \\ \infty & \text{if } \vec{x} \text{ is infeasible} \end{cases}$$

Therefore we can write

$$p^* = \min_{\vec{x}} \max \vec{\lambda} \geq 0,\ \vec{\nu} L(\vec{x},\vec{\lambda},\vec{\nu})$$

By the minmax inequality,

$$p^* \geq d^*$$

Theorem 6.12 (Slater's Condition)
For a convex problem, strong duality holds if

$$\exists \vec{x}_0 \text{ such that } f_i(\vec{x}_0) < 0 \ \ \forall i$$

In another word, the point $\vec{x}_0$ is strictly feasible. Or,

$$\vec{x}_0 \in \text{ RelativeInterior}(\mathbf{D})$$

For the purpose of this class we can think of RelativeInterior as the Interior.

Theorem 6.13 (Refined Slater's Condition)
Convex problem, $f_1, f_2, \ldots, f_k$ that are affine,

$$\exists \vec{x}_0 \text{ such that } f_i(\vec{x}_0) \leq 0 \ \ \forall i = 1, 2, \cdots, k \text{ AND } f_i(\vec{x}_0) < 0 \ \ \forall i = k+1, \cdots, m$$

Assume that first k constraints are affine, and the rest are whatever non-linear things that you want them to be. You are allowed to have equality for the affine things, as long as the non-affine things all satisfy the above strict inequality. So you don't have to find a point that satisfies strict inequalities on the affine things too. Sometimes it is easier to find such point.

Remark 6.14
As long as the problem is feasible, strong duality will always hold for linear program.

6.c. Partitioning Problem

Definition 6.15 (partitioning problem)

$$\min_{\vec{x}_i,\, \vec{x}_i^2=1} \vec{x}^\top W \vec{x}, \;\; W \in \mathbb{S}^n$$

Remark 6.16

Partitioning problem is not a convex problem. This is because that $\vec{x}_i$ can only be either +1 or -1, thus the domain is descrete.

The lagrangian for the partitioning problem is

$$\begin{aligned} L(\vec{x},\vec{\nu}) &= \vec{x}^\top W \vec{x} + \sum_{i=1}^n \vec{\nu}_i(\vec{x}_i^2 - 1) \\ &= \vec{x}^\top W \vec{x} + \vec{x}^\top diag(\vec{\nu})\vec{x} - \sum_{i=1}^n \vec{\nu}_i \\ &= \vec{x}^\top (W + diag(\vec{\nu}))\vec{x} - \sum_{i=1}^n \vec{\nu}_i \end{aligned}$$

If $(W + diag(\vec{\nu}))$ is PSD, the problem is convex. If it is NSD, the problem is concave. The g for the partitioning problem is

$$\begin{aligned} g(\vec{\nu}) &= \min_{\vec{x}} L(\vec{x},\vec{\nu}) \\ &= \begin{cases} -\sum_{i=1}^n \vec{\nu}_i & \text{if } (W + diag(\vec{\nu})) \text{ PSD} \\ -\infty & \text{otherwise} \end{cases} \end{aligned}$$

We see that this problem is trivial if $(W + diag(\vec{\nu}))$ is not PSD. We can rewrite this problem as its dual:

$$\max \sum_{i=1}^n \vec{\nu}_i, \;\; \text{subject to} \;\; (W + diag(\vec{\nu})) \geq 0$$

This is a special kind of problem called **Semi-definite Program**. It is a convex problem that can be efficiently solved. Particularly, for this problem, we choose

$$\vec{\nu} = \lambda_{min}(W)$$

and we have the lower bound

$$p^* \geq n\lambda_{min}(W)$$

This is an example of transforming a very hard-to-solve problem into a convex problem that we can efficiently solve.

