PCA:

- Form the basis of many algos.
- Technique for dimensionality reduction.
- compared to the "JL-Lemma" (random projection):
  - Randomly picked vectors $r_1, \dots, r_k$
  - projected X on $r: \langle x_1, r_1 \rangle, \dots, \langle x_1, r_k \rangle$
    r_i's were not meaningful wrt S; picked randomly.
  - preserved the Euclidean dist btwn points.
  - In pratice, $k > 100$ for random projections to work.
  - For PCA, we can still choose $k = 2$.
  - PCA looks at S to come up with new representation
- High level goal of PCA is to find vectors $v_1, \dots, v_k$ st. $\forall x \in S$, $x \approx \sum_{j=1}^{k} a_j v_j$.
  (approx.)

- Note about preprocessing of S: Normalize each feature.
  - subtract the mean of each feature from each data point;
  - Normalize by the standard deviation of each feature, bc want features to be similiar range,

  To do so: $\forall$ feature j, compute:

$$\sigma_j \left(\begin{matrix} \text{STD of the} \\ j^{th} \text{ feature} \end{matrix} \right) = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \left(x_j^{(i)} \right)^2}$$

  Then Divide all the j^{th} feature by σ_j.

Summary: $x_j^{(i)} \leftarrow \dfrac{x_j^{(i)} - \mu_j}{\sigma_j}$, $\mu_j = \dfrac{1}{m} \sum_{i=1}^{m} x_j^{(i)}$

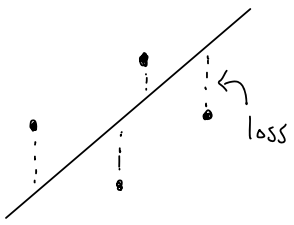
$\sigma_j^2 = \dfrac{1}{m} \left(x_j^{(i)} - \mu_j \right)^2$

How to begin:

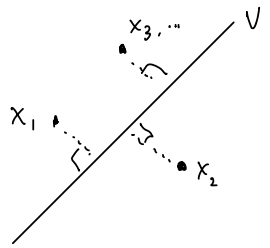
1) Find v_1 (first vector)

Look for a vector that minimizes square-distance:

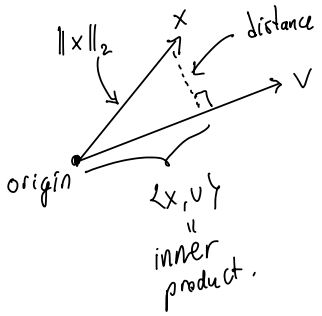
$$\min_{v,\ \|v\|_2 = 1} \frac{1}{m} \sum_{j=1}^{m} \left(\text{distance btwn } x_j \text{ and } v \right)^2$$



Regression



PCA



$$\underbrace{\langle x, v \rangle^2}_{a^2} + \underbrace{(\text{dist from } x \text{ to } v)^2}_{b^2} = \underbrace{\|x\|^2}_{c^2}$$

trying to find v s.t.:
- b is small
- a^2 is large.

c^2: fixed.

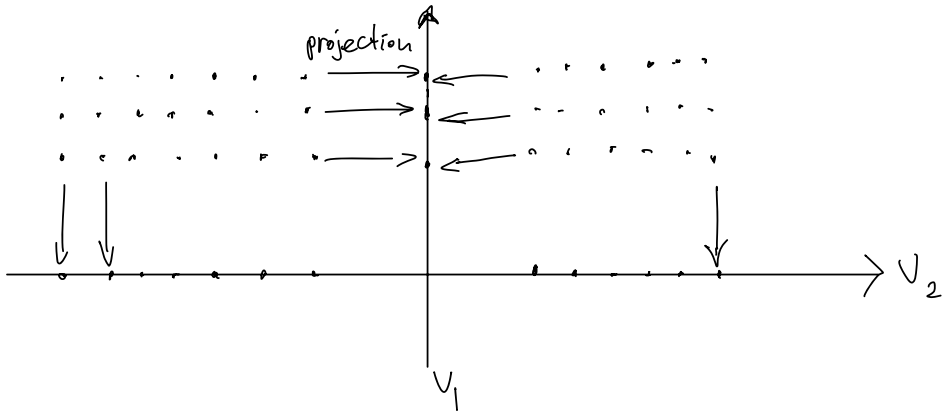
equivalently: want v that maximizes $\langle x, v \rangle^2$.

