Class Notes

Dustin Leatherman

June 20, 2020

Contents

1 Review & Introduction (2020/03/31)

1.1 Review

Orthogonal: Vectors are orthogonal when the dot product = 0.

1.1.1 Basis

$$\begin{aligned} \underset{(n\times 1)}{\vec{y}} &= \underset{(n\times p)}{A}\underset{(p\times 1)}{\vec{x}} \\ &= B\vec{c} \\ &= \Sigma c_i \vec{b_i} \text{ (most } c_i = 0) \end{aligned} \tag{1}$$

A: Basis Matrix

Properties of a Good Basis

- not all are orthogonal
- Allows for a sparse vector to be used ad the constant vector $\vec{c}$

Identity Matrices are the *worst* basis because most coefficients are non-zero.

2-Sparse Vector

$$\vec{c} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 0 \\ 0 \\ 4 \end{bmatrix} \tag{2}$$

Very important!

> When dealing with Natural images and a good basis, there is a sparse vector.

1.1.2 Kernel

The kernel of a linear mapping is the set of vectors mapped to the 0 vector. The kernel is often referred to as the **null space**. Vectors should be linearly independent.

$$Ker(A) = \vec{x} \in \mathbb{R}^n \colon A\vec{x} = \vec{0} \tag{3}$$

A must be designed such that the Kernel of A does not contain any s-sparse vector other than $\vec{0}$

Main Idea: For (1), reduce $\vec{y}$ to a K-Sparse matrix to reduce the amount of non-zero numbers.

1.2 Linear Algebra Review

$$\vec{u} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}, \vec{v} = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \tag{4}$$

$$\begin{aligned} \underset{(1\times3)(3\times1)}{\vec{u}^T\vec{v}} &= \begin{bmatrix} 1 & 2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = 1 + 2 - 2 = 1 \\ &= \vec{u} \cdot \vec{v} \end{aligned} \tag{5}$$

$$\underset{(3\times1)(1\times3)}{\vec{u}\ \vec{v}^T} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 4 \\ -1 & -1 & -2 \end{bmatrix} \tag{6}$$

$\vec{u}\ \vec{v}^T \neq \vec{u}^T\ \vec{v}$

1.2.1 Inner Product

$$\begin{aligned} < \vec{a}, \vec{b} > &= \vec{a} \cdot \vec{b} \\ &= \vec{a}^T \vec{b} \end{aligned} \tag{7}$$

1.2.2 Cauchy-Schwartz Inequality

$$\vec{a} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \tag{8}$$

$$\begin{aligned} &| < \vec{a}, \vec{b} > | \leq \sqrt{1^2 + 2^2 + (-1)^2} \times \sqrt{1^2 + 1^2 + 2^2} \\ &| < \vec{a}, \vec{b} > | \leq ||\vec{a}||_2\ ||\vec{b}||_2 \text{ (euclidean/l2-norm)} \end{aligned} \tag{9}$$

1.2.3 Norms

Why is the l1 norm preferred for ML opposed to the classic l2 norm?

Philosophically,

If we looked at a sphere in l2 norm, the shadow casted would be a circle regardless of the direction of the light.

Looking at a sphere in the l1 norm is shaped as a tetrahedron. The shadow cast by a tetrahedron is different for different angles so observing the shadow provides a lot more context about the sphere.

1. Euclidean/l2

   Sphere: $||\vec{x}||_2 = \sqrt{(-4)^2 + 3^2} = \sqrt{25} = 5$

   (a) FOIL Given 2 fixed vectors x,y. Consider the l2-norm squared:

$$f(t) = ||x + ty||_2^2$$

$$\begin{aligned} f(t) =& ||x + ty||_2^2 \\ =& < x + ty, x + ty > \\ =& < x, x > + t < x, y > + t < y, x > + t^2 < y, y > \\ =& ||x||_2^2 + 2t < x, y > + t^2 ||y||_2^2 \end{aligned} \tag{10}$$

   Note: t<x,y> and t<y,x> can be combined because their dot-products are equivalent. $\vec{x} \cdot \vec{y} = \vec{y} \cdot \vec{x}$

   When using Machine Learning, don't use l2 norms. Use l1

   (b) Derivative

$$\begin{aligned} \frac{d}{dt}(||x + ty||_2^2) =& 2 < x, y > + 2t||y||_2^2 \\ =& 2x^T y + 2ty^T y \end{aligned} \tag{11}$$

2. Simplex/l1

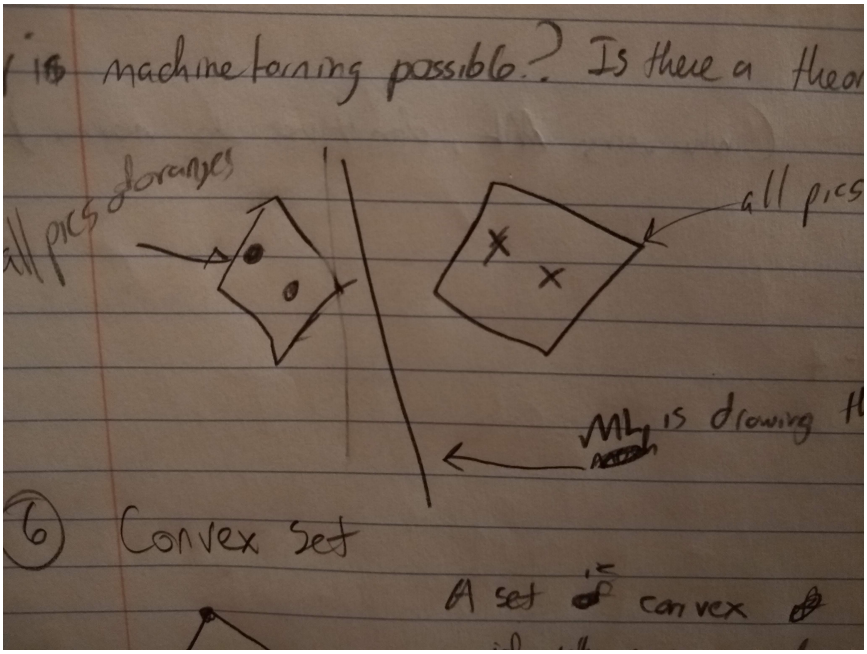
   Sphere: $||\vec{x}||_1 = |-4| + |3| = 7$

3. Infinity

   Sphere: $||\vec{x}||_\infty = Max|-4|, |3| = 4$

1.3 Optimization

Why is Machine Learning Possible? Is there a theoretical guarantee?



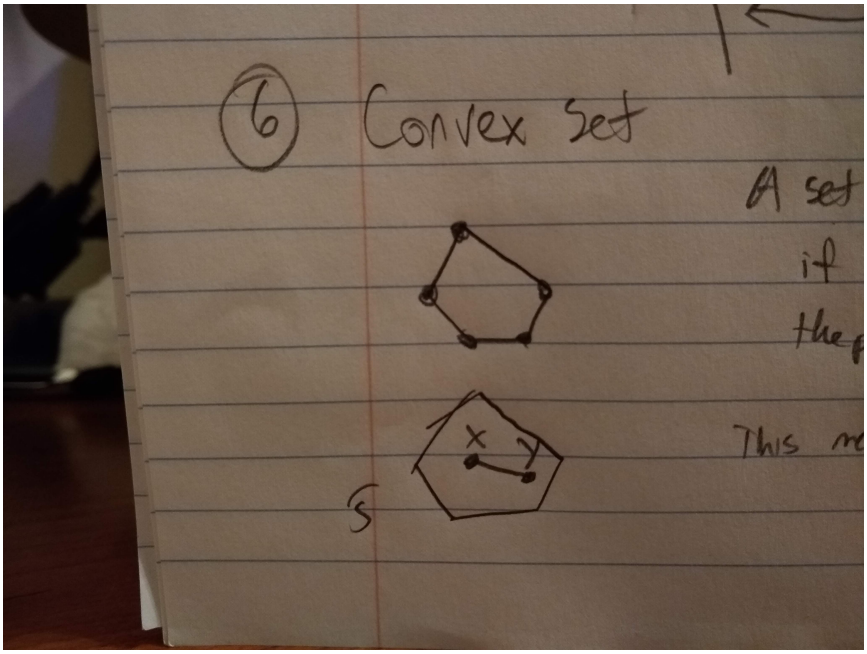
Imagine A is the set of all dogs and B is the set of all Cats

If the sets are convex and do not overlap, there exists a line between them which acts as a divider for determining whether a new pic belongs in A or B.

1.4 Convex Set

A set is convex if whenever X and Y are in the set, then for $0 \leq t \leq 1$ the points $(1-t)x + ty$ must also be in the set.

- #+ATTR$_{\LaTeX}$: scale=0.5



1.5 Separating Hyper-plane Theorem

Let C and D be 2 convex sets that do not intersect. i.e. the sets are **disjoint**.
Then there exists a vector $\vec{a} \neq 0$ and a number b such that.

$$a^T x \leq b \forall x \in C$$

and

$$a^T x \geq b \forall x \in D$$

The Separating Hyper-plane is defined as x: $a^T x = b$ for sets C, D.
This is the theoretical guarantee for ML

vector a is perpendicular to the plane b.

2 Why Separating Hyperplane Theorem & Subspace Segmentation Example (2020/04/07)

2.1 Why is Separating Hyper-plane Theorem true?

2.1.1 Math Background

Let $x = d - c,\ y = u - d$

1. Square of the l_2-norm is the inner product

$$\|x\|_2^2 = \langle x, x \rangle = x^T x$$

$$(d-c)^T(d-c) = \|d-c\|_2^2$$

2. Expansion of Vectors

$$\begin{aligned} &\|x+ty\|_2^2 \\ =&\langle x+ty, x+ty\rangle \\ =&\|x\|_2^2 + 2t\langle x, y\rangle + t^2\|y\|_2^2 \end{aligned} \tag{12}$$

3. Derivative of vector products

$$\frac{d}{dt}(\|x+ty\|_2^2) = 2x^T y + 2ty^T y$$

$$\frac{d}{dt}(\|x+ty\|_2^2)|_{t=0} = 2x^T y$$

$$\frac{d}{dt}(\|d+t(u-d)-c\|_2^2)|_{t=0} = 2(d-c)^T(u-d)$$

2.1.2 Separating Hyper-plane Theorem

C, D are convex disjoint sets. Thus there exists a vecto $\vec{a} \neq 0$ and a number b such that

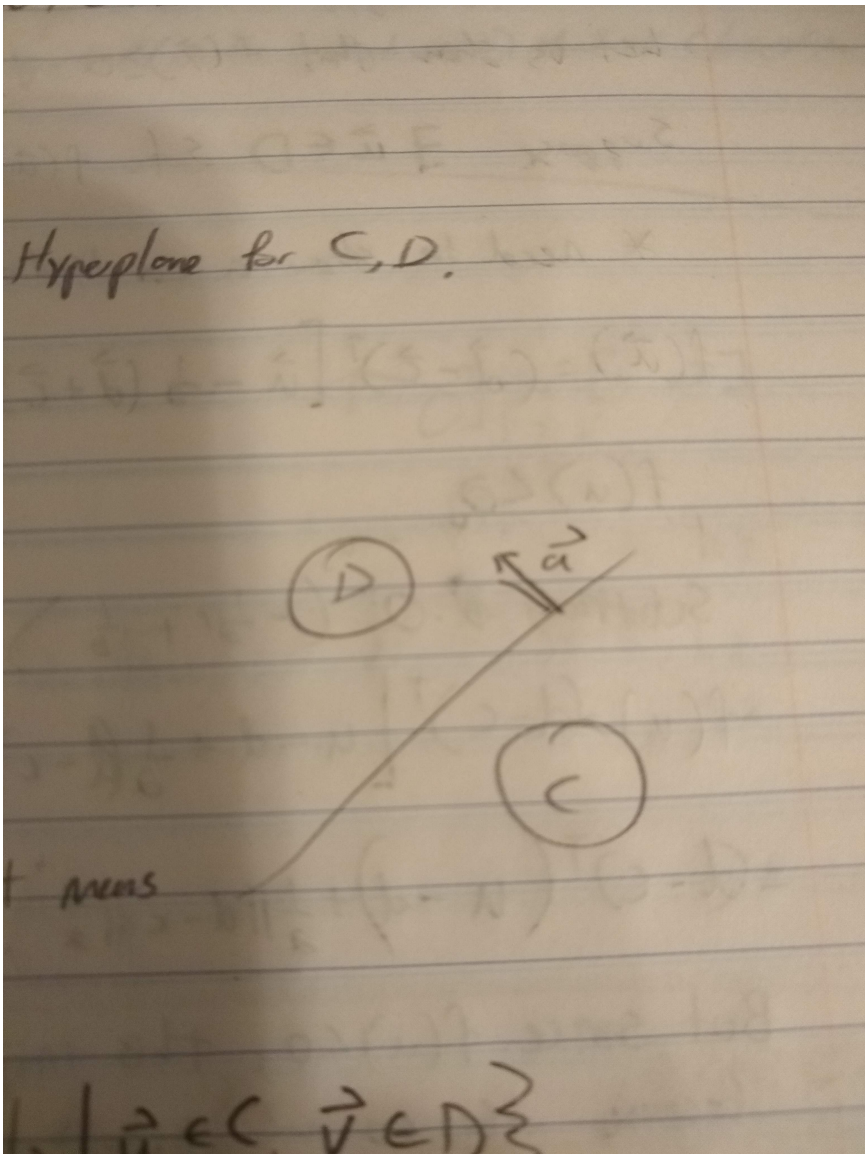
$$a^T x \leq b, \forall x \in C$$

and

$$a^T x \geq b, \forall x \in D$$

$x : a^T x = b$ is the separating hyper-plane for C,D.
When $b = 0$, then inconclusive answer.

