Methods for sparse analysis of high-dimensional data, II

Rachel Ward

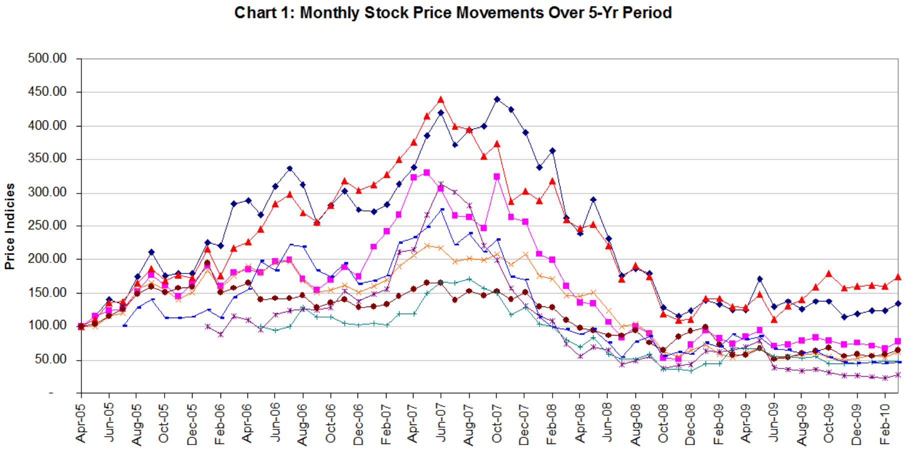
May 26, 2011

High dimensional data with low-dimensional structure

300 by 300 pixel images $= 90,000$ dimensions

High dimensional data with low-dimensional structure

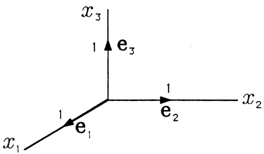
High dimensional data with low-dimensional structure

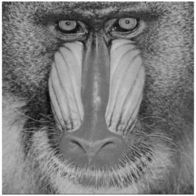


We need to recall some ...

- Euclidean geometry
- Statistics
- Linear algebra

Euclidean Geometry





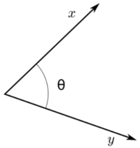
- An element of $\mathbb{R}^n$ is written
  $\mathbf{x} = \left(x_1, x_2, ..., x_n\right)$

- $\mathbb{R}^n$ is a vector space:
  - $\mathbf{x} + \mathbf{y} = \left(x_1 + y_1, x_2 + y_2, ..., x_n + y_n\right)$
  - $a\mathbf{x} = \left(ax_1, ax_2, ..., ax_n\right)$

- $\mathbf{x} = (x_1, x_2, ..., x_n) = \sum_{j=1}^{n} x_j \mathbf{e}_j$
  where

  $\mathbf{e}_1 = (1, 0, ..., 0), \quad \mathbf{e}_2 = (0, 1, ..., 0), ...$
  $\mathbf{e}_n = (0, 0, ..., 1)$

  are the standard basis vectors.



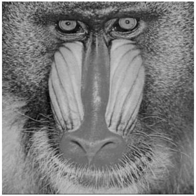
- The inner product between $\mathbf{x}$ and $\mathbf{y}$ is:
  $\langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + x_2 y_2 + \ldots + x_n y_n = \sum_{j=1}^{n} x_j y_j$

- $\|\mathbf{x}\| := \langle \mathbf{x}, \mathbf{x} \rangle^{1/2} = (x_1^2 + x_2^2 + \ldots + x_n^2)^{1/2}$ is the Euclidean length of $\mathbf{x}$. It is a norm:
  - $\|\mathbf{x}\| = 0$ if and only if $\mathbf{x} = 0$.
  - $\|a\mathbf{x}\| = |a|\|\mathbf{x}\|$
  - triangle inequality: $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$

- $\frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|x\|\|y\|} = \cos(\theta)$

- $\mathbf{x}$ and $\mathbf{y}$ are orthogonal (perpendicular) if and only if $\langle \mathbf{x}, \mathbf{y} \rangle = 0$

Statistics



$$\mathbf{x} = (x_1, x_2, x_3, \dots, x_n) \in \mathbb{R}^n$$

- Sample mean: $\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j$
- Standard deviation:

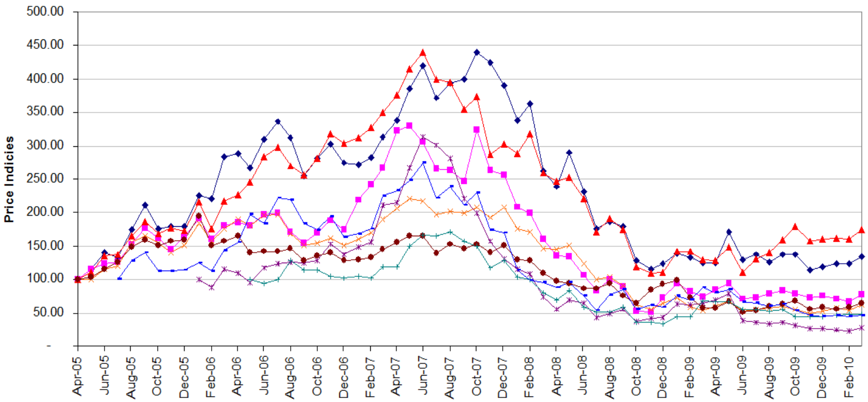
$$s = \sqrt{\frac{\sum_{j=1}^{n}(x_j - \bar{x})^2}{n-1}} = \frac{1}{\sqrt{n-1}}\sqrt{\langle \mathbf{x} - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle}$$

- Variance: $s^2 = \frac{1}{n-1}\langle \mathbf{x} - \bar{\mathbf{x}}, \mathbf{x} - \bar{\mathbf{x}} \rangle = \frac{1}{n-1}\|\mathbf{x} - \bar{\mathbf{x}}\|^2$
- Suppose we have p data vectors $\left\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\right\}$
  - Covariance: $\mathrm{Cov}(\mathbf{x}_j, \mathbf{x}_k) = \frac{1}{n-1}\langle \mathbf{x}_j - \bar{\mathbf{x}}_j, \mathbf{x}_k - \bar{\mathbf{x}}_k \rangle$
  - Covariance matrix for 3 data vectors $\left\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\right\}$:

$$\mathcal{C} = \begin{pmatrix} cov(\mathbf{x}_1, \mathbf{x}_1) & cov(\mathbf{x}_1, \mathbf{x}_2) & cov(\mathbf{x}_1, \mathbf{x}_3) \\ cov(\mathbf{x}_2, \mathbf{x}_1) & cov(\mathbf{x}_2, \mathbf{x}_2) & cov(\mathbf{x}_2, \mathbf{x}_3) \\ cov(\mathbf{x}_3, \mathbf{x}_1) & cov(\mathbf{x}_3, \mathbf{x}_2) & cov(\mathbf{x}_3, \mathbf{x}_3) \end{pmatrix}$$

- Covariance matrix for p data vectors has p columns and p rows

Chart 1: Monthly Stock Price Movements Over 5-Yr Period



What does the covariance matrix look like?

Linear Algebra

Eigenvectors

Suppose $\mathcal{A}$ is a $p \times p$ matrix. If $\mathcal{A}\mathbf{v} = \lambda\mathbf{v}$, then we say $\mathbf{v}$ is an eigenvector of $\mathcal{A}$ with eigenvalue λ.

Are these eigenvectors?

$$\mathcal{A} = \begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix}, \qquad \mathbf{v} = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$$

$$\mathcal{A} = \begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix}, \qquad \mathbf{v} = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

- If $\mathbf{v}$ is an eigenvector of $\mathcal{A}$ with eigenvector λ, then $\alpha\mathbf{v}$ is also an eigenvector of $\mathcal{A}$ with eigenvector λ. We will always use the normalized eigenvector $\|\mathbf{v}\| = 1$.

- Any real-valued and symmetric matrix $\mathcal{C}$ has n eigenvectors $\left\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\right\}$ which form an orthonormal basis for $\mathbb{R}^n$ (a.k.a. rotated coordinate view).

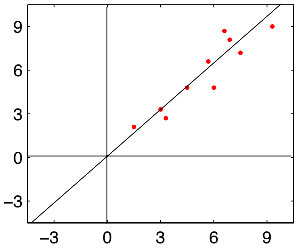
- Any $\mathbf{x} \in \mathbb{R}^n$ can be expressed in this basis via $\mathbf{x} = \sum_{j=1}^{n} \langle \mathbf{x}, \mathbf{v}_j \rangle \mathbf{v}_j$.

