Machine Learning for Data Science (CS4786) Lecture 5

Random Projections

Course Webpage:
http://www.cs.cornell.edu/Courses/cs4786/2016fa/
Which Direction to Pick?

PCA direction
Which Direction to Pick?

Direction has large covariance
Say \( w_1 \) and \( v_1 \) are the directions we choose to project in views 1 and 2 respectively we want these directions to maximize,

\[
\frac{1}{n} \sum_{t=1}^{n} (y_t[1] - \frac{1}{n} \sum_{t=1}^{n} y_t[1]) \cdot (y'_t[1] - \frac{1}{n} \sum_{t=1}^{n} y'_t[1]) \]

\[
\sqrt{\frac{1}{n} \sum_{t=1}^{n} (y_t[1] - \frac{1}{n} \sum_{t=1}^{n} y_t[1])^2} \sqrt{\frac{1}{n} \sum_{t=1}^{n} (y'_t[1] - \frac{1}{n} \sum_{t=1}^{n} y'_t[1])}
\]
Hence we want to solve for projection vectors $w_1$ and $v_1$ that

\[
\text{maximize } w_1^\top \Sigma_{1,2} v_1 \\
\text{subject to } w_1^\top \Sigma_{1,1} w_1 = v_1^\top \Sigma_{2,2} v_1 = 1
\]

\[
\begin{pmatrix}
\Sigma \\
\Sigma_{11} & \Sigma_{12} \\
\Sigma_{21} & \Sigma_{22}
\end{pmatrix}
= \text{COV}\left(\begin{pmatrix} X \\ X \end{pmatrix}ight)
\]
\[ W_1 = \text{eigs}(\Sigma_{11}^{-1} \Sigma_{12} \Sigma_{22}^{-1} \Sigma_{21}, K) \]

\[ W_2 = \text{eigs}(\Sigma_{22}^{-1} \Sigma_{21} \Sigma_{11}^{-1} \Sigma_{12}, K) \]
CCA DEMO
i can't believe how awful is this movie i was expecting it to be really good especially with the actors that were in the cast this is depressing i'm so bummed that they ruined such a good plot

bummed to see such a bad game what an awful performance by everyone on the team as if everyone played to loose need to improve hitters more but fielders were also worse today one of the worst performance in the history of baseball

oh man this war movie was just too depressing for me some scenes were simply awful even though the plot closely follows the novel which i've read i was bummed at the end and had to secretly go cry

i will tell you what is wrong with it it is dead that's what is wrong with it i had just about enough of this that team is definitely deceased tired and shagged out after a long game you say look matey not a single soul in that lineup would to hit a single ball even if i put 4000-volts through them they are bleeding demised they are not pitching they passed on period plus pretty sure they must smell of awful elderberries after that game you should be depressed like me and share the negativity please
this was so hilarious that's the best movie i've seen in a while i didn't know this actor before but he is so funny i was laughing from start to finish

it was hilarious to see playing these kids against experts throughout the game they were just running here and there and trying to get to the ball which they couldn't even once this was funny for viewers but organizers should ensure that inexperienced teams don't play against the experienced ones to keep the game interesting

dude that movie was so funny right i was laughing in like fits during some of the scenes i know the plot is supposed to be thought-provoking but i found it hilarious i really should stop laughing all the time but who cares right

now what seems to be the problem he says after leaning on the coach's limb body after a fast pitch struck him during the game his face was icy serious not laughing at all unlike everyone else jen said 'it is the coach he is not moving at all is he dead he said slowly course not we answered laughing again thank god
well that was a funny movie i enjoyed the plot with all those twists you never knew what was going to happen especially in this last scene i wasn't expecting this outcome at all haha

was it a game at all i felt as if everyone was just trying to stay warm by making as little move as possible laziness of fielders was making it appear as if they were running in 0.5x speed mode haha strikers made good use of pitch they got and it was an easy win

lol i can't even sit properly now i have a tummy ache because of all the rofling that actor's head looked like a volcano haha i swear it looked like it was about to erupt and his brains would spill out haha

fans at the game are encouraged to get out of their seats stretch a bit and sing take me out to the ball game that is the closest baseball gets to a halftime haha
really love that movie we saw yesterday. I was really excited since I knew it was going to be released this week and I haven't been disappointed at all. I especially enjoyed the acting of the actors; they were so good.