2.1.3 Why is it true?



$$\begin{aligned} \vec{a}^T\vec{x} &\leq b \text{ on side C} \\ a\vec{^T}\vec{x} &\geq \text{ on side D} \end{aligned} \tag{13}$$

Goal: Prove $\vec{a}$ exists as that means a separating hyperplane exists.

$$dist(C,D) = min\|\vec{u}-\vec{v}\|_2 | \vec{u} \in C, \vec{v} \in D = \|\vec{c}-\vec{d}\|_2$$

where $\|\vec{u}-\vec{v}\|_2$ is the euclidean distance.
Let $\vec{a} = \vec{d} - \vec{c},\ b = \frac{1}{2}(\|\vec{d}\|_2^2 - \|\vec{c}\|_2^2)$
We will show that

$$f(\vec{x}) = a^T x - b$$

has the property that

$$f(\vec{x}) \leq 0,\ \forall \vec{x} \in C$$

and

$$f(\vec{x}) \geq 0,\ \forall \vec{x} \in D$$

Note: $(\vec{d}-\vec{c})^T\frac{1}{2}(\vec{d}+\vec{c}) = \frac{1}{2}(\|\vec{d}\|_2^2 - \|\vec{c}\|_2^2)$
What does showing something mean?
Let us show that $F(\vec{x}) \geq 0,\ \forall \vec{x} \in D$ (Argue by Contradiction)
Suppose $\exists \vec{u} \in D$ such that $f(\vec{x}) < 0$

$$f(\vec{u}) = (\vec{d}-\vec{c})^T[\vec{u} - \frac{1}{2}(\vec{d}+\vec{c})] = (\vec{d}-\vec{c})^T\vec{u} - \frac{1}{2}(\|\vec{d}\|_2^2 - \|\vec{c}\|_2^2)$$

Subtract 0

$$f(u) = (d-c)^T[u - d + \frac{1}{2}\|d-c\|]$$

$u - \frac{1}{2}d + \frac{1}{2}c$
$u - d + \frac{1}{2}d - \frac{1}{2}c$

$$f(u) = (d-c)^T(u-d) + \frac{1}{2}\|d-c\|_2^2$$

Now we observe that

$$\frac{d}{dt}(\|d + t(u-d) - c\|_2^2)|_{t=0} = 2(d-c)^T(u-d) < 0$$

and so for some small $t > 0$,

$$\|d + t(u - d) - c\|_2^2 < \|d - c\|_2^2$$

$g'(t) < 0$ means decreasing. Thus $g(t) < g(0)$.
Let's call point $p = d + t(u - d)$
Then

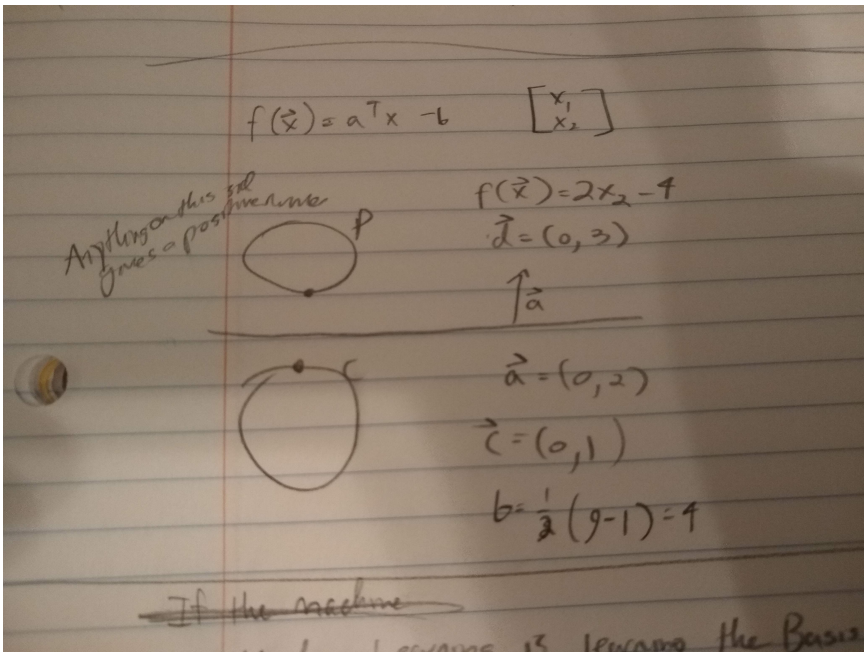
$$\|p - c\|_2^2 < \|d - c\|_2^2$$

This is a contradiction. Both d and u are in set D. Thus by the definition of convexity, $p = (1 - t)d + tu$

D is a convex set so p must also be in D. This situation is impossible since d is the point in D that is closest to c.

2.1.4 Example

Let $f(\vec{x}) = a^T x - b$



2.2 Subspace Segmentation Example

Machine Learning is learning the Basis A. If we can deduce that a vector $\vec{x}$ is a linear combination of A, then a vector is a subspace of Basis A and we

know that it belongs to A.

$$V_1 = (x, y, z) \in R^3 : z = 0$$
$$V_2 = (x, y, z) \in R^3 : x = 0, y = 0$$

V_i is the affine variety (it is also a Ring, Module)
Apply a Veronase map with degree 2 to lift up from 3 to 6 dimensions.

$$\nu_n \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x^2 \\ y^2 \\ z^2 \\ xy \\ xz \\ yz \end{bmatrix}, \nu_n : R^3 \to R^6$$

$$\begin{aligned} z_1 = (3, 4, 0), z_2 = (4, 3, 0), \\ z_3 = (2, 1, 0), z_4 = (1, 2, 0), \\ z_5 = (0, 0, 1), z_6 = (0, 0, 3), z_7 = (0, 0, 4) \end{aligned} \tag{14}$$

Plug the sample points into the Veronase map to produce a matrix L

$$L = \begin{bmatrix} 9 & 16 & 4 & 1 & 0 & 0 & 0 \\ 16 & 9 & 1 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 9 & 6 \\ 12 & 12 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \in R^{6 \times 7}$$

solve for $\vec{c}$, where $\vec{c}^T L = \vec{0}$

$$\vec{c}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \vec{c}_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Rank(L) = 4 (since there are 4 linearly independent rows)

$$\begin{aligned} q_1(X) =& \vec{c}^T \nu_n(X) \\ =& xz \\ q_2(X) =& \vec{c}_2^T \nu_n(X) \\ =& yz \end{aligned} \tag{15}$$

We have:

$$\begin{aligned} q_1(X) &= xz \quad V_1 = (z = 0) \\ q_2(X) &= yz \quad V_2 = (x = 0, y = 0) \end{aligned} \tag{16}$$

Observe:
$V_1 \cup V_2 = ((x, y, z) \in R^3 : q_1(X) = 0, q_2(X) = 0)$
Construct the Jacobian matrix
$\mathrm{J(Q)(X)} = \begin{bmatrix} \frac{\partial q_1}{\partial x} & \frac{\partial q_1}{\partial y} & \frac{\partial q_1}{\partial z} \\ \frac{\partial q_2}{\partial x} & \frac{\partial q_2}{\partial y} & \frac{\partial q_2}{\partial z} \end{bmatrix} = \begin{bmatrix} z & 0 & x \\ 0 & z & y \end{bmatrix}$

1. When $z = z_1 = (3, 4, 0), J(Q)(z_1) = \begin{bmatrix} 0 & 0 & 3 \\ 0 & 0 & 4 \end{bmatrix}$

   When $z = z_3 = (2, 1, 0)$, $J(Q)(z_3) = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix}$

   The right null space of $J(Q)(z_1)$ has basis $\vec{b}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \vec{b}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$

2. When $z = z_5 = (0, 0, 1)$, $J(Q)(z_5) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$

   When $z = z_7 = (0, 0, 4)$, $J(Q)(z_7) = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \end{bmatrix}$ The right null space of $J(Q)(z_5)$ has basis $\vec{b} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

$C = [\vec{c}_1 | \vec{c}_2]$

3 Sparse Representation & Problem P0 . P1 (2020/04/14)

3.1 Big Idea

Your Data is a vector $x \in R^N$ where all vectors are column vectors. Each x is s-sparse i.e. each vector has at **most** s non-zero entries. Let s = 5000. We don't know where the non-zero entries are located.

Let $\underset{(m \times N)}{A}$, $m < N$
$N = 100,000,\ m = 20,000$
Short + Wide Matrix

This is the opposite of the kinds of matrices seen in Linear Regression which are tall and skinny.

What if we can design a matrix $A \in R^{m \times N}$ so that for each s-sparse $\vec{x} \in R^N$, you can store $\vec{y}$ instead? ($A\vec{x} = \vec{y}$)

Q: Is there a way to get back $\vec{x}$ from $\vec{y}$? We observe $\vec{y}$.

A: Yes!

Properties of A

- A cannot be the 0 matrix.
- if $\vec{x}_1$ is s-sparse and $\vec{x} \neq 0$, what if $\vec{x}_1$ is in $ker(A)$? No! that would return $\vec{0}$ which means we cannot reconstruct the original matrix since there are multiple vectors in Ker(A).

Using Techniques from 1955

1. Is $\vec{x}$ the inverse of $\vec{y}$ or psuedo-inverse, or Moore-Penrose inverse, or...?

$$\begin{aligned} \vec{y} &= A\vec{x} \\ A^{\#}\vec{y} &= A^{\#}A\vec{x} \text{ where } \mathrm{A}^{\#}A = I \end{aligned} \tag{17}$$

Doesn't work! This is because there is no way to guarantee that $\vec{x}$ is a s-sparse vector.

1. Can we use gradient descent to solve for $\vec{x}$ to minimize $\|\vec{y} - A\vec{x}\|_2$

   No! Why?

   pick any vector $\vec{v} \in Ker(A)$. $\vec{y} = A(\vec{x} + \vec{v})$ however, $(\vec{x} + \vec{v})$ may not be sparse.

New math was needed to solve this problem so it was created in 2005 by Donoho, Candes, and Tao using the \$l$_1$\$-norm instead of the euclidean norm (l_2).

3.2 Background

\$l$_1$\$-norm: $\|x\|_1 = |x_1| + |x_2| + |x_3|$

\$l$_2$\$-norm: $\|x\| = \sqrt{|x_1|^2 + |x_2|^2 + |x_3|^2}$

For $\vec{x} \in R^n$, $\vec{y} \in R^N$, then

$$\|\vec{x}+\vec{y}\| \leq \|x\|_1 + \|y\|_1$$

For a norm to be valid, it must uphold the **Triangle Inequality**.
$\vec{a}$ is one side of a triangle, $\vec{b}$ is a second side, third side, ...

$$\begin{aligned} |\vec{a}+\vec{b}| &\leq |\vec{a}| + |\vec{b}| \\ \|\vec{x}+\vec{y}\|_1 &\leq \|\vec{x}\|_1 + \|\vec{y}\|_1 \\ \|\vec{x}+\vec{y}\|_2 &\leq \|\vec{x}\|_2 + \|\vec{y}\|_2 \\ \|\vec{x}+\vec{y}\|_2 &\leq \|\vec{x}\|_\infty + \|\vec{y}\|_\infty \end{aligned} \tag{18}$$

It also must be distributive:
If $\vec{x}_1 + \vec{x}_2 = \vec{y}$, then $(\vec{x}_1 + \vec{x}_2)\cdot\vec{a} = \vec{y}\cdot\vec{a}$ for any $\vec{a}$

$$\langle \vec{x}_1 + \vec{x}_2, \vec{a}\rangle = \langle \vec{y}, \vec{a}\rangle \rightarrow \langle \vec{x}_1, \vec{a}\rangle + \langle \vec{x}_2, \vec{a}\rangle = \langle \vec{y}, \vec{a}\rangle$$

3.3 Warm-up

$A = [\vec{a}_1|...|\vec{a}_N]$
$\|\vec{a}_j\|_2 = 1 = \langle \vec{a}_j, \vec{a}_j\rangle$
Let $\vec{v} \in Ker(A),\ \vec{v} \neq \vec{0},\ \vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ ... \\ v_N \end{bmatrix}$
Assume $\vec{a}_j$ are unit vectors.
Pick $i = 3$ observations.