- $\mathcal{C}\mathbf{x} = \sum_{j=1}^{n} \lambda_j \langle \mathbf{x}, \mathbf{v}_j \rangle \mathbf{v}_j$

- $\mathcal{C} = \mathcal{P}\mathcal{D}\mathcal{P}^{-1}$ is diagonalizable:

$$\mathcal{P} = \begin{bmatrix} --- & \mathbf{v}_1 & --- \\ --- & \mathbf{v}_2 & --- \\ & \vdots & \\ --- & \mathbf{v}_n & --- \end{bmatrix}, \quad \mathcal{D} = \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & & \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix}$$

Example



$$\mathbf{x} = \big(7.5,\ 1.5,\ 6.6,\ 5.7,\ 9.3,\ 6.9,\ 6,\ 3,\ 4.5,\ 3.3\ \big),$$
$$\mathbf{y} = \big(7.2,\ 2.1,\ 8.7,\ 6.6,\ 9,\ 8.1,\ 4.8,\ 3.3,\ 4.8,\ 2.7\ \big)$$

$$cov(\mathbf{x},\mathbf{y}) = \frac{1}{n-1}\,\langle \mathbf{x} - \bar{\mathbf{x}}, \mathbf{y} - \bar{\mathbf{y}}\rangle\,,$$

$$\mathcal{C} = \begin{pmatrix} cov(\mathbf{x},\mathbf{x}) & cov(\mathbf{x},\mathbf{y}) \\ cov(\mathbf{x},\mathbf{y}) & cov(\mathbf{y},\mathbf{y}) \end{pmatrix} = \begin{pmatrix} 5.549 & 5.539 \\ 5.539 & 6.449 \end{pmatrix}$$

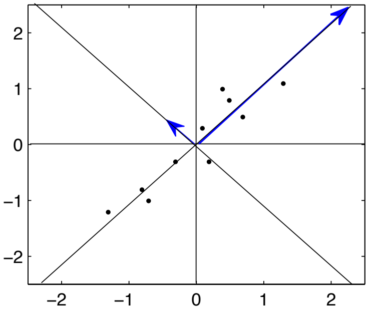


Figure: $\mathbf{x} - \bar{\mathbf{x}}$ *vs.* $\mathbf{y} - \bar{\mathbf{y}}$

Eigenvectors / values for $\mathcal{C}$:

- $\mathbf{v}_1 = \begin{pmatrix} .6780 \\ .7352 \end{pmatrix}, \lambda_1 = 11.5562$
- $\mathbf{v}_2 = \begin{pmatrix} -.7352 \\ .6780 \end{pmatrix}, \lambda_2 = .4418$

- $\mathbf{v}_1$ the first principal component of the data $(\mathbf{x}, \mathbf{y})$, and $\mathbf{v}_2$ the second 'principal component', and so-on ...
- Prove: $\mathbf{v_1}$ is in the direction of the 'least squares fit' to the centered data $\left(x_j - \bar{x}, y_j - \bar{y}\right), \quad j = 1, 2, ..., n.$

Principal component analysis

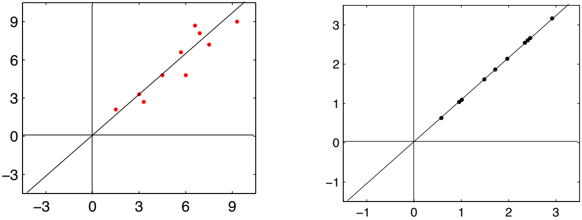


Figure: Original data and projection onto first principal component

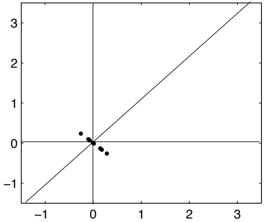
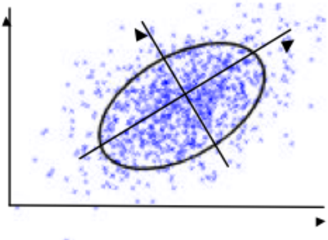


Figure: Residual

Principal component analysis



"Best fit ellipsoid" to the data

Principal component analysis

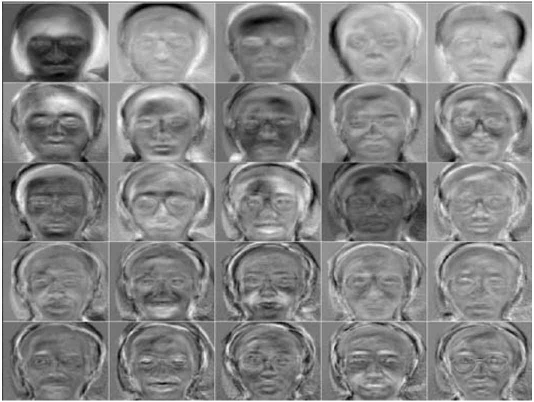
- The covariance matrix is written as $\mathcal{C} = \mathcal{P}\mathcal{D}\mathcal{P}^{-1}$, where

$$\mathcal{P} = \begin{bmatrix} --- & \mathbf{v}_1 & --- \\ --- & \mathbf{v}_2 & --- \\ & \vdots & \\ --- & \mathbf{v}_n & --- \end{bmatrix}, \quad \mathcal{D} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & & \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}$$

Suppose that $\mathcal{C}$ is $n \times n$ but $\lambda_{k+1} = \cdots = \lambda_n = 0$. Then the underlying data is low-rank

Suppose that $\mathcal{C}$ is $n \times n$ but λ_k through λ_n are very small. Then the underlying data is approximately low-rank.

