Big Data Analysis Application and Practice (XAI605)

Other Linear Dimensionality Reduction Methods

2023 Spring

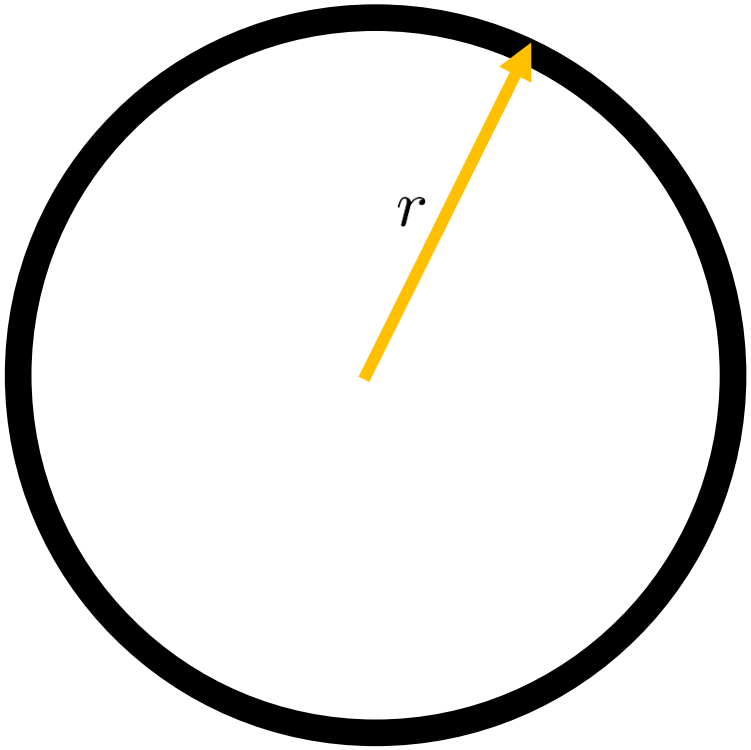
Instructor: Sejun Park

Counter intuitive property in high dim.

- Volume of a sphere of radius r

$$\text{Vol}(\mathbb{B}_n(r)) = \frac{\pi^{n/2} r^n}{\Gamma(1+n/2)}$$

Γ : gamma function



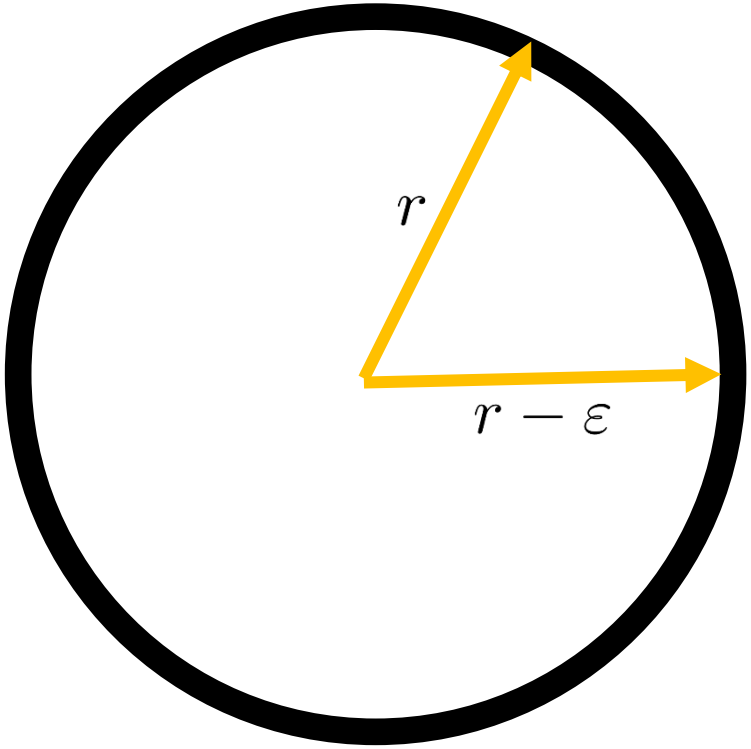
Counter intuitive property in high dim.

- Region around the shell occupies most volume of a sphere

$$\mathrm{Vol}(\mathbb{B}_n(r)) = \frac{\pi^{n/2} r^n}{\Gamma(1+n/2)}$$

$$\frac{\mathrm{Vol}(\mathbb{B}_n(r)) - \mathrm{Vol}(\mathbb{B}_n(r-\varepsilon))}{\mathrm{Vol}(\mathbb{B}_n(r))}$$

$$= 1 - \frac{(r-\varepsilon)^n}{r^n}$$



Curse of dimensionality in ML

- **Curse of dimensionality:** problems in data analysis that occurs as the dimensionality increases
  - Typically, under the setup: #samples $= n < p =$ dimension

- Statistical aspects
  - In high-dimensional data, it becomes more likely to find spurious or random correlations between unrelated features
  - This is often called “overfitting”