1. Multiply by 1. Be Sneaky.
   $v_i = v_i\langle \vec{a}_i, \vec{a}_i\rangle$
2. $\vec{v} \in Ker(A)$

$$\begin{aligned} &v_1a_1 + v_2a_2 + ... + v_na_n = \vec{0} \\ \rightarrow &\langle v_1a_1 + ... + v_Na_N, a_i\rangle = \langle \vec{0}, a_i\rangle \\ \rightarrow &\langle v_1a_1, a_i\rangle + ... + \langle v_Na_N, a_i\rangle = \langle \vec{0}, a_i\rangle \end{aligned} \tag{19}$$

Keep $v_3\langle a_3, a_i\rangle$ on the left side. Move everything to the other side. Thus,

$$v_i = \langle v_i a_i, a_i \rangle = - \sum_{j=1, j \neq i} v_j \langle a_j, a_i \rangle$$

Since $i = 3$, $v_3 \langle a_3, a_i \rangle = v_i$

$$|v_i| \leq \sum_{j=1, ji} |v_j| \cdot |\langle a_j, a_i \rangle|$$

What is the absolute value of a single number in $Ker(A)$? There is a relation between v_i and the rest of the entries in $\vec{v}$.

Why "=" becomes $\leq$

For example, if -2 = 3 + (- 5), then

3.4 Getting Ready to Formulate the Problem

3.4.1 Problem P0

Find the s-sparse $\vec{x} \in R^N$ such that $\vec{y} = A\vec{x}$.

Ex. Problem 1 HW 1.

Find a 2-sparse vector $\vec{x} \in R^8$ such that $\vec{y} = A\vec{x}$.

There are $\binom{8}{2}$ 2-sparse vectors. (28).

Imagine N = 100,000 and s = 5000. Not feasible to try all sparse-vectors.

3.4.2 Problem P1 (Convex Optimization)

Given $A \in R^{m \times N}$ and measurement $\vec{y} = R^m$, solve the optimization problem,

$$\underset{x \in R^N}{min} \|x\|_1$$

subject to constraint $y = A\vec{x}$

Find a condition on matrix A, so that solving P1 will recover the s-sparse vector $x \in R^N$

3.5 Null Space Property of Order s

3.5.1 Setting up Notation

Let $\vec{v} \in Ker(A),\ \vec{v} \neq \vec{0}$

Let the set of indices , where $\vec{v}[j] \neq 0$ to be S.

e.g. $\vec{x} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 2 \\ 3 \\ 0 \\ 4 \end{bmatrix}$

$S = \{3, 5, 7\}$ (non-zero indices. Also called the support vector of $\vec{v}$).

$|S| = s$ (number of elements. i.e. sparsity)

$\bar{S} = \{1, 2, 4, 6\}$ (complement. i.e zero indices)

$$\vec{v} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 2 \\ -2 \\ 2 \end{bmatrix}, \vec{v}_S = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 2 \\ 0 \\ 2 \end{bmatrix}, \ \vec{v}_{\bar{S}} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ -2 \\ 0 \end{bmatrix}$$

$\vec{v} = \vec{v}_S + \vec{v}_{\bar{S}}$

3.5.2 Definition

Let A be a $m \times N$ matrix.

Let S be a subset or $\{1, 2, 3, ..., N\}$. Suppose $N = 50$, and $S = \{3, 5, 7\}$

1. We say that a matrix A satisfies the null space property with respect to a set S if
$$\|\vec{v}_S\|_1 < \|\bar{S}\|, |\forall \vec{v} \vec{\in} Ker(A)$$

2. If it satisfies the null space property with respect to any set S of size s where S is a subset of $\{1, 2, 3, ..., N\}$. $s < N$

If a matrix satisfies this property, what does it buy us?

If a matrix A satisfies the Null Space property of order s, then solving problem P1 will solve P0. i.e. you can recover any s-sparse vector $\vec{x}$ from the measurement y where $\vec{y} = A\vec{x}$

If A has a small coherence, then it satisfies the Null Space Property of order s.

Let $A = [\vec{a}_1|...|\vec{a}_N]$

$$\mu_1 = \max_{j \neq k} |\langle \vec{a}_j, \vec{a}_k \rangle|$$

Assume $\vec{a}_j$ has l_2-norm equal to 1.

3.5.3 Theorem

Same assumptions as above.

Suppose $\mu_1 \cdot s + \mu_1 \cdot (s-1) < 1$

The matrix satisfies the Null Space property of order s.

Remarks

1. $\mu_1(2s-1) < 1$ if true, then A satisfies NSP of order s. It is not a necessary condition. It is a sufficient condition.

2. From the warm up, if we fix an index i, then for $\vec{v} \in Ker(A)$,

$$|v_i| \leq \sum_{j=1, j \neq i} |v_j| \cdot |\langle \vec{a}_j, \vec{a}_i \rangle| \tag{20}$$

1. Note that $|v_i|$ is just one term in $\|v\|_1$ because

$$\|v\|_1 = |v_1| + |v_2| + ...$$

3.5.4 Proof

Given A is an $m \times N$ matrix. $A = [\vec{a}_1|...|\vec{a}_N]$.

Suppose $\|\vec{a}_j\| = 1,\ \mu_1 \cdot s + \mu_1 \cdot (s-1) < 1$

Show that NSP of order s holds.

i.e.

$$\|\vec{v}_S\| < \|\vec{v}_{\bar{S}}\|, \forall \vec{v} \in ker(A) | \{\vec{0}\}$$

and for every set

$$S \subset \{1, 2, 3, ..., N\} \text{with} |S| = s$$

Let $\vec{v} = Ker(A)$

$$\vec{v} = \begin{bmatrix} v_1 \\ v_2 \\ ... \\ v_n \end{bmatrix}$$

$A\vec{v} = v_1\vec{a}_1 + ... + v_N\vec{a}_N = \vec{0}$
Let $S \subset \{1, 2, ..., N\},\ |S| = s$. Pick any $\vec{a}_i, i \in S$
Then $v_i = v_i\langle\vec{a}_i, \vec{a}_i\rangle$. Also, $v_1\langle\vec{a}_i, \vec{a}_i\rangle + ... + v_N\langle\vec{a}_N, \vec{a}_i\rangle = 0$

$$\begin{aligned}
&\rightarrow v_i = v_i\langle\vec{a}_i, \vec{a}_i\rangle = -\sum_{j=1, j\neq i} v_i\langle\vec{a}_j, \vec{a}_i\rangle \\
&\rightarrow v_i = -\sum_{l\in S} v_l\langle\vec{a}_l, \vec{a}_i\rangle - \sum_{j\in S, j\neq i} v_j\langle\vec{a}_j, \vec{a}_i\rangle \\
&\rightarrow |v_i| \leq \sum_{l\in S} |v_l||\langle\vec{a}_l, \vec{a}_i\rangle| + \sum_{j\in S, j\neq i} |v_j||\langle\vec{a}_j, \vec{a}_i\rangle|
\end{aligned} \tag{21}$$

sum over all $i \in S$ to get
$\|\vec{v}_S\|_1 = \sum_{i\in S} |v_i|$

This adds up all the inequalities for one inequality to rule them all.

$$\begin{aligned}
&\leq \sum_{i\in S}\sum_{l\in\bar{S}} |v_l| \cdot |\langle\vec{a}_l, \vec{a}_i\rangle| + \sum_{i\in S}\sum_{j\in S, j\neq i} |v_j| \cdot |\langle\vec{a}_j, \vec{a}_i\rangle| \\
&= \sum_{l\in\bar{S}} |v_l| \sum_{i\in S} |\langle\vec{a}_l, \vec{a}_i\rangle| + \sum_{j\in S} |v_j| \sum_{i\in S, i\neq j} |\langle\vec{a}_j, \vec{a}_i\rangle| \\
&\leq \sum_{l\in S} |v_l|\mu_1 \cdot s + \sum_{j\in S} |v_j|\mu_1(s-1) \\
\|\vec{v}_S\|_1 \leq &\mu_1 \cdot s\|\vec{v}_{\bar{S}}\| + \mu_1(s-1)\|\vec{v}_{\S}\|
\end{aligned} \tag{22}$$

$$(1 - \mu_1(s-1))\|\vec{v}_{\bar{S}}| \ \leq \mu_1 \cdot s\|\vec{v}_S\|$$

Since $\mu_1(s-1) + \mu_1(s) < 1$ by hypothesis, so $1 - \mu_1(s-1) \geq \mu_1(s)$ and hence $\|\vec{v}_S\|_1 < \|\vec{v}_{\bar{S}}\|_1$

3.6 Ways to Solve P1

There are 8 algos to solve P1. The worst performing one is Linear programming.

This is one of the Algos

3.6.1 Algos

$A = \begin{bmatrix} 1 & 1 \end{bmatrix}$ $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$a_{11} = a_{12} = 1$

$Q = \begin{bmatrix} \frac{1}{w_1} & 1 \\ 0 & \frac{1}{w_2} \end{bmatrix}$

1. Minimize $\|\vec{x}_1\|$ subject to $\vec{y} = A\vec{x}$

$$\begin{aligned} \vec{y} =& (AA^T)(AA^T)^{-1}\vec{y} \\ \vec{y} =& A(A^T(AA^T)^{-1}\vec{y}) \end{aligned} \tag{23}$$

Why not let $\vec{x} = (A^T(AA^T)^{-1}\vec{y})$
maybe we can do better.
$\vec{y} = AQA^T(AQA^T)\vec{y}$
Why not let $\vec{x} = (QA^T(AQA^T)^{-1}\vec{y})$
How to choose Q?

1. $min \sum_{i=1}^{N} W_i x_i^2$ subject to $\vec{y} = A\vec{x}$
   This is not the l_1-norm but it would be if $w_i = \frac{1}{|x_i|}$.
   solve 2. then substitute w_i

2. min: $w_1 x_1^2 + w_2 + x_2^2$ subject to $y = a_{11}x_1 + a_{12}x_2$
   $f(x_1) = w_1 x_1^2 + w_2(y - x_1)^2$
   $f'(x_1) = 0$ solve for x_1
   $2w_1 x_1 + 2(y - x_1)(-1)w_2 = 0$
   $x_1 = \frac{w_2}{w_1 + w_2} y,\ x_2 = \frac{w_1}{w_1 + w_2} v$

$$\begin{aligned} AQA^T =& \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{w_1} & 0 \\ 0 & \frac{1}{w_2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ =& \frac{w_1 + w_2}{w_1 w_2} \end{aligned} \tag{24}$$

$$QA^T(AQA^T)^{-1}y = \begin{bmatrix} \frac{1}{w_1} \\ \frac{1}{w_2} \end{bmatrix} \frac{w_1 w_2}{w_1 + w_2} y \tag{25}$$

4 Sparse Representation pt 2 (2020/04/21)

4.1 Historical Perspective

Why is the visual system so powerful? Hypothesis is our brain uses sparse representation of Visual Data.

Let a picture $\vec{y} = c_1\vec{b}_1 + ... + c_n\vec{b}_n$

so that most c_j are zero.

Sparse representation used to be called Sparse Coding.

Robust Facial Recognition uses Sparse Subspace Clustering.

Given 19 x 19 images, let $Y = [\vec{Y}_1|...|\vec{Y}_{45}],\ \vec{y}_j \in R^{361}$

19 * 19 = 361

Given Y, solve for matrix C

$$Y = YC,\ diag(C) = \vec{0}$$

Since we don't want $Y_i = Y_i$, that is why the constraint $diag(C) = \vec{0}$ is introduced. It ensures that a group of vectors can be a linear combination of others.

Each column of C is sparse since we want all column vectors to be a linear combination of a smaller set of columns.