6.d. LP and Duality

Example 6.17

Consider linear program

$$\min_{A\vec{x}\leq\vec{b}} \vec{c}^\top \vec{x}$$

The lagrangian is

$$\begin{aligned} L(\vec{x},\vec{\lambda}) &= \vec{c}^\top \vec{x} + \vec{\lambda}^\top (A\vec{x} \leq \vec{b}) \\ &= (A^\top \vec{\lambda} + \vec{c})^\top \vec{x} - \vec{b}^\top \vec{\lambda} \end{aligned}$$

The g is

$$\begin{aligned} g(\vec{\lambda}) &= \min_{\vec{x}} L(\vec{x},\vec{\lambda}) \\ &= \begin{cases} -\infty & \text{if } A^\top \vec{\lambda} + \vec{c} \neq 0 \\ -\vec{b}^\top \vec{\lambda} & \text{if } A^\top \vec{\lambda} + \vec{c} = 0 \end{cases} \end{aligned}$$

The first case is trivial. In order to get a non-trivial bound, we want to calculate

$$d^* = \max_{\vec{\lambda}\geq 0,\ A^\top \vec{\lambda} + \vec{c} = 0} -\vec{b}^\top \vec{\lambda}$$

This is our dual problem.

Example 6.18 (Winery)

Consider the following business problem. You have 200 kilos of merlot and 300 kilos of shiras. You have two recipes:

- Blend 1: 4 kilos merlot + 1 kilo shiras, sell \$20
- Blend 2: 2 kilos merlot + 3 kilos shiras, sell \$15

You want to make q_1 of blend 1 and q_2 of blend 2. How to optimize our revenue? We set up problem for this

$$\max_{q_1,\, q_2 \geq 0} 20q_1 + 15q_2$$

Subject to

$$4q_1 + 2q_1 \leq 200$$
$$q_2 + 3q_2 \leq 300$$

Let's say we sell the leftover grapes. Let λ_1 be the rate to sell marlot and λ_2 be the rate to sell shiras. Then we write new optimization problem

$$\begin{aligned} &\max_{q_1,\, q_2 \geq 0} 20q_1 + 15q_2 + \lambda_1(200 - 4q_1 + 2q_1) + \lambda_2(300 - q_2 + 3q_2) \\ =& \max_{q_1,\, q_2 \geq 0} (20 - 4\lambda_1 - \lambda_2)q_1 + (15 - 2\lambda_1 - 3\lambda_2)q_2 + 200\lambda_1 + 300\lambda_2 \end{aligned}$$

Note that the first line looks like a lagrangian. From the second line we can see that

- if $(20 - 4\lambda_1 - \lambda_2)$ negative, do not make blend 1. If it is positive, make as much blend 1 as you can.
- if $(15 - 2\lambda_1 - 3\lambda_2)$ negative, do not make blend 2. If it is positive, make as much blend 2 as you can.

The strategy seems clear. But what if we have

$$20 - 4\lambda_1 - \lambda_2 = 0 \text{ AND } 15 - 2\lambda_1 - 3\lambda_2 = 0$$

? Then we do not care whether to sell the grapes or to make grapes into wine. These lambda's are called the **shadow prices** of grapes. What is our minimum revenue under these conditions? It is

$$\min_{\lambda_1,\, \lambda_2 \geq 0} 200\lambda_1 + 300\lambda_2$$

Subject to

$$20 - 4\lambda_1 - \lambda_2 = 0 \text{ AND } 15 - 2\lambda_1 - 3\lambda_2 = 0$$

If plug in the numbers, you can see that the original problem and the shadow price problem are primal and dual of each other. Note that for this particular problem, the feasible region of the dual problem is a point.

6.e. Duality Certificates

Definition 6.19 (certificate)
Let (λ_1, ν_1) be a dual feasible point, meaning that it satisfy the constaints of the dual. Let x_1 be the primal feasible point. Then we know that

$$p^* \geq g(\lambda_1, \nu_1)$$

This implies that

$$f_0(x_1) - p^* \leq f_0(x_1) - g(\lambda_1, \nu_1)$$

This means that if $f_0(x_1) - g(\lambda_1, \nu_1)$ is small, $f_0(x_1) - p^*$ is also small. This could be used as the stopping condition for optimization programs. Another way of writing this is that

$$p^* \in [g(\lambda_1, \nu_1), f_0(x_1)]$$

If the strong duality holds, we would also have

$$d^* \in [g(\lambda_1, \nu_1), f_0(x_1)]$$

Note that if the strong duality does not hold, the above about d^* might not be true due to possibly large duality gap.