Why?: $b^2 = c^2 - a^2$. c^2 is fixed, so if max $a^2 = \langle x, v \rangle^2 \Rightarrow$ min b^2.

Find: $\max_{\|v\|_2 = 1} \frac{1}{m} \sum_{j=1}^{m} \underbrace{\langle x^i, v \rangle}_{\text{dot product}}^2$ ← direction of maximal variance

Remember: mean of the training set is 0, bc we subtracted the mean out of it.

$$\text{var}[X] = E[X^2] - \underbrace{[E[X]]^2}_{0}$$



which projection preserves the structure of the training set? V_2!

The above was for 1 vector. What about for k-vectors? (components)

$$\max_{\text{subspaces } S \text{ of dimension } k} \left[\frac{1}{m} \sum_{j=1}^{m} (\text{length of } x^j \text{ projected onto } S)^2 \right]$$

A really nice/preferable basis would be an orthonormal basis, $V_1, \dots, V_k$.

Note: Orthonormal basis = vectors are unit + orthogonal to one another.

$$(\text{distance from } X \text{ to } S)^2 = \underbrace{\|X\|^2}_{c^2} - \underbrace{\langle X_1, V_1 \rangle^2 + \dots + \langle X_k, V_k \rangle^2}_{a^2}$$

PCA Objective:

$$\max_{V_1, \dots, V_k \text{ orthogonal}} \frac{1}{m} \sum_{j=1}^{m} \sum_{i=1}^{k} \langle x^j, V_i \rangle^2$$

$$= \langle x^1 \cdot V_1 \rangle^2 + \langle x^1 \cdot V_2 \rangle^2 + \dots \langle x^1 \cdot V_k \rangle^2$$
$$+$$
$$\vdots$$
$$+$$
$$\langle x^m \cdot V_1 \rangle^2 + \langle x^m \cdot V_2 \rangle^2 + \dots + \langle x^m \cdot V_k \rangle^2$$

want orthogonal basis that:

- maximizes the variance along these directions that are orthogonal.

Assume we have $v_1, \ldots, v_k$, How to rewrite X?

$$X = \langle X_1 v_1 \rangle \cdot v_1 + \langle X_1 v_2 \rangle \cdot v_2 + \ldots + \langle X_j v_k \rangle \cdot v_k$$

(· = multiplication)

$= \sum$ projection onto the basis vectors.

$\Rightarrow X$ can be written as a vector in $\mathbb{R}^k$ corresponding to these projections.

Application 1: Understanding genomes:

Took 1400 people from Europe.
Each person was represented according to 200,000 genome markers in their genome.
Each person = vector of 200,000 features.
This corresponds to matrix of $1400 \times 200{,}000$.

- Ran PCA on this data to find vectors v_1 and v_2
- Each person corresponds to 2 numbers.
  They plotted these 2 numbers; color code each point according to country of origin.

Application 2:

Image Data Compression.
Strategy for compression data:
Each data point is an image (vector of pixels)
Each image has 65,000 pixels (65,000 features).
Images of faces. Run PCA on dataset.
$k = 100$ to 150.
Image = linear combination of 150 vectors of length 65,000

The Big Question:

How do we find these $v_1, \dots, v_k$'s?

$$\max_{\substack{v_1,\dots,v_k \\ \text{orthonormal}}} \left[\frac{1}{m} \sum_{j=1}^{m} \sum_{i=1}^{k} \langle x^j, v_i \rangle^2 \right]$$

Let X be an m by n matrix. $\frac{1}{m} X^T X$ (n by n matrix)

$v \longleftarrow$ a column vector $= \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$

$v^T \longleftarrow$ a row vector $= [\cdots]$.

$v^T v \longleftarrow$ inner product (scalar):

$$1\begin{bmatrix} \cdots \end{bmatrix}_k \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}_1 = 1\begin{bmatrix} \end{bmatrix}_1$$

$v v^T \longleftarrow$ outer product (matrix).

$$k\begin{bmatrix} \vdots \\ \vdots \end{bmatrix}_1 \begin{bmatrix} \cdots \end{bmatrix}_k = k\begin{bmatrix} \end{bmatrix}_k$$

sample covariance matrix. $(i,j)^{th}$ entry of $X^T X$ corresponds to "how similar is feature i to feature j?"