4.2 Example - Handwritten Digit Recognition

Given 28 x 28 images, Let $B = [\vec{y}_1|...|\vec{y}_{4000}]$ where each $\vec{y}_j \in R^{784}$

- 800 images of 0, 1-800
- 800 images of 1, 801-1600
- 800 images of 2, 1601-2400
- 800 images of 3, 2401-3200
- 800 images of 8, 3201-4000

Let $\vec{f}$ be a new image of 2. Solve for X such that $\vec{f} = B\vec{x}$

Assume $\vec{x}$ is 20-sparse.

We would like to see the only **non-zero** entries at position 1601-2400.

Columns outside the range may be non-zero as well. There is a 95% probability that a digit will be 2, 5% it will be another digit.

4.2.1 Qualitative Theorem

Given $A^{m \times N}$ with $m << N$. If A is a Gaussian random matrix, then with overwhelming high probability, it satisfies some Exact Recovery Condition for s-sparse Vectors.

For most large undetermined systems of linear equations, the minimal l_1-norm solution is also the sparsest solution.

Topics of Research:

- Theory of Random Matrices
- Banach Spaces

4.3 Solving P1 solves P0. Why?

PO

Find the s-sparse $\vec{x} \in R^N$ such that $\vec{y} = A\vec{x}$.

P1

$A \in R^{m \times N}$ and measurement $\vec{y} \in R^m$. Solve optimization problem,

$$\min_{x \in R^N} \|x\|_1$$

subject to the constraint $y = A\vec{x}$

Suppose $\vec{y} = A\vec{x}$ and $\vec{y} = A\vec{z}$. Suppose $\vec{x}$ is a sparse vector and $\vec{z}$ is **not**.

We want to show that $\|\vec{x}\|_1 < \|\vec{x}\|_1$ - Null Space property of order S

$\|\vec{x}\|_1 = \|\vec{x} - \vec{z}_S + \vec{z}_S\|_1$ - $\vec{z}$ restricted to some Set S. (Subtract 0 so we can use triangle inequality).

Let $\vec{v} = \vec{x} - \vec{z},\ \vec{v} \in Ker(A)$

$A(\vec{x} + \vec{z}) = A\vec{v} = \vec{0}$

$$\begin{aligned}
\|\vec{x}\|_1 \leq & \|\vec{x} - \vec{z}_S\|_1 + \|z_S\|_1 && && \text{(26a)}\\
= & \|\vec{v}_S\|_1 + \|\vec{z}_S\|_1 && && \text{(26b)}\\
< & \|\vec{z}_S\|_1 + \|\vec{v}_{\bar{S}}\|_1 && \text{via Null Space Property} && \text{(26c)}\\
= & \| - \vec{z}_{\bar{S}}\|_1 + \|z_S\|_1 && \|x_{\bar{s}}\|_1 = 0 \text{ since x is sparse} && \text{(26d)}\\
= & \|\vec{z}\|_1 && && \text{(26e)}
\end{aligned}$$

4.4 Adjoint

Let $T: V \to W$. For example, T can be a matrix from R^3 to R^2. In this case, V is R^3 and W is R^2

We write T^* for the adjoint of T.

$$\forall x \in V,\ \forall y \in W,\ \langle Tx, y\rangle = \langle x, T^* y\rangle$$

Horrible way to think of it, when T is a matrix, the adjoint is the same as the transpose.

Q: When A is an orthogonal matrix, what is A^*A? I

Hint: each column has l_2-norm 1, distinct cols are perpendicular.

Q: When A is an orthogonal matrix, why is $\|Ax\|_2 = \|x\|_2$ for every vector x? (This is known as an isometry)

$$\|Ax\|_2^2 = \langle Ax, Ax\rangle = \langle x, A^*Ax\rangle = \langle x, x\rangle = \|x\|_2^2$$

This shows that $\|Ax\|_2^2$ is not too different than $\|x\|_2^2$

4.5 Restricted Isometry Property (RIP)

$A \in R^{m\times N}$ satisfies the restricted isometry property of order s and level δ_s $(0 < \delta_s \leq 1)$

$$(1-\delta_s)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1+\delta_s)\|x\|_2^2,\ \forall \text{ s-sparse } x \in R^N$$

> Any s columns of the matrix A are **nearly** orthogonal to each other.

Q: What can we say about $|\langle (I - A^*A)x, x\rangle|$ when vector is s-sparse?

This is a small number.

Let $u,\ v \in R^N$ and $S \in \{1, 2, 3, ..., N\},\ |S| = s$

What can we say about the following?

$$|\langle u, (I - A * A)v\rangle|$$

We would like to be able to say

$|\langle u, (I - A^*A)v\rangle| \leq \delta_t \|u\|_2 \|v\|_2$

4.5.1 How to think about RIP?

Suppose A satisfies the restricted isometry property of order s.

Intuition: **Hopefully**, the matrix A^*A behaves like the Identity Matrix. $(I - A^*A)$ is small.

If you take some s-sparse vector $\vec{x}$ and multiply it by $I - A^*A$, hopefully, the resulting vector will also be small.

4.5.2 Algorithm

Consider the following vectors,

$$\vec{x}_1 = \begin{bmatrix} 10 \\ -20 \\ 3 \\ -4 \\ 5 \\ -6 \\ -7 \\ 8 \\ 4 \end{bmatrix}, \ \vec{x}_2 = \begin{bmatrix} 10 \\ -20 \\ 0 \\ 0 \\ 0 \\ 0 \\ -7 \\ 8 \\ 0 \end{bmatrix}$$

Hard Threshold

$\tau_s(\vec{x})$ is the vector that keeps the s entries that are the largest in Absolute Value.

Example: When $s = 4, \ \tau_s(\vec{x}_1) = \vec{x}_2$

$\tau_s(\cdot)$ is an operator that takes a vector and will output a sparse vector.

$$\begin{aligned} \vec{u}_n =& \vec{x}_n + A^*(\vec{y} - A\vec{x}_n), \text{ where } \vec{y} = A\vec{x} && \text{(27a)} \\ =& \vec{x}_n + (A^*A\vec{x} - A^*A\vec{x}_n) && \text{(27b)} \\ =& (I - A^*A)\vec{x}_n + A^*A\vec{x} && \text{(27c)} \end{aligned}$$

- expect $\vec{u}_n$ close to $\vec{x}$
- however, $\vec{u}_n$ may not be sparse. Thus use $\tau_s(\cdot)$

  Iterative Hard Thresholding

$$\vec{x}_{n+1} = \tau_x(\vec{x}_n + A^*(\vec{y} - A\vec{x}_n))$$

4.6 Operator Norm

$\|A\| = \underset{x \neq 0}{max} \frac{\|Ax\|_2}{\|x\|_2}$

How much influence does A have on a vector x? Shrink, stretch, compress?

Describes how big a matrix is. If A is 2 x 3, then take $\vec{x} \in R^3,\ x \neq 0$

What is

$$\|A\| = max\{\|Ax\|_2 \colon \|x\|_2 = 1\}$$

4.6.1 Inner Product

Let A be a matrix . The inner product of two vectors Ax and y has this property,

$$|\langle Ax, y \rangle| \leq \|A\| \cdot \|x\|_2 \|y\|_2$$

Where $\|A\|$ is the operator norm of A.

By Cauchy-Schwartz Inequality,

$$\|\langle Ax, y \rangle\| \leq \|Ax\|_2 \cdot \|y\|$$

By def,

$$\|Ax\| \leq \|A\| \cdot \|x\|_2$$

Thus,

$$\|\langle Ax, y \rangle\| \leq \|A\| \cdot \|x\|_2 \cdot \|y\|_2$$

5 Sparse Representation Pt 3 (2020/04/28)

5.1 Expanding on RIP

Expanding upon RIP

Any S columns of the matrix A are nearly orthogonal to each other.

5.2 Expanding on IHT

Expanding upon the IHT Algorithm,

$\tau_x(\cdot)$ is an non-linear operator that outputs a sparse matrix. The operator is non-linear because it does not *change* the dimensions on the vector. i.e. $R^n \to R^n$. You will not be able to find a matrix that will return the same output as this operator.

$\tau_s(\vec{x}_1) = x_2$

Which means both $\vec{x}_1$ and $\vec{x}_2$ have an inner product.

The IHT algorithm is described below:

$$\begin{aligned}\vec{u}_n =& \vec{x}_n + A^*(\vec{y} - A\vec{x}_n), \text{ where } \vec{y} = A\vec{x} && (28a)\\ =& \vec{x}_n + (A^*A\vec{x} - A^*A\vec{x}_n) && (28b)\\ =& (I - A^*A)\vec{x}_n + A^*A\vec{x} && (28c)\end{aligned}$$

We expect $\vec{u}_n$ is close to $\vec{x}$.

What does it mean for a matrix A to be small? matrix A is small when $A\vec{x}$ is small.

5.3 IHT Proof

Suppose A satisfies RIP of order 3s with

$$\delta_{3s} < \frac{1}{2}$$

δ_{3s}: relaxation.

$3s$: every 3s columns need to be orthogonal

$\frac{1}{2}$: how far from orthogonality the difference can be.

Then the sequence $\{\vec{x}_n\}$ defined by

$$\vec{x}_{n+1} = \tau_S(\vec{x}_n + A^*(\vec{y} - A\vec{x}_n))$$

will converge to $\vec{x}$

Note: 3s-sparse vectors and s-sparse vectors are **not** the same.

5.3.1 How to think about this?

u and v are 2s-sparse.

Let S_1 be the support of u. Meaning $S_1 = \{j : u(j) \neq 0\}$

Let S_2 be the support of v.

Let S be the union of S_1 and S_2. Assume $|S| = 3s$
If A satisfies RIP of order 3s. Then
$|\langle u, (I - A^*A)v\rangle| \leq \delta_{3s}\|u\|_2 \cdot \|v\|_2$

$$\|\langle u, (I - A^*A)\rangle\| \leq \|u\|_2 \ \|v(I - A^*A)\|_2 \tag{29a}$$
$$\leq \|u\|_2 \ \|v\delta_{3s}\|_2 \tag{29b}$$
$$\leq \delta_{3s} \ \|u\|_2 \ \|v\|_2 \tag{29c}$$

5.3.2 Explanation: Why is the theorem true?

We want to find a constant $\lambda,\ 0 \leq \lambda < 1$ s.t.

$$\|x_{n+1} - x\|_2 \leq \lambda\|x_n - x\|_2,\ \forall\ n = 1, 2, 3, ...$$

Why?

$$\begin{aligned} \|x_4 - x\|_2 \leq & \lambda\|x_3 - x\|_2 \\ \|x_3 - x\|_2 \leq & \lambda\|x_2 - x\|_2 \\ \|x_2 - x\|_2 \leq & \lambda\|x_1 - x\|_2 \end{aligned} \tag{30}$$

Therefore,

$$\|x_4 - x\|_2 \leq \lambda^{n-1} \ \|x_1 - x\|_2 \tag{31}$$

In general,

$$\|x_{n+1} - x\|_2 \leq \lambda^{n-1} \ \|x_1 - x\|_2 \tag{32}$$

as $n \to \infty,\ \lambda \to 0$ (because $\lambda < 1$)

$$\vec{x}_{n+1} = \tau_S(\vec{x}_n + A^*(\vec{y} - A\vec{x}_n))$$

and

$$x_{n+1} = \tau_S(u_n)$$

$x_{n+1},\ x$ are s-sparse.
Key Observation: Which one (x_{n+1} or x) is a better approximation to u_n?

x_{n+1}
Thus,

$$\|u_n - x_{n+1}\|_2^2 \leq \|u_n - x\|_2^2 \tag{33}$$

What is $u_n - x$?

$$\begin{aligned} u_n - x =& x_n + A^*A(x - x_n) - x && \text{(34a)} \\ =& (I - A^*A)x_n + (A^*A - I)x && \text{(34b)} \\ =& (I - A^*A)(x_n - x) && \text{(34c)} \end{aligned}$$

What is $u_n - x_{n+1}$?

$$\begin{aligned} \|u_n - x_{n+1}\|_2^2 =& \|u_- x_{n+1} - (x - x)\|_2^2, && \text{subtract 0} \\ & && \text{(35a)} \\ =& \|(u_n - x) - (x_{n+1} - x)\|_2^2, && \text{square of l2 norm os inner product} \\ & && \text{(35b)} \\ =& \langle (u_n - x) - (x_{n+1} - x), (u_n - x) - (x_{n+1} - x) \rangle && \text{(35c)} \\ =& \|u_n - x\|_2^2 - 2\langle u_n - x, x_{n+1} - x \rangle + \|x_{n+1} - x\|_2^2 && \text{(35d)} \end{aligned}$$

From the above two formulas, we getattr

$$-2\langle u_n - x, x_{n+1} - x \rangle + \|x_{n+1} - x\|_2^2 \leq 0 \tag{36}$$

This is the same as

$$\|x_{n+1} - x\|_2^2 \leq 2\langle u_n - x, x_{n+1} - x \rangle$$

What is $u_n - x$?

$$u_n - x = (I - A^*A)(x_n - x)$$

$$\langle u_n - x, x_{n+1} - x \rangle = \langle (I - A^*A)(x_n - x), x_{n+1} - x \rangle$$

Thus,

$$u = x_{n-x},\ v = x_{n+1} - x$$

Why? $x_n - x$ is 2s-sparse and $x_{n+1} - x$ is also 2s-sparse.