6.f. Complementary Slackness

Theorem 6.20 (complementary slackness)
Consider the following situation: the primal optimal $\vec{x}^*$ and dual optimal $\vec{\lambda}^*, \vec{\nu}^*$. Assume strong duality holds, i.e. $p^* = d^*$. We have

$$p^* = f_0(\vec{x}^*) = d^* = g(\vec{\lambda}^*, \vec{\nu}^*)$$

And

$$\begin{aligned} g(\vec{\lambda}^*, \vec{\nu}^*) &= \min_{\vec{x}} \left(f_0(\vec{x}) + \sum_{i=1}^m \lambda_i^* f_i(\vec{x}) + \sum_{i=1}^p \nu_i^* h_i(\vec{x}) \right) \\ &\leq f_0(\vec{x}^*) + \sum_{i=1}^m \lambda_i^* f_i(\vec{x}^*) + \sum_{i=1}^p \nu_i^* h_i(\vec{x}^*) \end{aligned} \tag{1}$$

Since we are considering primal problem with constraints including $f_i(\vec{x}) \leq 0$ and the dual $d^* = \max_{\vec{\lambda} \geq 0} g(\vec{\lambda}, \vec{\nu})$, we know the term $\sum_{i=1}^m \lambda_i^* f_i(\vec{x}^*)$ is at most zero. In addition, the constraints of the primal problem includes $h_i(\vec{x}) = 0$ and thus $\sum_{i=1}^p \nu_i^* h_i(\vec{x}^*)$ is zero. Therefore we could write

$$g(\vec{\lambda}^*, \vec{\nu}^*) \leq f_0(\vec{x}^*) + 0 + 0 = f_0(\vec{x}^*) \tag{2}$$

If the above holds, then both inequalities (1) and (2) must in fact be equalities!

Proof of Complementary Slackness: TODO

6.g. KKT Conditions

Remark 6.21

Karush-Kuhn-Tucker are the three people responsible for these conditions.

Theorem 6.22 (KKT Conditions)

The following enumerated conditions are *necessary*, but not necessarily *sufficient*, conditions to the optimality of primal optimal $\vec{x}^*$ and dual optimal $\vec{\lambda}^*, \vec{\nu}^*$. Assume strong duality holds, i.e. $p^* = d^*$. Convex OR non-convex problem. Both object function and the contraints differentiable. Then, the fact that primal optimal $\vec{x}^*$ and dual optimal $\vec{\lambda}^*, \vec{\nu}^*$ implies

1. $f_i(\vec{x}^*) \leq 0 \ \ \forall i = 1, \cdots, m$
2. $h_i(\vec{x}^*) = 0 \ \ \forall i = 1, \cdots, p$
3. $\lambda_i^* \geq 0 \ \ \forall i = 1, \cdots, m$
4. $\lambda_i^* f_i(\vec{x}^*) = 0 \ \ \forall i = 1, \cdots, m$ (Complementary Slackness)
5. $\nabla f_0(\vec{x}^*) + \sum \lambda_i^* \nabla f_i(\vec{x}^*) + \sum \nu_i^* \nabla h_i(\vec{x}^*) = 0$

The fifth condition means that $\vec{x}^*$ minimizes $L(\vec{x}, \vec{\lambda}^*, \vec{\nu}^*)$, which is implied by the Main Theorem.