Note that $X^T X$ is a symmetric matrix. Proof: $X^T X$.

(i,j) = inner product of row i of X^T with column j of X.

= " " of column i of X with column j of X.

= " " of column j of X with " i of X.

Fact: all eigenvalues of symmetric matrices is ≥ 0.

For matrix A, vector v is an eigenvector if $A \cdot v = \lambda v$, $\lambda \in \mathbb{R}$.

(λ: eigenvalue.)

Definition: An orthogonal matrix is one where all columns are orthonormal; square matrix A is orthogonal if:

$$A^T A = I \text{ , and } AA^T = I$$

ex: $\begin{bmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 5 \end{bmatrix}$

Spectral theorem: Every symmetric matrix A has an eigendecomposition: $A = Q \cdot D \cdot Q^T$

(Q: orthogonal matrix; D: diagonal matrix)

$$D = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

entries of D are the eigenvalues of A.

Let's try to compute v_1.

Recall X is the matrix corresponding to S (training set).

X is m by n.

$$X \cdot v = \underset{m}{\begin{bmatrix} - & x_1 & - \\ - & x_2 & - \\ & \vdots & \\ - & x_m & - \end{bmatrix}} \underset{n}{\begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix}} \underset{1}{} = \underset{m}{\begin{bmatrix} \langle x_1, v \rangle \\ \vdots \\ \langle x_m, v \rangle \end{bmatrix}} \underset{1}{}$$

$$\Rightarrow (Xv)^T \cdot (Xv) = \sum_{i=1}^{m} \langle x_i, v \rangle^2$$

previously we stated:

$$\Updownarrow$$

$$v^T x^T x v$$

↑ find v that max this.

⇒ Equivalent: Find a V that maximizes $V^T \underbrace{(X^T X)}_{A} V$

"maximizing a quadratic form".

$\underset{V,\ \|V\|_2 = 1}{\text{maximize}} \left[V^T A V \right]$ "max quadratic form".

Let's look at a simple case: A is diagonal,

$$A = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \lambda_3 & \\ 0 & & & \ddots \lambda_n \end{bmatrix} \qquad \lambda_1 \geq \lambda_2 \geq \lambda_3 \cdots \lambda_n \geq 0.$$

If want to "max quadratic form":

pick $V = (1, \cdots, 0)$, since λ_1 is the largest, so put all the weight in it.

if A = diagonal

$$\Rightarrow V^T A V = (V_1, \cdots, V_n) \cdot \begin{pmatrix} \lambda V_1 \\ \vdots \\ \lambda_n V_n \end{pmatrix} = \sum_{i=1}^{n} V_i^2 \cdot \lambda_i$$

We don't know if A is diagonal in general.

But we do know: $A = Q \cdot D Q^T$ (spectral theorem)

A: symmetric; Q: orthogonal; D: diagonal.

⇒ A is "almost" diagonal;

Let $D = e_1 = (1, 0, 0, \cdots, 0)$

Choose $V = (Q \cdot e_1)$ ← maximizes the top eigenvector of A.

PCA Part II:

We studied the following optimization problem:

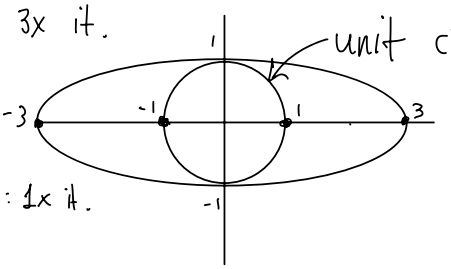
$$\max_{v,\ \|v\|_2=1} \left[v^T A v \right]$$

(matrix A corresponded to a covariance matrix $X^T X$ or $\frac{1}{m} X^T X$. — normalized.)

last time: we also discussed the "easy case" when A was a diagonal matrix.

$$\begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}$$

(3: x-axis; 3x it. 1: y-axis; 1x it.)



maps unit circle onto an ellipse.