We have shown that

$$\langle u_n - x, x_{n+1} - x \rangle \leq \delta_{3s} \|x_n - x\|_2 \cdot \|x_{n+1} - x\|_2$$

$$\begin{aligned} \|x_{n+1} - x\|_2^2 &\leq 2\delta_{3s} \|x_n - x\|_2 \cdot \|x_{n+1} - x\|_2 \\ \|x_{n+1} - x\|_2 &\leq 2\delta_{3s} \cdot \|x_n - x\|_2 \end{aligned} \tag{37}$$

The hypothesis is $\delta_{3s} < \frac{1}{2}$ and so $0 \leq \lambda < 1$

$$\|x_{n+1} - x\|_2 \leq \lambda \|x_n - x\|_2 \tag{38}$$

Explanation succeeded

5.4 Convex Functions

Pick any norm, $\|\cdot\|_1,\ \|\cdot\|_2$

We have the triangle inequality

$$\|x + y\| \leq \|x\| + \|y\| \tag{39}$$

Suppose we define $f(x) = \|x\|$ for any $x \in R^d$ and $0 \leq \theta \leq 1$.

$$\begin{aligned} f(\theta x = (1-\theta)y) =& \| + (1-\theta)y\| \leq \|\theta x\| + \|(1-\theta)y\| \\ =& \theta \|x\| + (1-\theta)\|y\| \end{aligned} \tag{40}$$

Hence, $f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$ so $f(x)$ is a convex function.

5.5 Convex Optimization

Suppose you have a convex function defined over a convex set C, and you want to find the minimum of the function over the set C.

What do you have? A convex optimization problem!

Let $f(x)$ be a convex function over R^d. Minimize $f(x)$ subject to $Ax = b$.

The domain D is the set of $x \in R^d$ such that $Ax = b$.

If $Ax = b$, and $Ay = b$, then $A(tx + (1-t)y) = b$. Thus D is a convex set.

If x and y are both in D, then the line segment joining x and y is entirely in D.

5.6 Why is convex optimization important?

Fundamental property of Convex optimization:

Any _local minimum of a convex function f over a convex set C **must** also be a global minimum of f over C.

6 Gradient Descent (2020/05/05)

6.1 Method of Steepest Descent

Let $x \in R^3,\ y \in R^3$. these are column vectors in R^3

$$\begin{aligned} f(x) =& f(x_1, x_2, x_3) \\ f(y) =& f(y_1, y_2, y_3) \\ G(y) =& G(y_1, y_2, y_3) \end{aligned} \tag{41}$$

$\nabla f(x)$ is a gradient vector. The convention is that the gradient is a **row** vector.

$G(y) = f(y) - \nabla f(x)y$

$$\begin{aligned} \nabla f(x) \equiv& (\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3}) \\ \nabla f(x)y =& \frac{\partial f}{\partial x_1} y_1 + \frac{\partial f}{\partial x_2} y_2 + \frac{\partial f}{\partial x_3} y_3 \\ =& \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \frac{\partial f}{\partial x_3} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \end{aligned} \tag{42}$$

6.1.1 Warm Up

$\nabla G(y) = \nabla[f(y) - \nabla f(x)y] = \nabla f(y) - \nabla f(x)$

We assume

$$f(x) - f(y) - \nabla f(y)(x - y) \leq \frac{b}{2}\|x - y\|_2^2$$

> This assumption drives from Taylor's Theorem where the Hessian Matrix (Matrix of 2ND Derivatives) is bounded by the largest Eigenvalue.

For any given x, consider the function

$$G(y) = f(y) - \nabla f(x)y$$

G is convex.
$G(y) \equiv G_x(y)$ because G depends on x.
Suppose x is the minimizer of $G(y)$

$$G(x) \leq G(y - \frac{1}{b}\nabla G(y))$$

and

$$\nabla G(y) = \nabla[f(y) - \nabla f(x)y] = \nabla f(y) - \nabla f(x)$$

We assume $f(x)$ is C^1 and satisfies the condition:

$$\forall x,\ y,\ f(x) - f(y) \leq \nabla f(y)(x - y) + \frac{b}{2}\|x - y\|_2^2$$

C^1: continuously differentiable. What happens y = x?

$$\nabla G(x) = \nabla f(x) - \nabla f(x) = 0$$

meaning that its a minimum, which is a global minimum because G is converse. Thus explaining why x is a minimizer?

$G(y - a) - G(y)$
Let $x = y - a, a = \frac{1}{b}\nabla G(y)$

When making an assumption, make an assumption that allows you to learn something interesting.

$$\begin{aligned} &\leq \nabla G(y)(x - y) + \frac{b}{2}\|x - y\|_2^2 \\ &= \nabla G(y)(-a) + \frac{b}{2}\|x - y\|_2^2 \\ &= \nabla G(y)(-\frac{1}{b}\nabla G(y)^T) + \frac{b}{2}\ \frac{1}{b^2}\|\nabla G(y)\|_2^2 \end{aligned} \tag{43}$$

We just demonstrated

$$\begin{aligned} &G(y - \frac{1}{b}\nabla G(y)) - G(y) \\ &\leq \nabla G(y)(-\frac{1}{b}\nabla G(y)^T) + \frac{b}{2}\ \frac{1}{b^2}\|\nabla G(y)\|_2^2 \end{aligned} \tag{44}$$

6.1.2 Proving Gradient Descent

$$\nabla G(y) = \nabla[f(y) - \nabla f(x)y] = \nabla f(y) - \nabla f(x)$$

$$\begin{aligned}
&\rightarrow f(x) - f(y) - \nabla f(x)(x-y) && \text{(45a)}\\
&= f(x) - \nabla f(x)x - (f(y) - \nabla f(x)y) && \text{(45b)}\\
&= G(x) - G(y) && \text{(45c)}\\
&= G(y - \frac{1}{b}\nabla G(y)) - G(y) && \text{(45d)}\\
&\leq \nabla G(y)(-\frac{1}{b}\nabla G(y)^T) + \frac{b}{2}\ \frac{1}{b^2}\|\nabla G(y)\|_2^2 && \text{(45e)}\\
&= -\frac{1}{2b}\|\nabla G(y)\|_2^2 && \text{(45f)}\\
&= -\frac{1}{2b}\|\nabla f(x) - \nabla f(y)\|_2^2 && \text{(45g)}
\end{aligned}$$

[g] says

$$f(x) - f(y) - \nabla f(x)(x-y) \leq -\frac{1}{2b}\|\nabla f(x) - \nabla f(y)\|_2^2$$

We define a sequence of vectors

$$x_{k+1} = x_k - \frac{1}{b}g_k$$

$x_{k+1} = x_k - \frac{1}{b}\nabla f(x_k)$

Using $\frac{1}{b}$ is **Bold**. The old style updated the step at each iteration which results in less iterations but more compute.

$h = \frac{1}{b}$

Let us write

$$\begin{aligned}
d_k &= x_k - x^* \\
\delta_k &= f(x_k) - f(x^*)
\end{aligned} \tag{46}$$

How far the current estimate is from the minimum

Actual Error

Thus,

$$d_{k+1} = x_{k+1} - x^*$$

Apply [g] with $x = x_k,\ y = x^*$

$$\begin{aligned} &f(x_k) - f(x^*) - g_k^T(x_k - x^*) \leq -\frac{1}{2b}\|\nabla f(x_k) - \nabla f(x^*)\|_2^2 \\ &\rightarrow \delta_k \leq g_k^T d_k - \frac{1}{2b}\|g_k\|_2^2 \end{aligned} \tag{47}$$

because $g_k = \nabla f(x_k)$ and $d_k = x - x^*$
G: scalar everything else: vector
Look Closer!

$$x_{k+1} - x_k = -\frac{1}{b}g_k$$

<= Using $x_{k+1} - \frac{1}{b}g_k$
$g_k = -b(x_{k+1} - x_k)$

$$\begin{aligned} \delta_k \leq & g_k^T d_k - \frac{1}{2b}\|g_k\|_2^2 && \text{(48a)} \\ = & -b(x_{k+1} - x_k)^T d_k - \frac{b}{2}\|x_{k+1} - x_k\|_2^2 && \text{(48b)} \\ = & -\frac{b}{2}(\|x_{k+1} - x_k\|_2^2 + 2(x_{k+1} - x_k)^T d_k) && \text{(48c)} \\ = & -\frac{b}{2}(\|d_{k+1} - d_k\|_2^2 + 2(d_{k+1} - d_k)^T d_k) && \text{(48d)} \\ = & \|d_{k+1} - d_k\|_2^2 + 2(d_{k+1} - d_k)^T d_k && \text{(48e)} \\ = & (\langle d_{k+1}, d_{k+1}\rangle - 2\langle d_{k+1}, d_k\rangle + \langle d_k, d_k\rangle) + (2d_{k+1}^T d_k - 2d_k^T d_k) && \text{(48f)} \\ = & (\langle d_{k+1}, d_{k+1}\rangle - 2\langle d_{k+1}, d_k\rangle + \langle d_k, d_k\rangle) + (2\langle d_{k+1}, d_k\rangle - 2\langle d_k, d_k\rangle) && \text{(48g)} \\ = & -\frac{b}{2}(\langle d_{k+1}, d_{k+1}\rangle - \langle d_k, d_k\rangle) && \text{(48h)} \\ = & \frac{b}{2}(\|d_k\|_2^2 + \|d_{k+1}\|_2^2) && \text{(48i)} \\ & && \text{(48j)} \end{aligned}$$

To summarize,

$$\delta_k \leq \frac{b}{2}(\|d_k\|_2^2 - \|d_{k+1}\|_2^2)$$

$$\sum_{i=1}^{n} \delta_i \leq \frac{b}{2}(\|d_0\|_2^2 - \|d_n\|_2^2 \leq \frac{b}{2}\|d_0\|_2^2$$

What do we know about convergent series?
If $\sum_{k=1}^{\infty} \delta_k$ is convergent, then $\delta_k \to 0$ as $k \to \infty$

6.2 Global Convergence

Start with any x_0. We define the sequence of vectors

$$x_{k+1} = x_k - \frac{1}{b} g_k$$

$x_{k+1} = x_k - \frac{1}{b}\nabla f(x_k)$
Then, $f(x_k) - f(x^*) \to 0$ as $k \to \infty$
We can pick N as large as we want,

$$\sum_{k=0}^{N} \delta_k \leq \frac{b}{2}\|d_0\|_2^2$$

Recall that $g_k \equiv \nabla f(x_k)$ and $g_{k+1} \equiv \nabla f(x_{k+1})$
We can also show that $\|g_{k+1}\| \leq \|g_k\|$
The length of the gradient vectors are monotone decreasing.
We've shown that

$$f(x) - f(y) - \nabla f(x)(x-y) \leq -\frac{1}{2b}\|\nabla f(x) - \nabla f(y)\|_2^2$$

Similarly,

$$f(y) - f(x) - \nabla f(y)(y-x) \leq -\frac{1}{2b}\|\nabla f(x) - \nabla f(y)\|_2^2$$

Summing the above inequalities yields

$$-\nabla f(x)(x-y) - \nabla f(y)(y-x) \leq -\frac{1}{b}\|\nabla f(x) - \nabla f(y)\|_2^2$$

which means,

$$(\nabla f(x) - \nabla f(y))(x-y) \geq \frac{1}{b}\|\nabla f(x) - \nabla f(y)\|_2^2 \quad **$$

Let $x = x_{k+1},\ y = x_k$. Then, from (**),

$$(x_{k+1} - x_k)^T (g_{k+1} - g_k) \geq \frac{1}{b}\|g_{k+1} - g_k\|_2^2$$

But $x_{k+1} = x_k - \frac{1}{b} g_k$ so that

$$-\frac{1}{b}(g_k)^T(g_{k+1}-g_k) \geq \frac{1}{b}\|g_{k+1}-g_k\|_2^2$$

$$\begin{aligned}
-\frac{1}{b}(g_k)^T(g_{k+1}-g_k) &\geq \frac{1}{b}\|g_{k+1}-g_k\|_2^2 && (49a)\\
-(g_k)^T(g_{k+1}-g_k) &\geq \|g_{k+1}-g_k\|_2^2 && (49b)\\
-g_k^T g_{k+1} + g_k^T g_k) &\geq \|g_{k+1}-g_k\|_2^2 && (49c)\\
\langle g_k, g_k\rangle - \langle g_k, g_{k+1}\rangle &\geq \|g_{k+1}-g_k\|_2^2 && (49d)\\
\langle g_k, g_k\rangle - \langle g_k, g_{k+1}\rangle &\geq \langle g_{k+1}, g_{k+1}\rangle - 2\langle g_{k+1}, g_k\rangle + \langle g_k, g_k\rangle && (49e)\\
\langle g_k, g_{k+1}\rangle &\geq \langle g_{k+1}, g_{k+1}\rangle && (49f)\\
\|g_{k+1}\|_2^2 &\leq g_{k+1}^T g_k && (49g)
\end{aligned}$$

$$\begin{aligned}
\|g_{k+1}\|_2^2 &\leq g_{k+1}^T g_k && (50a)\\
&\leq \|g_{k+1}\|\|g_k\| \qquad \text{By Cauchy-Schwartz} && (50b)
\end{aligned}$$

That means, $\|g_{k+1}\| \leq \|g_k\|$, which is the desired conclusion

6.3 About Gradient Descent

Gradient Descent is *not* a single method. It is a large collection of methods.