Theorem 6.23 (KKT Conditions Part II)

Convex and differentiable problems. Strong duality not necessarily holds. Sufficient conditions. $\tilde{x}, \tilde{\lambda}, \tilde{\nu}$ are points. f's are convex and h's are affine. If the points satisfy:

1. $f_i(\tilde{x}) \leq 0 \ \ \forall i = 1, \cdots, m$
2. $h_i(\tilde{x}) = 0 \ \ \forall i = 1, \cdots, p$
3. $\tilde{\lambda}_i \geq 0 \ \ \forall i = 1, \cdots, m$
4. $\tilde{\lambda}_i f_i(\tilde{x}) = 0 \ \ \forall i = 1, \cdots, m$
5. $\nabla f_0(\tilde{x}) + \sum \tilde{\lambda}_i \nabla f_i(\tilde{x}) + \sum \tilde{\nu}_i \nabla h_i(\tilde{x}) = 0$

Then $\tilde{x}, \tilde{\lambda}, \tilde{\nu}$ are primal+dual optimal. Note that the strong duality is not needed for sufficiency only, but for it to be an iff statement you need strong duality:

optimality $\iff$ the points satisfy KKT + problem convex + h_i affine + slater conditions

Proof of KKT Conditions: TODO

7. Linear Programs

Remark 7.1

$$\vec{a}^\top \vec{x} = \vec{b} \iff \vec{a}^\top \vec{x} \le \vec{b} \;\; AND \;\; \vec{a}^\top \vec{x} \ge \vec{b}$$

Theorem 7.2

All LP's can be translated into the following standard form:

$$\min_{A\vec{x}=\vec{b}\ \ \vec{x}\ge 0} \vec{c}^\top \vec{x}$$

1. How to **eliminate inequality**? For expression

$$\sum_{j=1}^{n} a_{ij} x_j \le b$$

We can rewrite it as

$$\sum_{i=1}^{n} a_{ij} x_j + S_i = b_i, \quad S_i \ge 0$$

2. How to get $x_i \ge 0$ for all x_i? If x_i is unconstrained, we can always express it as the difference of two positive numbers: $x_i = x_i^+ - x_i^-$. For example:

$$\begin{aligned} &\min 2x_1 + 4x_2 \\ s.t.\ \ & x_1 + x_2 \ge 3 \\ & 3x_1 + 2x_2 = 14 \\ & x_1 \ge 0 \end{aligned}$$

We can express the constraints by introducing a slack variable x_3

$$\begin{aligned} x_1 + x_2 - x_3 = 3 \ \ x_3 \ge 0 \\ x_2 = x_2^+ - x_2^- \end{aligned}$$

By doing this we can rewrite the original problem as

$$\begin{aligned} &\min 2x_1 + 3x_2^+ - 4x_2^- \\ s.t.\ \ & x_1 + x_2^+ - x_2^- - x_3 = 3 \\ & 3x_1 + 2x_2^+ - 2x_2^- = 14 \\ & x_1 \ge 0 \\ & x_2^+ \ge 0 \\ & x_2^- \ge 0 \\ & x_3 \ge 0 \end{aligned}$$

Definition 7.3 (Polyhedron)

$$Set\{\vec{x} \in \mathbb{R}^n \mid A\vec{x} \geq \vec{b}\}\ A \in \mathbb{R}^{m*n}\ \ \vec{b} \in \mathbb{R}^m$$

Is called a polyhedron. Standarf form:

$$Set\{\vec{x} \in \mathbb{R}^l \mid c\vec{x} = \vec{d};\ \vec{x} \geq 0\}$$

Intuitively, P is the feasible region of a LP.

Definition 7.4 (Extreme points of a polyhedron)

$x \in P$ is an extreme point (i.e. vertex of P) if we cannot find two vectors $\vec{y}, \vec{z} \neq \vec{x},\ \vec{y}, \vec{z} \in P$ and $\lambda \in [0,1]$ such that $\vec{x} = \lambda\vec{y} + (1-\lambda)\vec{z}$.
That is, a point x is a vertex iff we cannot find two other point on this polyhedron such that these two points form a convex combination of x.

Theorem 7.5

P has an extreme point iff P does not contain a line.

Theorem 7.6

Consider

$$\min_{A\vec{x} \leq \vec{b}} \vec{c}^\top \vec{x}$$

Assume 1. P has an extreme point, 2. optimal solution exists and is finite. Then there exists an optimal solution that is an extreme point of P. (Note that this is not saying all solutions are extreme points.)