Eigenfaces



The first few principal components (a.k.a. eigenvectors of the covariance matrix) for a database of many faces. Different components accentuate different facial characteristics

Eigenfaces

Top left face is projection of bottom right face onto its first principal component. Each new image from left to right corresponds to using 8 additional principal components for reconstruction

Eigenfaces

The projections of non-face images onto first few principal components

Fast principal component analysis

Randomized principal component analysis

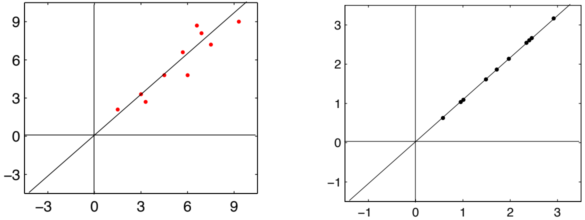


Figure: Original data and projection onto first principal component

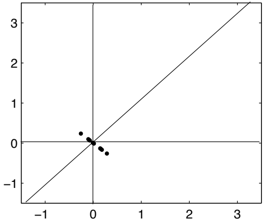


Figure: Residual

Inspiration: power iteration

- Suppose that $\mathcal{A}$ is an $n \times n$ real-valued matrix. Then $\mathcal{A}$ has n non-negative eigenvalues and n orthonormal eigenvectors, so $\mathcal{A}$ is diagonalizable:

$$\mathcal{A} = \mathcal{P}^{-1}\mathcal{D}\mathcal{P},$$

where

$$\mathcal{D} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & & \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}, \quad \mathcal{P} \quad = \quad \begin{bmatrix} - - - & \mathbf{v}_1 & - - - \\ - - - & \mathbf{v}_2 & - - - \\ & \vdots & \\ - - - & \mathbf{v}_n & - - - \end{bmatrix}$$

Inspiration: power iteration

- Suppose that $\mathcal{A}$ is an $n \times n$ real-valued matrix. Then $\mathcal{A}$ has n non-negative eigenvalues and n orthonormal eigenvectors, so $\mathcal{A}$ is diagonalizable:

$$\mathcal{A} = \mathcal{P}^{-1}\mathcal{D}\mathcal{P},$$

where

$$\mathcal{D} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & & \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}, \quad \mathcal{P} = \begin{bmatrix} --- & \mathbf{v}_1 & --- \\ --- & \mathbf{v}_2 & --- \\ & \vdots & \\ --- & \mathbf{v}_n & --- \end{bmatrix}$$

- Power iteration for computing $\mathbf{v}_1/\lambda_1$: a Gaussian random $\mathbf{x}_0$, and iterate $\mathbf{x}_{k+1} = \mathcal{A}\mathbf{x}_k/\|\mathcal{A}\mathbf{x}_k\|$. If $\lambda_1 > \lambda_2$, then $\mathbf{x}_k \to \mathbf{v}_1$.

Inspiration: power iteration

- Suppose that $\mathcal{A}$ is an $n \times n$ real-valued matrix. Then $\mathcal{A}$ has n non-negative eigenvalues and n orthonormal eigenvectors, so $\mathcal{A}$ is diagonalizable:

$$\mathcal{A} = \mathcal{P}^{-1}\mathcal{D}\mathcal{P},$$

where

$$\mathcal{D} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & & \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}, \quad \mathcal{P} = \begin{bmatrix} --- & \mathbf{v}_1 & --- \\ --- & \mathbf{v}_2 & --- \\ & \vdots & \\ --- & \mathbf{v}_n & --- \end{bmatrix}$$

- Power iteration for computing $\mathbf{v}_1/\lambda_1$: a Gaussian random $\mathbf{x}_0$, and iterate $\mathbf{x}_{k+1} = \mathcal{A}\mathbf{x}_k/\|\mathcal{A}\mathbf{x}_k\|$. If $\lambda_1 > \lambda_2$, then $\mathbf{x}_k \to \mathbf{v}_1$.
- Problem: unstable for computing subsequent eigenvalues

Stable randomized power iteration

Consider an $n \times n$ symmetric real matrix $\mathcal{A}$, and suppose we want to compute first k eigenvectors and eigenvalues.

1. Set $\ell = k + 10$. Using a random number generator, form an $n \times \ell$ matrix G whose entries are i.i.d. Gaussian random variables of zero mean and unit variance

Stable randomized power iteration

Consider an $n \times n$ symmetric real matrix $\mathcal{A}$, and suppose we want to compute first k eigenvectors and eigenvalues.