1. Steepest Descent with a constant step size

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

2. Use a different step size at each iteration

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

6.3.1 Example

Select α_k to minimize $f(x_k - d_k g_k)$, where $g_k = \nabla f(x_k)$. Lots of algorithms to choose α_k

We assume $f(x)$ is C^1 and satisfies

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

If we assume f is convex, differentiable, and its gradient vector satisfies the Lipshitz Condition

$$\|\nabla f(x) - \nabla f(y)\| \leq b\|x - y\|$$

for any two points $x,\ y$, then the condition (*) is true.

6.4 Challenge

We have already demonstrated

$$\sum_{i=1}^{100} \delta_i \leq \frac{b}{2}\|d_0\|_2^2$$

and $\|g_{k+1}\| \leq \|g_k\|$. Our notation is $\delta_k = f(x_k) - f(x^*)$
You can show that the rate of convergence is given by

$$\delta_k \leq (\frac{1}{k+1})\frac{b}{2}\|d_0\|_2^2$$

TODO: Prove this out.

7 Lagrangian Multipliers (2020/05/12)

7.1 Prelude

Find MAX $x^2 + y^2$ subject to $x + y = 4$
Increase radius until it hits the slow of $x + y = 4$

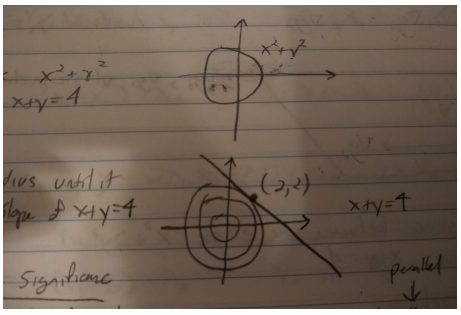


Figure 1: Prelude Drawgin

7.1.1 Geometric Significance

A $(x, y) = (2, 2)$ where MAX occurs: $\nabla f // \nabla g$
$f(x, y) = x^2 + y^2$; $\nabla f = (2x, 2y)$
$g(x, y) = x + y - 4 = 0$;
\$ ∇ g = (1, 1)\$

$$\begin{aligned} \nabla f =& \lambda \nabla g \\ \text{another way of saying parallel} & \\ =& (2x, 2y) = (4, 4) \end{aligned} \tag{51}$$

7.2 Lagrange Multipliers

with Several inequality constraints

Karush Kahn Tucker

Goal: Get the background to understand Lagrange Duality
Idea: Find a MAX or MIN of $f(x_1, x_2, y_1, y_2)$ subject to 3 requirements (constraints)

$$\begin{aligned} g_1(x_1, x_2, y_1, y_2) =& 0 \\ g_2(x_1, x_2, y_1, y_2) =& 0 \end{aligned} \tag{52}$$

- $f(\cdot)$ can have any number of variables
- can be subject to any constraint

Famous application in ML: SNMF (Semi-nonnegative Matrix Factorization)

7.2.1 Geometric Condition

The gradient of f is a Linear Combination of the gradients of g_1 and g_2. The number of λ = number of constraints.

$$\nabla f = \lambda_1 \nabla g_1 + \lambda_2 \nabla g_2$$

\$ ∇ g_1, \ ∇ g_2, \ ∇ f\$ lie in the same plane.

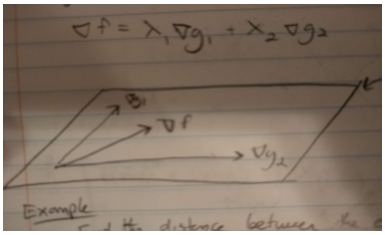


Figure 2: Hyperplane of Gradients and vector function

1. Example

   Find the distance between the ellipse $x^2+2y^2=1$ and the line $x+y=4$.

   Main Idea of the Solution

   Let (x_1, y_1) be any point on the ellipse and (x_2, y_2) be on any point on the line.

$$min\ d^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$$

   ^f

   subject to

$$x_1^2 + 2y_1^2 = 1,\ x_2 + y_2 = 4$$

   Setting: To find MIN of f, subject to $g_1 = 0$ and $g_2 = 0$ where

$$g_1 = x_1^2 + 2y_1^2 - 1,\ g_2 = x_2 + y_2 - 4$$

   Strategy: Let $F = f - \lambda_1 g_1 - \lambda_2 g_2$ where F is the **Lagrangian**

$$\begin{aligned} \nabla F =& \nabla f - \lambda_1 \nabla g_1 - \lambda_2 \nabla g_2 \\ 0 =& \nabla f - \lambda_1 \nabla g_1 - \lambda_2 \nabla g_2 \\ \nabla f =& \lambda_1 \nabla g_1 + \lambda_2 \nabla g_2 \end{aligned} \tag{53}$$

   Let

$$F = \frac{1}{2}[(x_1 - x_2)^2 + (y_1 - y_2)^2] - \frac{\lambda_1}{2}(x_1^2 + 2y_1^2 - 1) - \lambda_2(x_2 + y_2 - 4)$$

Take all partial derivatives. set $\nabla F = \vec{0}$

$$\frac{\partial F}{\partial x_1} = (x_1 - x_2) - \lambda_1 x_1 \to x_1 - x_2 = \lambda_1 x_1 \tag{54a}$$
$$\frac{\partial F}{\partial y_1} = (y_1 - y_2) - 2\lambda_1 y_1 \to y_1 - y_2 = 2\lambda_1 y_1 \tag{54b}$$
$$\frac{\partial F}{\partial x_2} = -(x_1 - x_2) - \lambda_2 \to x_2 - x_1 = \lambda_2 \tag{54c}$$
$$\frac{\partial F}{\partial y_2} = -(y_1 - y_2) - 2\lambda_2 \to y_2 - y_1 = \lambda_2 \tag{54d}$$
$$\tag{54e}$$

$$\lambda_2 = -\lambda_1 x_1, \ \lambda_2 = -2\lambda_1 y_1$$

(1)(3), (2)(4)

> $\lambda_1 \neq 0$. If $\lambda_1 = 0$, then $x_1 = x_2$ which means the ellipse and the line touch (which they don't). There is no common intersection point.

From (1), $\lambda \neq 0$, therefore $x_1 = 2y_1$

Since $x_1^2 + 2y_1^2 = 1$ and (x_1, y_1) is in the first quadrant, so using $x_1 = 2y_1$.

$$(x_1, y_1) = (\frac{2}{\sqrt{6}}, \ \frac{1}{\sqrt{6}})$$

Using (3)(4) to solve for (x_2, y_2).

Once we have (x_1, y_1), (x_2, y_2), compute $s^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$. The distance between the ellipse and the line is the value of d.

$$F(x_1, x_2, y_1, y_2) = f - \lambda_1 g_1 - \lambda_2 g_2$$

Then set $\nabla F = \vec{0}$.

What are Lagrangian Multipliers doing? It turns a constrained optimization problem into an UNCONSTRAINED optimization problem.

Does this method work in general?

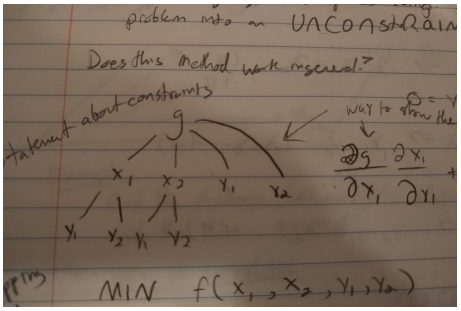


Figure 3: Breaking Down the Chain Rule

$\frac{\partial g}{\partial x_1}\frac{\partial x_2}{\partial y_1} + \frac{\partial g}{\partial x_2}\frac{\partial x_2}{\partial y_1} + \frac{\partial g}{\partial y_1} = 0$

C^1 mapping from $h_1 \to h_2$.

$\exists h_1,\ h_2$ such that

$$x_1 = h_1(y_1, y_2),\ x_2 = h_2(y_1, y_2)$$

Some function exists of y_1, y_2 called h_1, h_2. This comes from the **constraints**.

This is due to the **Implicit Function Theorem**. We never use f to determine h_1, h_2 which means we can have f be *anything*. It is important because we know f is a function y_1 and y_2 only, which means we only have to take derivatives of y_1 and y_2.

7.2.2 Explain Why Lagrange Multipliers Work

How do we know $\lambda_1,\ \lambda_2$ exist?

Optimization Problem: MIN $f(x_1, x_2, y_1, y_2)$ subject to constraints

$$\begin{aligned} g_1(x_1, x_2, y_1, y_2) &= 0 \\ g_2(x_1, x_2, y_1, y_2) &= 0 \end{aligned} \tag{55}$$

h_1 and h_2 are smooth as g_1 and g_2.

$\exists\ h_1,\ h_2$ such that $x_1 = h_1(y_1, y_2),\ x_2 = h_2(y_1, y_2)$

$$\begin{aligned} &\to g_1(h_1(y_1, y_2), h_2(y_1, y_2), y_1, y_2) = 0 \\ &\to g_2(h_1(y_1, y_2), h_2(y_1, y_2), y_1, y_2) = 0 \end{aligned} \tag{56}$$

Take partial derivatives with respect to y_1

- from g_1

$\frac{\partial g_1}{\partial y_1} + \frac{\partial h_1}{\partial y_1}\frac{\partial g_1}{\partial x_1} + \frac{\partial h_2}{\partial y_1}\frac{\partial g_1}{\partial x_2} = 0$

- from g_2, we get

$\frac{\partial g_2}{\partial y_1} + \frac{\partial h_1}{\partial y_1}\frac{\partial g_2}{\partial x_1} + \frac{\partial h_2}{\partial y_1}\frac{\partial g_2}{\partial x_2} = 0$

- to minimize $f(x_1, x_2, y_1, y_2)$

$\frac{\partial f}{\partial y_1} + \frac{\partial f}{\partial x_1}\frac{\partial h_1}{\partial y_1} + \frac{\partial f}{\partial x_2}\frac{\partial h_2}{\partial y_1} = 0$

$$\begin{bmatrix} \frac{\partial f}{\partial y_1} & \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \\ \frac{\partial g_1}{\partial y_1} & \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} \\ \frac{\partial g_2}{\partial y_1} & \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} \end{bmatrix} \begin{bmatrix} 1 \\ \frac{\partial h_1}{\partial y_1} \\ \frac{\partial h_2}{\partial y_1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} (*) \tag{57}$$

Let A be the 3×3 matrix on the left side. Suppose A^{-1} exists, then multiplying both side by A^{-1}.

$$\begin{bmatrix} 1 \\ \frac{\partial h_1}{\partial y_1} \\ \frac{\partial h_2}{\partial y_1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{58}$$

Not True thus A^{-1} cannot exist.

If ∇g_1 and ∇g_2 are linearly independent, then the top row is a linear combination of ∇g_1 and ∇g_2. (Otherwise the matrix would be invertible).

As long as your constraints are linearly independent, then the function is a linear combo of the gradients of the constraints.

7.3 Application

Non-Negative Matrix Factorization

$B \geq 0$ denotes a matrix with non-negative entries.