Proof of theorem 7.6: TODO

8. Quadratic Programs

8.a. Solutions to QP

Definition 8.1

$$\begin{aligned} &\min \frac{1}{2}\vec{x}^\top H\vec{x} + \vec{c}^\top\vec{x} \\ s.t.\ \ &A\vec{x} \le \vec{b} \\ &C\vec{x} = \vec{d} \end{aligned}$$

Is a quadratic program. In general, QPs are not convex. However, if $H = H^\top$ and H is PSD, then this is also convex.

Definition 8.2 (Moore-Penrose Pseudoinverse)
Let H be n*n matrix, $H = U\Sigma V^\top$, H has rank r, then

$$H = U\begin{bmatrix} \Sigma_{r*r} & 0 \\ 0 & 0 \end{bmatrix} V^\top$$

Then the Moore-Penrose Pseudoinverse of H is

$$H^\dagger = V\begin{bmatrix} \Sigma_{r*r}^{-1} & 0 \\ 0 & 0 \end{bmatrix} U^\top$$

And

$$\begin{aligned} HH^\dagger &= U_r U_r^\top \\ H^\dagger H &= V_r V_r^\top \\ HH^\dagger H &= H \end{aligned}$$

Theorem 8.3

Assume H symmetric and the problem is unconstrained. Then we can find the solution of QP based on the following decision tree.

1. H has at least one negative eigenvalue, then $p^* = -\infty$ by choosing eigenvector corresponding to negative eigenvalue.

2. H is PSD

2.1 $\vec{c} \in R(H)$. We rewrite f as

$$\begin{aligned} f(\vec{x}) &= \frac{1}{2}\vec{x}^\top H\vec{x} + \vec{c}^\top\vec{x} \\ &= \frac{1}{2}(\vec{x}-\vec{x}_0)^\top H(\vec{x}-\vec{x}_0) + \alpha \\ &= \frac{1}{2}\vec{x}^\top H\vec{x} + \frac{1}{2}\vec{x}_0^\top H\vec{x}_0 + \alpha \end{aligned}$$

$$\vec{c} = -H\vec{x}_0 \quad \alpha = \frac{1}{2}\vec{x}_0^\top H\vec{x}_0$$

Then we see that in order to minimize f, we want to choose $\vec{x} = \vec{x}_0$.

2.1.1 H is invertible (H is PD).

$$\vec{x}^* = -H^{-1}\vec{c}$$

2.1.2 H has non-trivial nullspace (H not invertible).

$$\vec{x}^* = -H^\dagger\vec{c} + \vec{\xi} \quad \vec{\xi} \in N(H)$$

2.2 $\vec{c} \notin R(H)$

$$\vec{c} = -H\vec{x}_0 + \vec{r} \quad \vec{r} \in N(H^\top)$$

$$\begin{aligned} f(\vec{r}) &= \frac{1}{2}\vec{r}^\top H\vec{r} + \vec{c}^\top\vec{r} \\ &= 0 + -(H\vec{x}_0 + \vec{r})^\top\vec{r} \\ &= -\vec{x}_0^\top H^\top\vec{r} - \vec{r}^\top\vec{r} \\ &= -\|\vec{r}\|_2^2 \end{aligned}$$

By taking large multiple of $\vec{r}$ we see that $p^* = \min f = -\infty$.

Theorem 8.4

Any equality constrained QP can be translated into unconstrained QP and read off solution based on previous theorem.

8.b. Applications of QP

Definition 8.5 (Linear Control)
A.k.a LQR problems, foundation for modern robotic control. Consider control problem

$$x(t+1) = Ax(t) + Bu(t)$$

Note that

$$x(t) = A^t x(0) + \sum_{i=1}^{t-1} A^{t-i-1} Bu(i)$$

Example: goal = to reach $\vec{g}$ by time T. Then we have optimization problem

$$\begin{aligned} &\min \|\vec{x}(T) - \vec{g}\|_2^2 + \sum_{t=0}^{T} \|u(t)\|_2^2 \\ s.t. \quad &x(t) = A^t x(0) + \sum_{i=0}^{t-1} A^{t-i-1} Bu(1) \end{aligned}$$

This is a complicated problem, but its constaints are entirely linear. With this optimization problem we can solve for explicit sequence of optimal control, instead of using recursion or dynamic programming.