Recall from linear algebra: rotational matrices,

These are orthogonal matrices, for example:

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$ = rotate θ degrees counterclockwise.

$$\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$ = rotate θ degrees clockwise,

(orthogonal matrices do not change the norms).

For Example; Remember A is diagonal.

$A = \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}$ $\Rightarrow$ solution to $\max_{v,\ \|v\|_2=1} \left[v^T \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix} v \right]$ is $v = (1, 0)$.

corresponds to choosing 3.

$$V^T A V = \sum_{i=1}^{n} v_i^2 d_i \quad , \quad \sum v_i^2 = 1.$$

A is diagonal

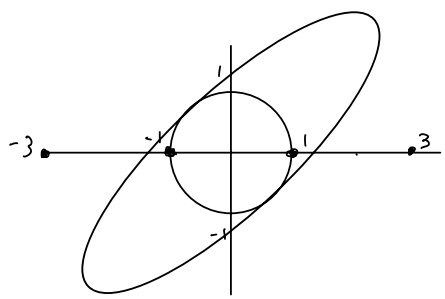
$\lambda_1, \lambda_2, \dots, \lambda_n \geq 0.$

Note: can write any covariance matrix (form: $X^T X$), as diagonal matrix · rotational matrix.

for example,

$$\underbrace{\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}}_{\text{covariane matrix. Symmetric.}} = \underbrace{\begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}}_{\text{rotates counterclockwise}} \cdot \underbrace{\begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}}_{\text{diagonal matrix. Stretching in X-axis.}} \cdot \underbrace{\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}}_{\text{rotates } 45^\circ \text{ clockwise.}}$$

$$A = X^T X = Q \quad D \quad Q^T$$



Spectral Theorem Revisited: X axis

Any symmetric matrix can be written as QDQ^T where Q is orthogonal and D is diagonal with real values on the diagonal, eigenvalues.

Furthermore: If $A = X^TX$ then all eigenvalues ≥ 0.

Claim 1: For any V, $V^TAV \geq 0$.

Because $A = X^TX$, $V^TAV = V^T(X^TX)V = V^TX^TXV$

$= \underbrace{(XV)^T \cdot XV}_{\text{inner product of self}} \geq 0$.

Claim 2: A cannot have negative eigenvalues (proof by contradiction).

Let's assume by contradiction that $\lambda_i < 0$, (i^{th} eigenvalue is negative).

Rewrite $\underbrace{A}_{\text{symmetric}} = \underbrace{QDQ^T}_{\text{spectral theorem}}$. Let's consider vector $Q \cdot e_i$,

, where $e_i = (0\ 0\ 0\ 0\ 0\ \underset{i^{th}}{1}\ \cdots\ 0\ 0)$,

$V = Q \cdot e_i$

Consider $V^TAV = \overbrace{(Qe_i)}^{V}{}^T \overbrace{(QDQ^T)}^{A} \overbrace{(Qe_i)}^{V}$

$= e_i^T \underbrace{Q^TQ}_{\text{Identity since orthogonal.}} D \underbrace{Q^TQ}_{\text{Identity since orthogonal.}} e_i = e_i^T \underbrace{De_i}_{\text{vector of 0 except 1 entry which is negative, in the } i^{th} \text{ position. e.g. } [0\ 0\ 0\ -1\ 0\ 0]} < 0$, which contradicts Claim 1.

Recap of PCA:

1) subtract the mean from your data.
2) Normalize the columns/features of your data.
3) compute eigenvalue/eigenvector decomposition of your matrix QDQ^T
4) The first k rows of Q^T are the k eigenvectors you're looking for.
   Top K principal components.

Prove that the i^{th} row of Q^T is an eigenvector of A:

i^{th} row of $Q^T = Q \cdot e_i$

$$\underbrace{A \cdot Q \cdot e_i}_{} = \overbrace{Q D \underbrace{Q^T Q}_{I}}^{A} \cdot e_i = \underbrace{Q D e_i}_{\text{diagonal matrix.}} = \underbrace{\lambda_i}_{} \underbrace{Q e_i}_{}$$

How do we compute this eigenvector/eigenvalue decomposition?

One method: "singular value decomposition" (SVD),

- Can use it to decompose, but expensive.
- Known; polynomial-time algorithm for computing SVD.

Another method; "power method".