We use the Frobenius Norm (Hilbert-Schmidt Norm)

$$\|B\|_F^2 = \Sigma_j \Sigma_k |B(j,k)^2|$$

Given an $m \times n$ non-negative matrix A, NMF is defined as

$$\min_{x \in R^{m \times r},\ Y \in R^{r \times m}} \|A - XY\|_F^2 \text{ s.t. } x \geq 0,\ Y \geq 0 \tag{59}$$

Where r is the parameter that controls the size of factors X, Y.

Let A be a 1000×1000 image. Each pixel is a nonnegative number from $0 \to 255$. We hope to discover the structure of A by writing $A = X_Y$ where X has 60 columns.

The column vectors $a_1, a_2, ..., a_{1000} \in R^{1000}$ belong to a 60-dimensional subspace. A has 1 million entries. X has 60,000 entries and Y has 60,000 entries.

Since A is a non-negative matrix, it makes sense that at least one of X and Y is non-negative.

$$a_1, a_2, a_3, ... \in Span\{x_1, x_2, ..., x_{60}\},\ \text{each}\ \ x_j \in R^{1000}$$

$a_5 = c_1 x_1 + c_2 x_2 + ... + c_{60} x_{60}$

This problem can be formulated as the following:

$$\min_{X,U \in R^{m \times r},\ Y,V \in R^{r \times m}} \|A - XY\|_F^2 \quad \text{s.t.} U = X,\ V = Y,\ \ x \geq 0,\ Y \geq 0 \tag{60}$$

Where we introduced artificial variables for matrices U,V.

We consider the **Augment Lagrangian** of the Problem.

"Augmented" means to increase in mathematical terms.

$$\mathcal{L} = \|A - XY\|_F^2 + \langle \Lambda, X - U \rangle + \langle \Phi, Y - V \rangle + \frac{\alpha}{2}\|X - U\|_F^2 + \frac{\beta}{2}\|Y - V\|_F^2$$

$\|A - XY\|_F^2$: Objective Function

$\Lambda,\ \Phi$: Lagrange Multipliers that are Matrices

Λ is the same size as X

Φ is the same size as Y

Inner product of A,B = $\langle A, B \rangle = Tr(A^T B)$

Remark: $\mathcal{L}$ is a function of entries X, Y, U, V.

It is possible to compute partial derivatives such as

$$\frac{\partial \mathcal{L}}{\partial x_{1,1}},\ \frac{\partial \mathcal{L}}{\partial x_{1,2}},\ \frac{\partial \mathcal{L}}{\partial y_{1,1}},\ \frac{\partial \mathcal{L}}{\partial y_{1,2}}$$

7.3.1 An Approach to NMF Using ADMM

Input: A $m \times n$ matrix A, Target Rank R.
Alternating Direction Method of Multipliers
Output: A $m \times r$ matrix U, an $r \times n$ matrix V.
K = 1, . . . , N

$$\begin{aligned} X_{k+1} =& (AY_k^T + \alpha U_k - \Lambda_k)(Y_k Y_k^T + \alpha I)^{-1} \\ Y_{k+1} =& (X_{k+1}^T X_{k+1} - \beta I)^{-1}(X_{k+1}^T \Lambda + \beta V_k - \Phi_k) \end{aligned} \tag{61}$$

Update U_{k+1} using X_{k+1} and Λ_k
Update V_{k+1} using Y_{k+1} and Φ_k
Update Λ_{k+1} and Φ_{k+1}

8 Lagrangian Multipliers & Optimal Margin Classifiers (2020/05/19)

8.1 Warm Up

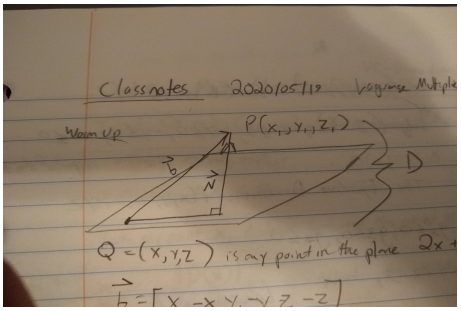


Figure 4: Warm Up Plane

$Q = (X, Y, Z)$ is any point in the plane $2x + 3y + 4z = 5$
$\vec{b} = [x, -x, y, -y, z, -z]$

$$
\begin{aligned}
D =& \frac{\|\vec{N} \cdot \vec{b}\|}{\|\vec{N}\|}, \ \vec{N} = [2,3,4] \\
=& \frac{\|2(x_1 - x) + 3(y_1 - y) + 4(z_1 - z)\|}{\|\vec{N}\|} \\
=& \frac{\|(2x_1 + 3y_1 + 4z_1) - (2x + 3y + 4z)\|}{\|\vec{N}\|} \\
=& \frac{\|(2x_1 + 3y_1 + 4z_1) - 5\|}{\|\vec{N}\|}
\end{aligned} \tag{62}
$$

Input: $(\vec{X}_1, Y_1), (\vec{X}_2, Y_2), (\vec{X}_3, Y_3)$

where $\begin{cases} Y_k = 1 & \vec{X}_k \in A \\ Y_k = -1 & \vec{X}_k \in B \end{cases}$

$\begin{cases} \vec{X} \in A & D(\vec{X}) > 0 \\ \vec{X} \in B & D(\vec{X}) \leq 0 \end{cases}$

$D(\vec{X}) = \vec{W} \cdot \vec{X}$

$D(X)$: The decision function

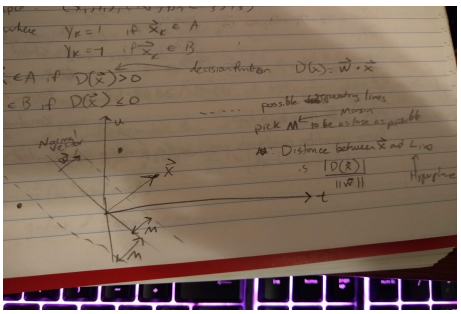


Figure 5: Warmup (cont)

8.2 Lagrange Multipliers

Example

Minimize xyz subject to $x + y + z \leq 3, \ -x \leq 0, \ -y \leq 0, -z \leq 0$

Outline of Main Idea: The objective function is $f(x, y, z) = xyz$

$$g_1 \leq 0, \ g_2 \leq 0, g_3 \leq 0, g_4 \leq 0$$

Constraints

$$\begin{aligned} g_1(x,y,z) =& x+y+z-3 \\ g_2(x,y,z) =& -x \\ g_3(x,y,z) =& -y \\ g_4(x,y,z) =& -z \end{aligned} \tag{63}$$

Lagrangian

$$F = f - \lambda_1 g_1 \lambda_2 g_2 - \lambda_3 g_3 - \lambda_4 g_4$$

$$F(x,y,z) = xyz = \lambda_1(x+y+z-3) + \lambda_2 x + \lambda_3 y + \lambda_4 z$$

Take all partial derivatives and set to 0

$$\begin{aligned} \frac{\partial F}{\partial x} =& yz - \lambda_1 + \lambda_2 = 0 \\ \frac{\partial F}{\partial y} =& xz - \lambda_1 + \lambda_3 = 0 \\ \frac{\partial F}{\partial z} =& yx - \lambda_1 + \lambda_4 = 0 \end{aligned} \tag{64}$$

The above constraints are know as the KKT condition. (Karush-Kuhn-Tucker)

More Constraints

The following constraints are derived from the Complementary Selectness Condition

$$\begin{aligned} \lambda_1(x+y+z-3) =& 0 \\ \lambda_2 X =& 0 \\ \lambda_3 Y =& 0 \\ \lambda_4 Z =& 0 \\ \lambda_i \geq& 0, \ i = 1,2,3,4 \end{aligned} \tag{65}$$

This is why the inequalities are stated as negative.

What is the main lesson here?

Having inequality constraints can make the problem complicated.

No implicit Function theorem is used to prove Lagrangian Multipliers with inequalities because it doesn't apply. The proof for why Lagrangian Multipliers work for inequality constraints is not present here.

8.3 Optimal Margin Classifiers (Vapnik)

Input: $(\vec{X}_1, Y_1), (\vec{X}_2, Y_2), (\vec{X}_3, Y_3)$

where $\begin{cases} Y_k = 1 & \vec{X}_k \in A \\ Y_k = -1 & \vec{X}_k \in B \end{cases}$

$\begin{cases} \vec{X} \in A & D(\vec{X}) > 0 \\ \vec{X} \in B & D(\vec{X}) \leq 0 \end{cases}$

$D(\vec{X}) = \vec{W} \cdot \vec{X}$

Pick any point x, distance between the point x and the Line (Separating Hyperplane)

A plane is the set of all points perpendicular to a normal vector.

$\frac{|D(x)|}{\|w\|}$

Desired State:

For $k = 1, 2, ..., p,\ Y_K = \{-1, 1\}$,

$$Y_k \frac{|D(x)|}{\|w\|} \geq M$$

Y_k assumes that we are treating distances as negative. This is just convention and may be dropped by other texts/sources.

Formulate our optimization problem:

$$\max_{w, \|w\|=1} M$$

Subject to,

$$Y_k D(X_k) \geq M,\ 1 \leq k \leq p$$

Insight: Maximizing the Margin M equivalent to

$$\min_w \|w\|$$

Subject to,

$$Y_k D(X_k) \geq 1,\ 1 \leq k \leq p$$

The maximum margin M is attained at $M^* = \frac{1}{\|W^*\|}$ where W^* is the optimal W in (8.3)

Reformulate with Lagrange Multipliers

$$\mathcal{L}(w, \lambda) = \frac{1}{2}\|W\|^2 - \sum_{k=1}^{p} \lambda_k [Y_k D(X_k) - 1]$$

minimize the square of norm. The gradient of $\frac{1}{2} norm^2 = w$

In $\mathcal{R}^2$, suppose $X_k = (t_k, u_k)$, then

$$D(X_k) = D(t_k, u_k) = w_1 t_k + w_2 u_k \rightarrow \frac{\partial D(X_k)}{\partial w_1} = t_k$$

$$\frac{\partial D(X_k)}{\partial w} \equiv (\frac{\partial D(X_k)}{\partial w_1}, \frac{D(X_k)}{\partial w_2})$$

$\frac{\partial D(X_k)}{\partial w_2} = u_k$

$$\frac{\partial \mathcal{L}}{\partial W} = W - \sum_{k=1}^{p} \lambda_k Y_k X_k = 0 \rightarrow W = \sum_{k=1}^{p} \lambda_k Y_k X_k$$

But $D(x) = W \cdot X$

$$\begin{aligned} D(x) =& (\sum_{k=1}^{p} \lambda_k Y_k X_k) \cdot X \ \leftarrow X_k \cdot X = \langle X_k, X \rangle \\ =& \sum_{k=1}^{p} \lambda_k Y_k \langle X_k, X \rangle \end{aligned} \tag{66}$$

Going back to the earlier example...
$(3x_1 + 17.2x_2 + 19.3X_3) \cdot X = 3\langle x_1, x \rangle + 17.2\langle x_2, x \rangle + 19.3\langle x_3, x \rangle$
Main Insight
The function $D(X)$ depends on x_1, x_2, x_3 only through $\langle x_1, x \rangle, \langle x_2, x \rangle, \langle x_3, x \rangle$
We don't care about the values in x_1, only the inner product of x_1 and x.

Engineering Principles

- All numbers = 5
- All functions are continuous
- All continuous functions are polynomials
- All polynomials are Linear

Why do we assume the decision function is a linear function? Now, $D(X)$ depends on x_1, x_2, x_3 through the inner products of X, So we pick a function $f(X)$ that depends only on the inner products with x: $\langle x_1, x\rangle, ..., \langle x_p, x\rangle$

Don't forget the inner product is symmetric. $\langle$ a, b $\rangle$ = $\langle$ b, a $\rangle$

$f(\bullet) = c_z K(\bullet, x_1) + ... + c_p K(\bullet, x_p)$
K is a Kernel function.
$K(x, x_1) = \langle x, x_1\rangle$
$K(x, x_1) = 1 + \langle x, x_1\rangle$ or $K(x, x_1)$ = a function of $\langle x, x_1\rangle$
Evaluating when p = 4
What happens at x_1 and x_2?

$$\begin{aligned} y_1 =& f(x_1) = c_1 K(x_1, x_1) + c_2 K(x_1, x_2) + c_3 K(x_1, x_3) + c_4 K(x_1, x_4) \\ y_2 =& f(x_2) = c_1 K(x_2, x_1) + c_2 K(x_2, x_2) + c_3 K(x_2, x_3) + c_4 K(x_2, x_4) \\ y_3 =& f(x_3) = c_1 K(x_3, x_1) + c_2 K(x_3, x_2) + c_3 K(x_3, x_3) + c_4 K(x_3, x_4) \\ y_4 =& f(x_4) = c_1 K(x_4, x_1) + c_2 K(x_4, x_2) + c_3 K(x_4, x_3) + c_4 K(x_4, x_4) \end{aligned} \tag{67}$$

$K(a, b) = (1 + \langle a, b\rangle)^4$
$y_1, y_2, y_3, y_4 \in \{-1, 1\}$
4 Linear equations in 4 Unknowns: c_1, c_2, c_3, c_4
$K(x_j, x_k) = (1 + \langle x_j, x_k\rangle)^4$
$Y = Ac$ where A is 4×4, $A(i, j) = K(X_j, x_k)$
We can solve for c_1, c_2, c_3, c_4 and obtain the function.
$f(\bullet) = c_z K(\bullet, x_1) + ... + c_p K(\bullet, x_p)$
When $\bullet$ is a new data point, we can predict whether it belongs to class A or B. If it is close to 1, it belongs to A. If it is close to -1, it is close to B.