9. Second-order Cone Problems

Definition 9.1 (cone)
Set of points $C \in \mathbb{R}^n$ is a cone iff

$$\alpha\vec{x} \in C \quad \text{if } \vec{x} \in C \quad \forall \alpha \geq 0$$

Definition 9.2 (convex cone)
C is a convex cone if

$$\alpha\vec{x} \in C \quad \forall \alpha \geq 0 \;\; \text{and } \theta_1, \theta_2 \geq 0$$

Then

$$\theta_1\vec{x}_1 + \theta_2\vec{x}_2 \in C$$

Example 9.3

$$C = \{(x, y) \mid y \geq 0\}$$

This cone has a feasible region of all region above x-axis.

Definition 9.4 (Polyhedron cone)

$$\text{Polyhedron: } \{\vec{x} \mid A\vec{x} - \vec{b}\}$$

Definition 9.5 (Ellipsoidal cone)
This is the kind of cone we are going to concern the most. Recall that

$$\vec{x}^\top P\vec{x} + \vec{q}^\top\vec{x} + r \leq 0, \quad P > 0$$

Defines a ellipsoid. Consider

$$\begin{aligned} \|A\vec{x} + b\|_2^2 &\leq c^2 \\ \vec{x}^\top A^\top A\vec{x} + 2\vec{b}^\top A\vec{x} + \vec{b}^\top\vec{b} - c^2 &\leq 0 \end{aligned}$$

Then

$$\{(\vec{x}, t) \mid \|A\vec{x} + \vec{b}t\|_2 \leq ct\}$$

is an ellipsoidal cone.

Remark 9.6 (Special case of Ellipsoidal cone (second-order cone))
Consider a second-order cone in $\mathbb{R}^3$

$$\{(\vec{x}_1, \vec{x}_2, t) \mid \sqrt{\vec{x}_1^2 + \vec{x}_2^2} \leq t\}$$

is the "ice cream cone" because it looks like a ice cream cup.

Definition 9.7 (SOCP)

$$\begin{aligned} & \min \vec{q}^\top \vec{x} \\ s.t. \quad & \|A_i \vec{x} + b_i\|_2 \leq \vec{c}_i^\top \vec{x} + \vec{d}_i \quad i = 1, 2, \cdots, m \end{aligned}$$

Is a SOCP. It is a program with constraints that takes the form of a cone.

Example 9.8

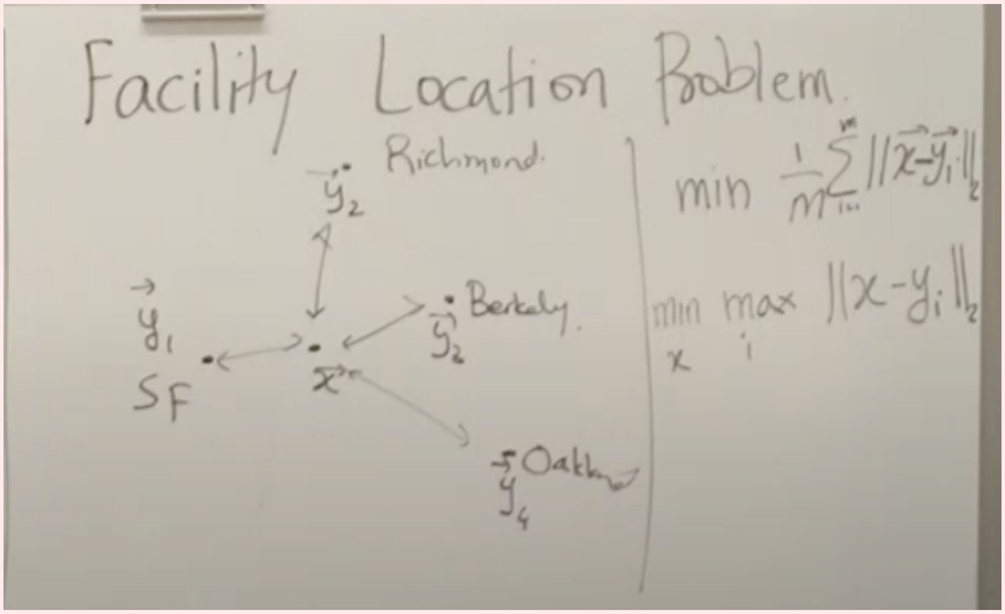
$$\min_x \sum_{i=1}^m \|A_i \vec{x} - \vec{b}_i\|_2$$