1. Set $\ell = k + 10$. Using a random number generator, form an $n \times \ell$ matrix G whose entries are i.i.d. Gaussian random variables of zero mean and unit variance
2. Form the $n \times \ell$ matrix $H^{(0)} = \mathcal{A}G$ and compute $Q^{(0)}$ whose columns form an orthonormal basis for the range of $H^{(0)}$.

Stable randomized power iteration

Consider an $n \times n$ symmetric real matrix $\mathcal{A}$, and suppose we want to compute first k eigenvectors and eigenvalues.

1. Set $\ell = k + 10$. Using a random number generator, form an $n \times \ell$ matrix G whose entries are i.i.d. Gaussian random variables of zero mean and unit variance
2. Form the $n \times \ell$ matrix $H^{(0)} = \mathcal{A}G$ and compute $Q^{(0)}$ whose columns form an orthonormal basis for the range of $H^{(0)}$.
3. Form $H^{(j)} = \mathcal{A}Q^{(j-1)}$ and compute $Q^{(j)}$ whose columns form an orthonormal basis for the range of $H^{(j)}$.

Stable randomized power iteration

Consider an $n \times n$ symmetric real matrix $\mathcal{A}$, and suppose we want to compute first k eigenvectors and eigenvalues.

1. Set $\ell = k + 10$. Using a random number generator, form an $n \times \ell$ matrix G whose entries are i.i.d. Gaussian random variables of zero mean and unit variance
2. Form the $n \times \ell$ matrix $H^{(0)} = \mathcal{A}G$ and compute $Q^{(0)}$ whose columns form an orthonormal basis for the range of $H^{(0)}$.
3. Form $H^{(j)} = \mathcal{A}Q^{(j-1)}$ and compute $Q^{(j)}$ whose columns form an orthonormal basis for the range of $H^{(j)}$.
4. Set $Q = Q^{(q)}$.

- **THEOREM:** With high probability, $\|\mathcal{A} - QQ^T\mathcal{A}\| \leq (nk)^{\frac{1}{2q}}\lambda_{k+1}$.

Stable randomized power iteration

Consider an $n \times n$ symmetric real matrix $\mathcal{A}$, and suppose we want to compute first k eigenvectors and eigenvalues.

1. Set $\ell = k + 10$. Using a random number generator, form an $n \times \ell$ matrix G whose entries are i.i.d. Gaussian random variables of zero mean and unit variance
2. Form the $n \times \ell$ matrix $H^{(0)} = \mathcal{A}G$ and compute $Q^{(0)}$ whose columns form an orthonormal basis for the range of $H^{(0)}$.
3. Form $H^{(j)} = \mathcal{A}Q^{(j-1)}$ and compute $Q^{(j)}$ whose columns form an orthonormal basis for the range of $H^{(j)}$.
4. Set $Q = Q^{(q)}$.

- **THEOREM:** With high probability, $\|\mathcal{A} - QQ^T\mathcal{A}\| \leq (nk)^{\frac{1}{2q}}\lambda_{k+1}$.
- Importance: Using a 'FFT'-based random matrix in place of G, this algorithm takes $O(n^2 \log(k))$ 'flops' to compute k eigenvalues / eigenvectors (Compare to standard $O(n^2k)$ flops)

Randomized Algorithm for PCA

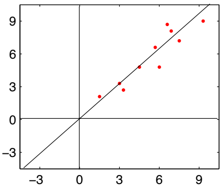
THEOREM: With high probability, $\|\mathcal{A} - QQ^T\mathcal{A}\| \le (nk)^{\frac{1}{2q}}\lambda_{k+1}$

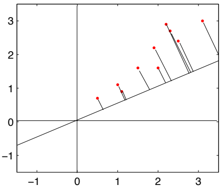
Why does $\|\mathcal{A} - QQ^T\mathcal{A}\| \approx \varepsilon$ give us eigenvalues and eigenvectors of $\mathcal{A}$?