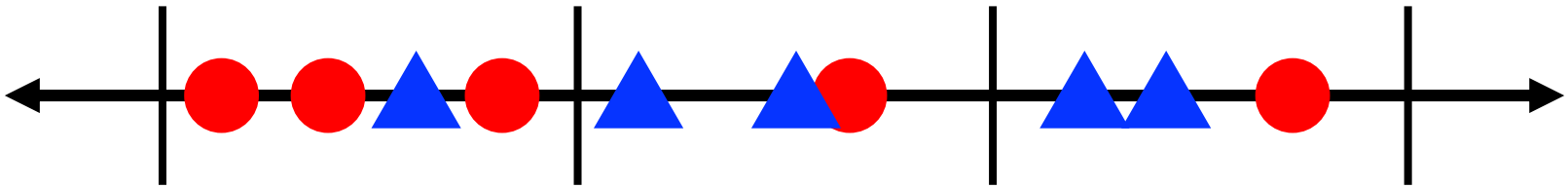
Curse of dimensionality in ML

- **Curse of dimensionality:** problems in data analysis that occurs as the dimensionality increases
  - Typically, under the setup: #samples $= n < p =$ dimension

- Computational aspects
  - Sampling from continuous/discrete distribution requires FLOPS at least exponential to the parameter dimension (until now), i.e., NP-hard

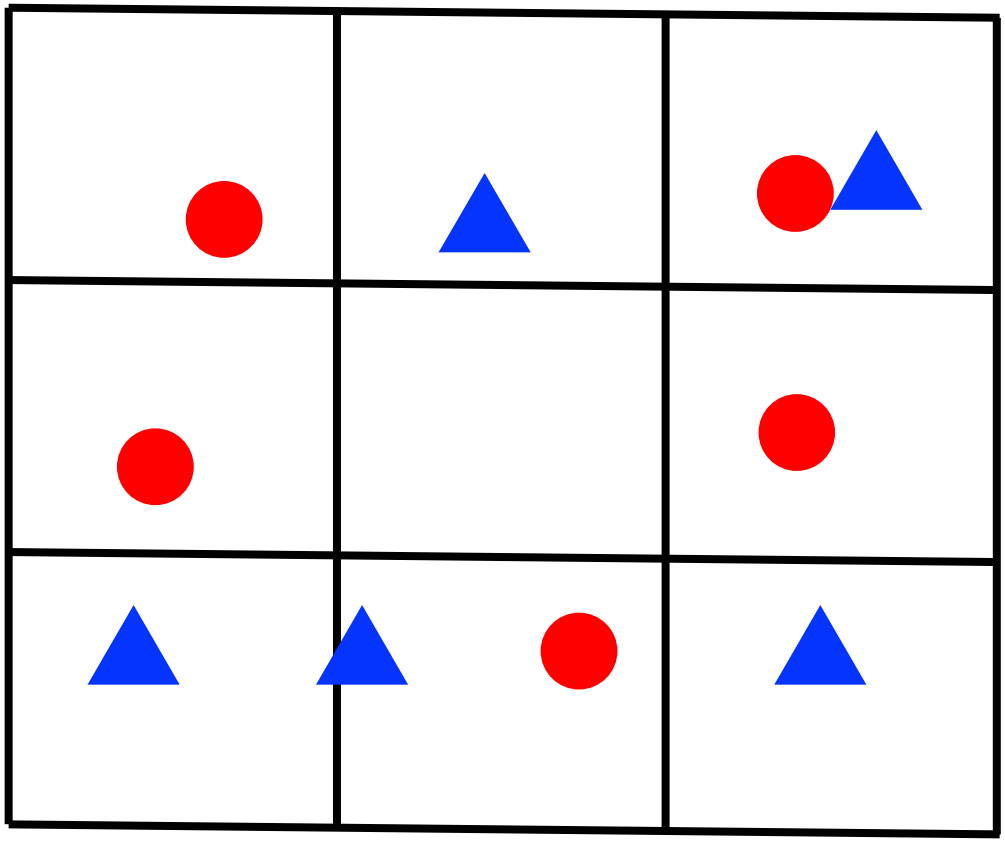
Curse of dimensionality in ML

- Consider a simple classification problem:
  - This is similar to the k-NN algorithm
  - First, split the state space (say a unit cube) by cubes of equal side length
  - Estimate the label of a new sample by the majority labels of training data in the cube containing the sample
- For low-dimensional data, this can be a good estimator of the true distribution