What an awesome game it was. Dwight Evans set the path to unprecedented victory when he made his very first strike on the pitch. He alone made the whole game enjoyable. Excited for the next match.

OMG, I totally loved yesterday's movie. We were all so excited to finally catch the third movie after months of scouring the fan pages for the plot. There are mixed opinions on the acting, but I think the actors did a brilliant job overall.

80 years old and was still playing the game. Stuff like this keeps you excited and motivated. You know, yes, he did break his back walking to the pitch to take the strike, but you know everyone has to expire and go to their maker at some point. He was lucky to do it while doing something he loved. I am sure he enjoyed every second of it. We should learn to enjoy this game too, like him, and reflect that on our strikes.
Recap
Given feature vectors $x_1, \ldots, x_n \in \mathbb{R}^d$, compress the data points into low dimensional representation $y_1, \ldots, y_n \in \mathbb{R}^K$ where $K << d$.

**Principal Component Analysis:**
- Find directions that maximize variance (spread)
- Find directions that minimize reconstruction error
Principal Component Analysis

1. $\Sigma = \text{cov}(X)$

2. $W = \text{eigs}(\Sigma, K)$

3. $Y = (X - \mu) \times W$
4. \( \hat{X} = Y \times W^T + \mu \)
Data can be split into pairs \((x_1, x_1'), \ldots, (x_n, x_n')\) where \(x_t's\) are \(d_1\) dimensional and \(x_t''s\) are \(d_2\) dimensional.

Goal: Compress \(x_1, \ldots, x_n\) into \(K\) dimensional vectors \(y_1, \ldots, y_n\) (or \(x_1', \ldots, x_n'\) into \(y_1', \ldots, y_n'\) or both).

- Retain information redundant between the two views.

**Canonical Correlation Analysis:**
- Find directions that maximize correlations between the projections in the two views.
CCA Algorithm

1. \( X \) = \( \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \)

2. \( \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} = \text{COV}(X) \)

3. \( W_1 = \text{eigs}(\Sigma_{11}^{-1}\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}, K) \)
Write $\tilde{x_t} = x_t x_t'$ the $d + d'$ dimensional concatenated vectors. Calculate covariance matrix of the joint data points $
abla = \nabla_1, 1 \nabla_1, 2 \nabla_2, 1 \nabla_2, 2$. The top $K$ eigen vectors of this matrix give us projection matrix for view I. Calculate $\nabla - 1, 2 \nabla_2, 1 \nabla - 1, 1 \nabla_1, 2 \nabla_1, 1$. The top $K$ eigen vectors of this matrix give us projection matrix for view II.

4. $Y_1 = X_1 - \mu_1 \times W_1$
back to single view: recap

\[
\begin{align*}
n & \quad X \\
\times & \quad d \ K \\
= & \quad n \ Y
\end{align*}
\]
The Tall, the Fat and the Ugly
The Tall, the Fat and the Ugly

\[
d \times n = d \sum
\]

\[
d W = \text{Eigs}(\sum, K)
\]
THE TALL, the Fat AND the Ugly

\[ X \]

\[ n \]

\[ d \]
The Tall, the Fat and the Ugly

\[ X \]

\[ \text{SVD}(X) \]

\[ n \times d \]

\[ U \times K \times V^T \]
THE TALL, THE FAT AND the Ugly

- $d$ and $n$ so large we can’t even store in memory
- Only have time to be linear in $\text{size}(X) = n \times d$

I there any hope?
$Y = X \times \begin{bmatrix} +1 & \ldots & -1 \\ -1 & \ldots & +1 \\ +1 & \ldots & -1 \\ \vdots \\ +1 & \ldots & -1 \\ K & \end{bmatrix} \frac{d}{\sqrt{K}}$
What does “it works” even mean?

Distances between all pairs of data-points in low dim. projection is roughly the same as their distances in the high dim. space.

That is, when $K$ is “large enough”, with “high probability”, for all pairs of data points $i, j \in \{1, \ldots, n\}$,

$$(1 - \epsilon) \|y_i - y_j\|_2 \leq \|x_i - x_j\|_2 \leq (1 + \epsilon) \|y_i - y_j\|_2$$
Why should Random Projections even work?!