- Form $\mathcal{B} = Q^T\mathcal{A}Q$.
- Diagonalize $\mathcal{B}$: $\quad \mathcal{B} = V\Lambda V^{-1}$
- Form $U = QV$. Then

$$
\begin{aligned}
\|\mathcal{A} - U\Lambda U^T\| &= \|\mathcal{A} - QV\Lambda V^{-1}Q^T\| \\
&= \|\mathcal{A} - QQ^T\mathcal{A}QQ^T\| \\
&\le \|\mathcal{A} - QQ^T\mathcal{A}\| + \|QQ^T\mathcal{A} - QQ^T\mathcal{A}QQ^T\| \\
&\le \varepsilon + \|QQ^T\|\varepsilon \\
&= 2\varepsilon
\end{aligned}
$$

Reducing dimensionality using random projections





Principal components:
Directions of projection are data-dependent

Random projections:
Directions of projection are *independent* of the data

Two situations where we use random projections::

1. Data is so high-dimensional that it is too expensive to compute principal components directly
2. You do not access to all the data at once, as in data streaming

Data streaming

- Massive amounts of data arrives in small time increments
- Often past data cannot be accumulated and stored, or when they can, access is expensive.

Data streaming

- $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ at time $(t_1, t_2, ..., t_n)$

Summary statistics that can be computed in one pass:

- Mean value: $\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j$
- Euclidean length: $\|\mathbf{x}\|^2 = \sum_{j=1}^{n} x_j^2$
- Variance: $\sigma^2(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^{n}(x_j - \bar{x})^2$

Data streaming

- $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ at time $(t_1, t_2, ..., t_n)$

Summary statistics that can be computed in one pass:

- Mean value: $\bar{x} = \frac{1}{n} \sum_{j=1}^{n} x_j$
- Euclidean length: $\|\mathbf{x}\|^2 = \sum_{j=1}^{n} x_j^2$
- Variance: $\sigma^2(\mathbf{x}) = \frac{1}{n} \sum_{j=1}^{n} (x_j - \bar{x})^2$

  Now we want to compare $\mathbf{x}$ to $\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n)$.
- The correlation $\left\langle \mathbf{x} - \bar{\mathbf{x}}, \tilde{\mathbf{x}} - \bar{\tilde{\mathbf{x}}} \right\rangle / \sigma(\mathbf{x})\sigma(\tilde{\mathbf{x}})$ is used to assess risk of stock $\mathbf{x}$ against market $\tilde{\mathbf{x}}$

Approach: introduce randomness

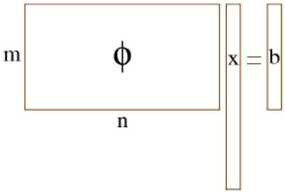
- Consider $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and vector $\varphi = (\varphi_1, \varphi_2, ..., \varphi_n)$ of (i.i.d.) unit normal Gaussian random variables:

$$\varphi_j \sim \mathcal{N}(0,1), \qquad \mathbb{P}\big(\varphi_j \geq x\big) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$$

- Consider

$$\begin{aligned} b &= \langle \varphi, \mathbf{x} \rangle - \langle \varphi, \tilde{\mathbf{x}} \rangle \\ &= \big(\varphi_1 x_1 + \varphi_2 x_2 + \cdots + \varphi_n x_n\big) - \big(\varphi_1 \tilde{x}_1 + \varphi_2 \tilde{x}_2 + \cdots + \varphi_n \tilde{x}_n\big) \\ &= \langle \varphi, \mathbf{x} - \tilde{\mathbf{x}} \rangle \end{aligned}$$

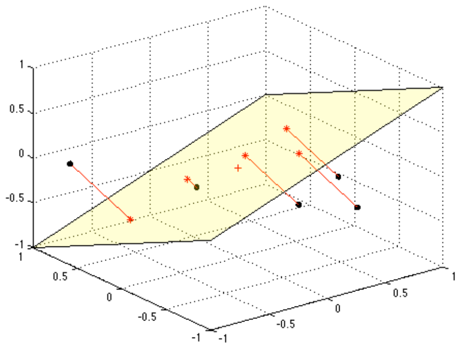
- Claim: $\mathbb{E} b^2 = \|\mathbf{x} - \tilde{\mathbf{x}}\|^2$



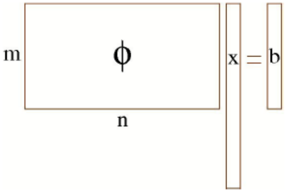
- More generally, for an $m \times N$ matrix Φ with i.i.d. Gaussian entries $\varphi_{i,j} \sim \mathcal{N}(0,1)$

$$\mathbb{E}\Big(\|b\|^2\Big) = \mathbb{E}(\|\frac{1}{\sqrt{m}}\Phi(\mathbf{x})\|^2) \quad = \quad \frac{1}{m}\mathbb{E}\Big(\sum_{i=1}^{m} \langle \varphi_i, \mathbf{x} \rangle^2\Big) = \|\mathbf{x}\|^2$$

Geometric intuition



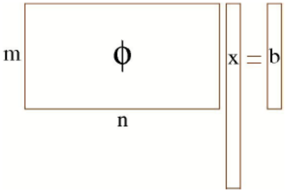
- The linear map $\mathbf{x} \to \frac{1}{\sqrt{m}}\Phi\mathbf{x}$ is similar to a *random projection* onto an m-dimensional subspace of $\mathbb{R}^n$
- *most* projections preserve geometry, but not all.