Can be formulated into a SOCP:

$$\begin{aligned} & \min_x \sum_{i=1}^m \|A_i \vec{x} - \vec{b}_i\|_2 \\ = & \min_{x, y_i, \|A_i \vec{x} - \vec{b}_i\|_2 = y_i} \sum_{i=1}^m y_i && \text{relax the equality into inequality} \\ = & \min_{x, y_i, \|A_i \vec{x} - \vec{b}_i\|_2 \leq y_i} \sum_{i=1}^m y_i \end{aligned}$$

Example 9.9 (Facility Location Problem)

Say we want to place a facility (playground or ER, etc.) and we want this facility to be close to people.



Example 9.10 (Trilateration/GPS)

A packet is transmitted at t_i^T and received at t_i^R with an offset δ such that

$$t_i^{\text{true}} = t_i^R + \delta$$

- Time of flight $f_i = t_i^{\text{true}} - t_i^T = t_i^R + \delta - t_i^T = \Delta_i + \delta$
- Distance $cf_i = c\Delta_i + c\delta = \|\vec{x} - \vec{q}_i\|_2$

Where c = the speed of light, x is the current location and q is the satellite location. We use 4 satellites, sqare all the equations, and substract them from the equation of satellite #4.

$$\|x - q_4\|_2^2 - \|x - q_1\|_2^2 = x^\top x - x^\top x + 0x + const$$
$$2(q_4 - q)^\top x + 2c^2(\Delta_4 - \Delta_1)\delta = c^2(\Delta_1^2 - \Delta_4^2) + \|q_4\|_2^2 - \|q_1\|_2^2$$

But what if instead of four satellites, we only have three satellites? We can solve this via SOCP by constructing optimization problem

$$\begin{aligned} & \min \delta \\ s.t.\quad & 2(q_3 - q_1)^\top x + 2c^2(\Delta_3 - \Delta_1)\delta = c^2(\Delta_1^2 - \Delta_3^2) + \|q_3\|_2^2 - \|q_1\|_2^2 \\ & 2(q_3 - q_2)^\top x + 2c^2(\Delta_3 - \Delta_2)\delta = c^2(\Delta_2^2 - \Delta_3^2) + \|q_3\|_2^2 - \|q_2\|_2^2 \\ & \|x - q_3\|_2 = c\Delta_3 + c\delta \end{aligned}$$

10. Newton's Method

Remark 10.1

We call gradient descent a "first-order method" because it works by taking the first-order derivative of the object function. Newton's Method is a "second-order method."

Definition 10.2

min f(x). Want $\vec{x}_0, \vec{x}_1$ converge to $\vec{x}$, which is the optimal. Then the Newton step is defined as

$$x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k)$$

Assuming that f is convex and Hessian is PD (so it's invertible).

Remark 10.3

For cases where Hessian is PSD, there is a family of method called **Quasi-Newton methods** which solve the problem using Newton's method approach but pretend the problem is second-order differentiable. It's a simple idea and we should not be intimidated by the jargon.

Remark 10.4

Newtons' method does not have a η. You can do Newton's method by manually plugging in a stepsize but by default the stepsize is always 1.

Remark 10.5 (Pros and Cons of the Newton's method)

Pros:

- Converge faster than GD

Cons:

- You have to do a Hessian inversion everytime, which is computationally expensive.

Sometimes it is cheaper to just compute the gradient, but gradient descent also takes more steps. Therefore most of the times it is unclear to us which method is computationally cheaper.