8.4 Applications of ML

8.4.1 Handwritten Digit Review

28×28 handwritten digits.

$$B = [\vec{y}_1, ..., \vec{y}_{4000}],\ \vec{y}_j \in \mathbb{R}^{784}$$

- 800 images of 0, first 800 columns
- 800 images of 1, next 800 columns

and so forth.

Let $\vec{f}$ be a new image of digit 2. Solve for x such that $\vec{f} = B\vec{x}$. $\vec{x}$ is 20-sparse. Would like to only see non-zero entries at position 1601-2400 (where the twos are)

B has 784 rows, 4000 columns.

Important

Let $G^{500\times 784}$ be a Gaussian Matrix.

$$G\vec{f} = GB\vec{x}$$

Let $A = GB$ and $\vec{y} = G\vec{f}$

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

Solve for $\vec{x}$ by minimizing the \$l$_1$\$-norm.

1. Application

   Each trial of an experiment consists of training the machine with samples of digits from 2,3,4,5. The goal is to distinguish 5 from 2,3,4.

   One trial consists of the following steps:

   - Take 1000 images of 2, Randomly select 800 and set aside 200
   - Take 1000 images of 3, Randomly select 800 and set aside 200
   - Take 1000 images of 4, Randomly select 800 and set aside 200
   - Take 1000 images of 5, Randomly select 800 and set aside 200
   - Use the 3200 selected images for training
     $(x_1, y_1), ..., (x_p, y_p),\ p = 3200$
     each $y_k = 1$ if digit is 5, and $y_k = -1$ if digits are 2, 3, 4.

     $$f(\diamond) = c_1 K(\diamond, x_1) + ... + c_p K(\diamond, x_p)$$

     Use the polynomial kernel suggested by Vapnik et al.

     $$K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^4$$

   For the 200 images of 4 set aside for testing, we want to see $f(\diamond) \approx -1$
   For the 200 images of 5 set aside for testing, we want to see $f(\diamond) \approx 1$

8.5 Opinion

Use Deep NN when sample $> 60,000$
Use Kernel Method when sample ≤ 5000
Viewpoints Regarding Progress in ML

- We have made 0 progress since 1992 (Vapnik et. al paper on Kernel Methods)
- Made 1 progress (IHT)
- We made a lot of progress (LeCun @ Facebook)

Professor Opinion: Until there's a theory of non-convex optimization, there will not be an understanding of why DNN works.

9 Geometric Lemma (2020/06/02)

- Let A be any 4×4 matrix.
- Let $V \in \mathcal{R}^4$ so that $\|v\|_2 = 1$ and $\|Av\|_2$ is Max.

Pick any $W \in \mathcal{R}^4$ so that $w \perp v,\ \|w\|_2 = 1$
Then,

$$Av \perp Aw$$

$\perp$: orthogonal, perpendicular

The Vector v must be special. If you just have 2 vectors $v_1 \perp v_2$, then we cannot say $Av_1 \perp Av_2$.

Use the Geometric Lemma to prove SVD (Eckhart Young Theorem)
Let A be any 4×4 matrix. Then there are square matrices U, Σ, V s.t.

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

$U^T U = I,\ V^T V = I$: Columns are orthogonal to each other.
$\Sigma = diag(\sigma_1, \sigma_2, \sigma_3, \sigma_4)$ where σ_1 is the largest and σ_4 is the smallest.

9.1 Warm Up

Let $V \in \mathcal{R}^4$ so that $\|v\|_2 = 1$ and $\|Av\|_2$ is Max.

This means among all vectors v of norm 1, $\langle Av, Au\rangle$ is as large as possible.
Pick any $w \in \mathcal{R}^4$ so that $w \perp v,\ \|w\|_2 = 1$

Let $u = \cos(\theta)v + \sin(\theta)w$ and $f(\theta) = \langle Av, Au\rangle$

$$\begin{aligned} f(\theta) =& A(\cos(\theta)v + \sin(\theta)w)^T A\cos(\theta)v + A\sin(\theta) \\ =& \cos^2(\theta)\langle Av, Av\rangle + \sin^2(\theta)\langle Aw, Aw\rangle + 2\cos(\theta)\sin(\theta)\langle Av, Aw\rangle \\ =& \cos^2(\theta)\langle Av, Av\rangle + \sin^2(\theta)\langle Aw, Aw\rangle + \sin(2\theta)\langle Av, Aw\rangle \end{aligned} \tag{68}$$

Trig definition $2\cos(\theta)\sin(\theta) = \sin(2\theta)$

$$\begin{aligned} f'(\theta) =& -2\cos(\theta)sin(\theta)\langle Av, Av\rangle + 2\sin(\theta)\cos(\theta)\langle Aw, Aw\rangle + 2\cos(2\theta)\langle Av, Aw\rangle \\ =& 2\langle Av, Aw\rangle \end{aligned} \tag{69}$$

but $f'(0) = 0$ so $\langle Av, Aw\rangle = 0$

9.2 Proof

Use Geometric Lemma to prove the following:

> Let A be any 3×10 matrix. Then there is a 3×3 matrix U and a 10×3 matrix V s.t.
>
> $$Av = U\Sigma$$

Where

$$U^TU = V^TV = I, \Sigma = diag(\sigma_1, \sigma_2, \sigma_3) \text{ with } \sigma_1 \geq \sigma_2 \geq \sigma_3 \geq 0$$

$$A[\vec{v}_1|\vec{w}|\vec{x}] = [\vec{U}_1|\vec{U}_2|\vec{U}_3] \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{bmatrix}$$

9.3 Netflix Problem

Lowe Ran Matrix $M \in \mathcal{R}^{m \times N}$ with many missing entries

EX: Barbara is a 512 x 512 image. Randomly erase half the image. Try to recover the original image.

We observe those entries in the index set Ω (The set Ω locates our samples.)

We want to recover the Matrix M, including all the missing entries.

Use the nuclear norm,

$$\|X\|_* = \sum_{k=1}^{n} \sigma_k(x)$$

> The nuclear norm is the sum of singular values. i.e. Trace of Σ
> It comes from Alexander Grothendieck, the purest of the pure mathematicians.
>
> A small nuclear norm *roughly* approximates low-rank.

Convex Optimization Problem
minimize $ $\sum_{k=1}^{n} \sigma_k (x)$ $
subject to $X_{ij} = M_{ij} \forall (i,j) \in \Omega$

9.3.1 Fast Augmented Lagrangian Algorithm for learning Low-Ran Matrices

Given a rank-2 matrix B that is 50×50 with missing entries. In each iteration, keep track of two matrices, Z and Y. At the iteration K, update the matrix Z by

$$Z_{k+1} = \mathcal{D}_\lambda (Z_k + \alpha_k V_k)$$

λ is a param, α_k is a step size, Y_k is the direction of descent. In addition, the soft-threshold operator $\mathcal{D}_\lambda$ is:

$$\mathcal{D}_\lambda (z) = U max(S - \lambda I, 0) W^T$$

$Z = USV^T$ is the SVD of Z.

10 Special Topics: Bag-of-Words Topic Modeling (2020/06/05)

Philosophy

What is the biggest responsibility and burden I can take on in order to greatly improve the fulfillment of my life?

Thinking of equations is the wrong way to think about math. If you think of math in this way, you miss the point entirely

Assumptions

- Each document is assigned a single topic
- Words in a document are drawn independently

Use Case:

- Genetic pattern recognition

K: \ of topics
d: \ of distinct words in vocabulary
$c \geq 3$: \ of words in each document
In this case, Matrices are not powerful enough. We need tensors.

10.1 Tensors

Tensor of Order 0: Number (5, e, e^{π}, ...) Tensor of Order 1: Vector Tensor of Order 2: Matrix Tensor of Order 3: Stack of Pancakes

- "3d" matrix (crutch for initial understanding)

e^{π}: don't know how to show its irrational.

10.2 Generative Process for a Document

1. The topic of a doc is a R.V. K = 4.

$$P(m = j) = w_j, \ j \in \{1, 2, 3, 4\}$$

$\sum_1^K w_i = 1$ because its a probability.

2. Given topic h, l words are drawn independently $\mu_1 = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \\ 0.2 \\ 0 \end{bmatrix}$

μ_1: vector of probabilities for a given word appearing in a document
$\mu_1 = \mu_2 = \mu_3 = ... \in \mathcal{R}^d$ for each word in doc $v \in \{1, 2, ..., l\}$
Trying to find $\mathcal{R}^d$

$$P(\text{word } v \text{ is } i | h = j) = \langle \vec{e}_i, \vec{\mu}_j \rangle = M_{ij} \ 1 \leq i \leq d, \ j \in \{1, 2, 3, 4\}$$

$M = [\vec{\mu}_1 | \vec{\mu}_2 | \vec{\mu}_3 \vec{\mu}_4]$: matrix of conditional probabilities.

10.3 Two Ideas

Pairs (i, j) = P(first word = i, second word = j). Pairs is d x d.
Not a real way to solve because you need the documents.
Triples(i,j,k) = P(first word i, second word = j, third word = k)
Actual Way to think of a Tensor
A Linear Map. Triples: $\mathcal{R}^d \to \mathcal{R}^{d \times d}$ Takes a vector and returns a matrix.
Let Random Vectors $\vec{x}_1, \vec{x}_2, ... \in \mathcal{R}^d$ represent the l words by setting

$$\begin{aligned} \vec{x}_v &= \vec{e}_i \iff \text{word v is i, } 1 \leq i \leq d \\ x_3 &= e_3 \iff \text{third word in doc is 5th word in vocab} \end{aligned} \tag{70}$$

Each word is a coordinate vector.

Don't ask what a tensor is but what does it do?

Pairs = $E[\vec{x}_1 \otimes \vec{x}_2]$
Triples = $E[\vec{x}_1 \otimes \vec{x}_2 \otimes \vec{x}_3]$
$\otimes$: Tensor Product
defined by
Triples($\vec{y}$) = $E[(\vec{x}_1 \otimes \vec{x}_2)\langle \vec{y}, \vec{x}_3 \rangle]$
$\langle \vec{y}, \vec{x}_3 \rangle$: Returns a number.
if $\vec{y}(k)$ is the kth entry in $\vec{y}$, then the (i, j) entry of Triples($\vec{y}$) is
Triples($\vec{y}$) (i, j) = $\sum_{k=1}^{d} y(k)$ Triples (i, j, k)

Insight

$Pairs = M diag(\vec{w}) M^T$

$Triples = M diag(M^T \vec{y}) diag(\vec{w}) M^T$

10.4 Explanation for Pairs

$$\begin{aligned} Pairs(i,j) =& Pr(\vec{x}_1 = \vec{e}_i, \vec{x}_1 = \vec{e}_j | h = t) \cdot Pr(h = t) \\ =& \sum_{t=1}^{4} Pr(\vec{x}_1 = \vec{e}_i | h = t) \cdot Pr(\vec{x}_2 = \vec{e}_j | h = t) \cdot Pr(h = t) \\ =& \sum_{t=1}^{4} M(i,t)\ M(j,t) \cdot w_t \\ =& M diag(\vec{w})\ M^T \end{aligned} \tag{71}$$

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% compute the matrix Pairs
%%%% Pairs(i,j) = Pr[ first word is i, second word is j]
%%%%
%%%% Each column in a Shelf is a document.
%%%% The documents are the columns of a Shelf.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Pairs = zeros(dwords, dwords);
for ii=1:dwords
  for jj = 1:dwords
    count = 0;
    total = 0;
    for doc=1:Num
      total = total + 1;
      if (Shelf(1,doc) == ii) && (Shelf(2,doc) == jj)
        count = count + 1;
        end
    end
    Pairs(ii,jj) = count/total;
    end
end
```