Concentration around expectation:

- For a fixed $\mathbf{x} \in \mathbb{R}^n$,

$$\mathbb{P}\Big(\|\frac{1}{\sqrt{m}}\Phi(\mathbf{x})\|^2 \geq (1+\varepsilon)\|\mathbf{x}\|^2\Big) \leq \exp\big(-\frac{m}{4}\varepsilon^2\big)$$



Concentration around expectation:

- For a fixed $\mathbf{x} \in \mathbb{R}^n$,

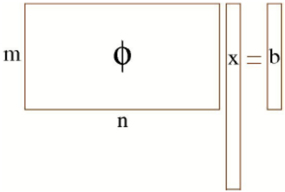
$$\mathbb{P}\Big(\|\frac{1}{\sqrt{m}}\Phi(\mathbf{x})\|^2 \geq (1+\varepsilon)\|\mathbf{x}\|^2 \Big) \leq \exp\big(-\frac{m}{4}\varepsilon^2 \big)$$

- For p vectors $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_p\}$ in $\mathbb{R}^n$

$$\mathbb{P}\Big(\exists \mathbf{x}_j : \ \|\frac{1}{\sqrt{m}}\Phi(\mathbf{x}_j)\|^2 \geq (1+\varepsilon)\|\mathbf{x}_j\|^2 \Big) \leq \exp\big(\log p - \frac{m}{4}\varepsilon^2 \big)$$

How small can m be such that this probability is still small?

Distance preservation of random matrices



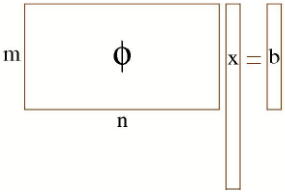
Theorem (Distance preservation)

Fix an accuracy $\varepsilon > 0$ *and probability of failure* $\eta > 0$. *Fix an integer* $m \geq c_{\varepsilon,\eta} \log(p)$, *and fix an* $m \times n$ *Gaussian random matrix* Φ.

Then with probability greater than $1 - \eta$,

$$\textit{For any fixed } \mathbf{x}: \quad \mathbb{P}\left(\left| \|\Phi \mathbf{x}\|^2 - \|\mathbf{x}\|^2 \right| \geq \varepsilon \|\mathbf{x}\|^2 \right) \leq 2e^{-c'_\varepsilon m}$$

for all j *and* k.



Corollary (Angle preservation)

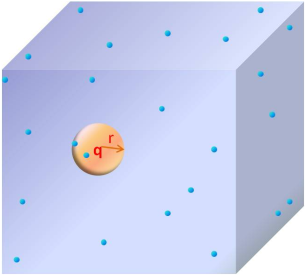
Fix an accuracy $\varepsilon > 0$ and probability of failure $\eta > 0$. Fix an integer $m \geq c_{\varepsilon,\eta} \log(p)$ and fix an $m \times n$ Gaussian random matrix Φ.

Then with probability greater than $1 - \eta$,

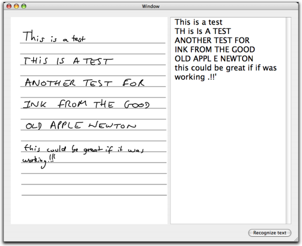
$$\left\| \frac{1}{m} \langle \Phi \mathbf{x}_j, \Phi \mathbf{x}_k \rangle - \langle \mathbf{x}_j, \mathbf{x}_k \rangle \right\| \leq \frac{\varepsilon}{2} (\|\mathbf{x}_j\|^2 + \|\mathbf{x}_k\|^2)$$

for all j and k.

The nearest-neighbors problem



- Find the closest point to a point **q** from among a set of points $S = \left\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\right\}$. Originally called the "post-office problem" (1973)



Similarity searching ...

The nearest-neighbors problem

- Find the closest point to a point $\mathbf{q}$ from among a set of points $S = \left\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\right\}$

-

$$\begin{aligned}
\mathbf{x}^* &= \arg\min_{\mathbf{x}_j \in S} \left\|\mathbf{q} - \mathbf{x}_j\right\|^2 \\
&= \arg\min_{\mathbf{x}_j \in S} \sum_{k=1}^{N} \left(q(k) - x_j(k)\right)^2
\end{aligned}$$

- Computational cost (number of 'flops') per search: $O(Np)$

- Computational cost of m searches: $O(Nmp)$.

- Curse of dimensionality: If N and p are large, this is a lot of flops!