Say $K = 1$. Consider any vector $\tilde{x} \in \mathbb{R}^d$ and let $\tilde{y} = \tilde{x} W$. Note that

$$\tilde{y}^2 = \left( \sum_{i=1}^{d} W[i, 1] \cdot \tilde{x}[i] \right)^2$$

$$= \sum_{i=1}^{d} (W[i, 1] \cdot \tilde{x}[i])^2 + 2 \sum_{i' > i} (W[i, 1] \cdot \tilde{x}[i]) (W[i', 1] \cdot \tilde{x}[i'])$$

$$= \sum_{i=1}^{d} W^2[i, 1] \tilde{x}^2[i] + \sum_{i' > i} (W[i, 1] \cdot W[i', 1]) \cdot (\tilde{x}[i] \cdot \tilde{x}[i'])$$

However $W^2[i, 1] = 1/K = 1$ when $K = 1$

$$= \sum_{i=1}^{d} \tilde{x}^2[i] + \sum_{i' > i} (W[i, 1] \cdot W[i', 1]) \cdot (\tilde{x}[i] \cdot \tilde{x}[i'])$$
Hence,

\[ \mathbb{E}[\tilde{y}^2] = \sum_{i=1}^{d} \tilde{x}^2[i] + \sum_{i' > i} \mathbb{E}[W[i, 1] \cdot W[i', 1]] \cdot (\tilde{x}[i] \cdot \tilde{x}[i']) \]

However \( W[i, 1] \) and \( W[i', 1] \) are independent and so

\[ \mathbb{E}[W[i, 1] \cdot W[i', 1]] = \mathbb{E}[W[i, 1]] \cdot \mathbb{E}[W[i', 1]] = 0 \]

Using this we conclude that

\[ \mathbb{E}[\tilde{y}^2] = \sum_{i=1}^{d} \tilde{x}^2[i] = \| \tilde{x} \|^2 \]
Why should Random Projections even work?!

Hence,

\[ \mathbb{E}[|\tilde{y}|^2] = \|\tilde{x}\|^2 \]

If we let \( \tilde{x} = x_s - x_t \) then

\[ \tilde{y} = \tilde{x}W = x_sW - x_tW = y_s - y_t \]

Hence for any \( s, t \in \{1, \ldots, n\} \),

\[ \mathbb{E}[|y_s - y_t|^2] = \|x_s - x_t\|^2 \]

Let's try this in Matlab …
Why should Random Projections even work?!

- Setting $K$ large is like getting $K$ samples.
- Specifically since we take $W$ to be random signs normalized by $\sqrt{K}$, for each $j \in [K]$, for any $\tilde{x}$ if $\tilde{y} = \tilde{x} W$, then

$$E[\tilde{y}^2[j]] = \|\tilde{x}\|_2^2 / K$$

Hence we can conclude that

$$E\left[\sum_{j=1}^{K} \tilde{y}^2[j]\right] = \sum_{j=1}^{K} E[\tilde{y}^2[j]] = \sum_{j=1}^{K} \frac{\|\tilde{x}\|_2^2}{K} = \|\tilde{x}\|_2^2$$

This is like taking an average of $K$ independent measurements whose expectations are $\|\tilde{x}\|_2^2$
Why should Random Projections even work?!

For large $K$, not only true in expectation but also with high probability

For any $\epsilon > 0$, if $K \approx \log(n/\delta)/\epsilon^2$, with probability $1 - \delta$ over draw of $W$, for all pairs of data points $i, j \in \{1, \ldots, n\}$,

$$(1 - \epsilon) \|y_i - y_j\|_2 \leq \|x_i - x_j\|_2 \leq (1 + \epsilon) \|y_i - y_j\|_2$$

Let's try on Matlab . . .

This is called the Johnson-Lindenstrauss lemma or JL lemma for short.
If we take $K = 69.1/\epsilon^2$, with probability 0.99 distances are preserved to accuracy $\epsilon$.
Why is this so Ridiculously Magical?

If we take $K = 69.1/\varepsilon^2$, with probability 0.99 distances are preserved to accuracy $\varepsilon$. 

$n = 1000$

$d = 10000$
If we take $K = 69.1/\varepsilon^2$, with probability 0.99 distances are preserved to accuracy $\varepsilon$