The ε-approximate nearest-neighbors problem

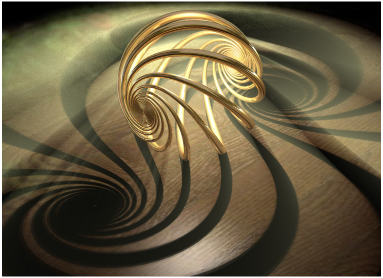
- Given a tolerance $\varepsilon > 0$, and a point $\mathbf{q} \in \mathbb{R}^N$, return a point $\mathbf{x}_\varepsilon^*$ from the set $S = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\}$ which is an ε-approximate nearest neighbor to $\mathbf{q}$:

$$\|\mathbf{q} - \mathbf{x}_\varepsilon^*\| \leq (1+\varepsilon)\|\mathbf{q} - \mathbf{x}^*\|$$

  This problem can be solved using random projections:

- Let Φ be an $m \times N$ Gaussian random matrix, where $m = 10\varepsilon^{-2}\log p$.
- Compute $\mathbf{r} = \Phi\mathbf{q}$. For all $j = 1, \ldots, p$, compute $\mathbf{x}_j \to \mathbf{u}_j = \Phi\mathbf{x}_j$. Computational cost: $O(Np\log(p))$.
- Compute $\mathbf{x}_\varepsilon^* = \arg\min_{\mathbf{x}_j \in S} \|\mathbf{r} - \mathbf{u}_j\|$. Computational cost: of m searches: $O(pm\log(p+m))$.

Total computation cost: $O((N+m)p\log(p+m)) << O(Np^2)$!



Random projections and sparse recovery

Theorem (Subspace-preservation)

Suppose that Φ is an $m \times n$ random matrix with the distance-preservation property:

$$\textit{For any fixed } \mathbf{x}: \quad \mathbb{P}\Big(\big|\|\Phi\mathbf{x}\|^2 - \|\mathbf{x}\|^2\big| \geq \varepsilon\|\mathbf{x}\|^2\Big) \leq 2e^{-c_\varepsilon m}$$

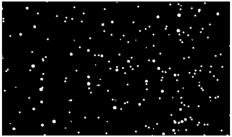
Let $k \leq c_\varepsilon m$ and let T_k be a k-dimensional subspace of $\mathbb{R}^n$. Then

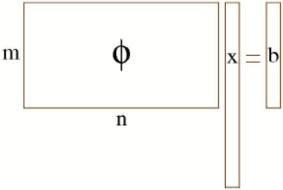
$$\mathbb{P}\Big(\textit{For all } \mathbf{x} \in T_k: \qquad (1-\varepsilon)\|\mathbf{x}\|^2 \leq \|\Phi\mathbf{x}\|^2 \leq (1+\varepsilon)\|\mathbf{x}\|^2\Big) \geq 1 - e^{-c'_\varepsilon m}$$

Outline of proof:

- A ε-cover and the Vitali covering lemma
- Continuity argument

Sparse recovery and RIP





Restricted Isometry Property of order k: Φ has the RIP of order k if

$$.8\|\mathbf{x}\|^2 \leq \|\Phi\mathbf{x}\|^2 \leq 1.2\|\mathbf{x}\|^2$$

for all k-sparse vectors $\mathbf{x} \in \mathbb{R}^n$.

Theorem

If Φ has RIP of order k, then for all k-sparse vectors $\mathbf{x}$ such that $\Phi\mathbf{x} = \mathbf{b}$,

$$\mathbf{x} = \arg\min\left\{\sum_{j=1}^{N} |z(j)| \quad : \quad \Phi\mathbf{z} = \mathbf{b}, \quad \mathbf{z} \in \mathbb{R}^n\right\}$$

Theorem (Distance-preservation implies RIP)

Suppose that Φ is an $m \times N$ random matrix with the subspace-preservation property:

$$\mathbb{P}\Big(\exists \mathbf{x} \in T_k : \quad (1-\varepsilon)\|\mathbf{x}\|^2 \leq \|\Phi\mathbf{x}\|^2 \leq (1+\varepsilon)\|\mathbf{x}\|^2\Big) \leq e^{-c'_\varepsilon m}$$

Then with probability greater than .99,

$$(1-\varepsilon)\|\mathbf{x}\|^2 \leq \|\Phi\mathbf{x}\|^2 \leq (1+\varepsilon)\|\mathbf{x}\|^2$$

for all $\mathbf{x}$ of sparsity level $k \leq c_\varepsilon m/\log(N)$.

Outline of proof:

- Bound for a fixed subspace T_k.
- Union bound over all $\binom{N}{k} \leq N^k$ subspaces of k-sparse vectors