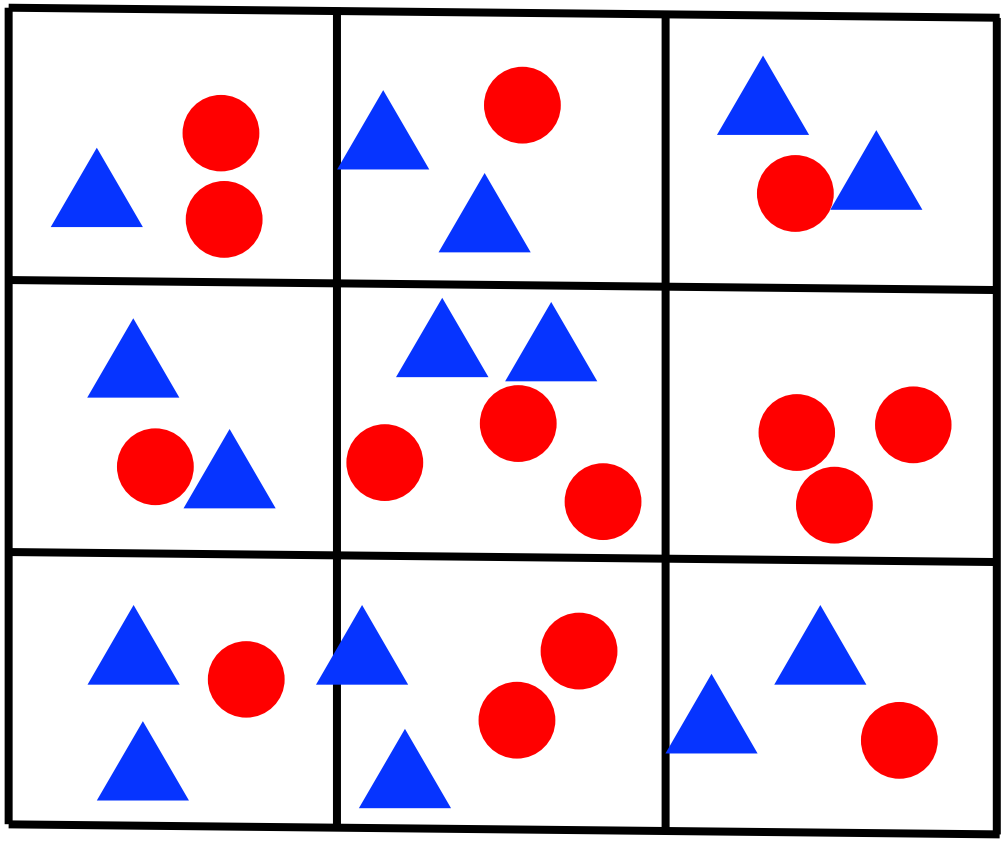
Curse of dimensionality in ML

- As the dimensionality of data increases, #samples in each cube decreases exponentially fast
  - Under fixed #sample



Curse of dimensionality in ML

- To have preserve the density of samples in each cube, we need exponentially many samples



High-dimensional data

- With big data embedded in bigger dimensional space, can we efficiently do dimension reduction?
  - e.g., we want to preserve the pairwise distance between data (MDS)

- We may use PCA (or equivalently MDS) for small k
  - Finding basis requires O(#dim x #data x #data) FLOPS
    - For both SVD & power iteration under assuming naïve matrix multiplication
  - In addition, we need to perform additional matrix product for projecting original data to the found subspace

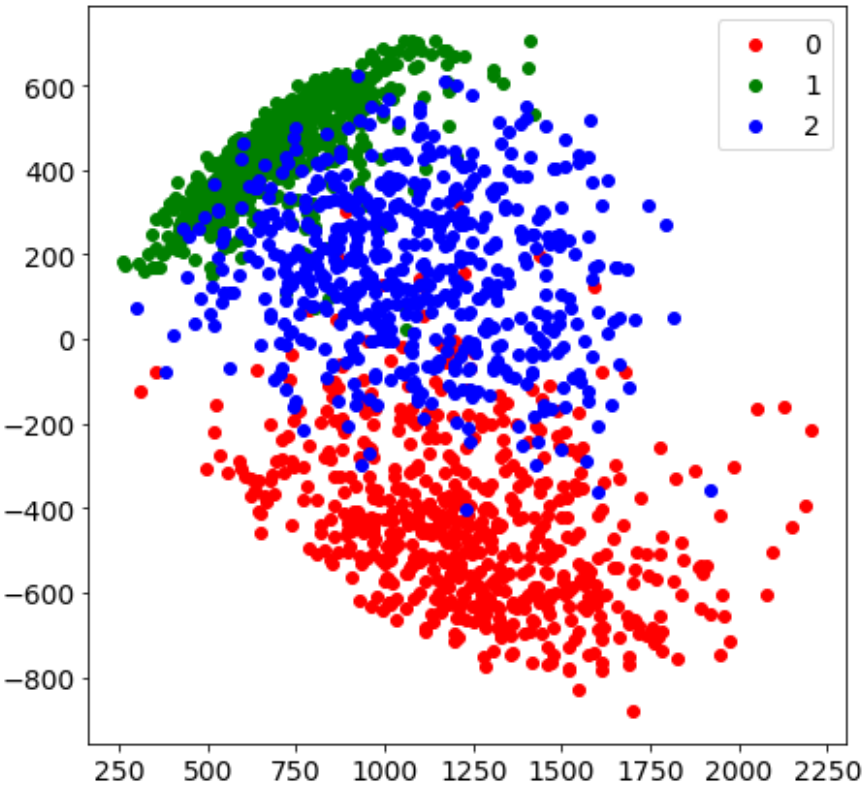
High-dimensional data

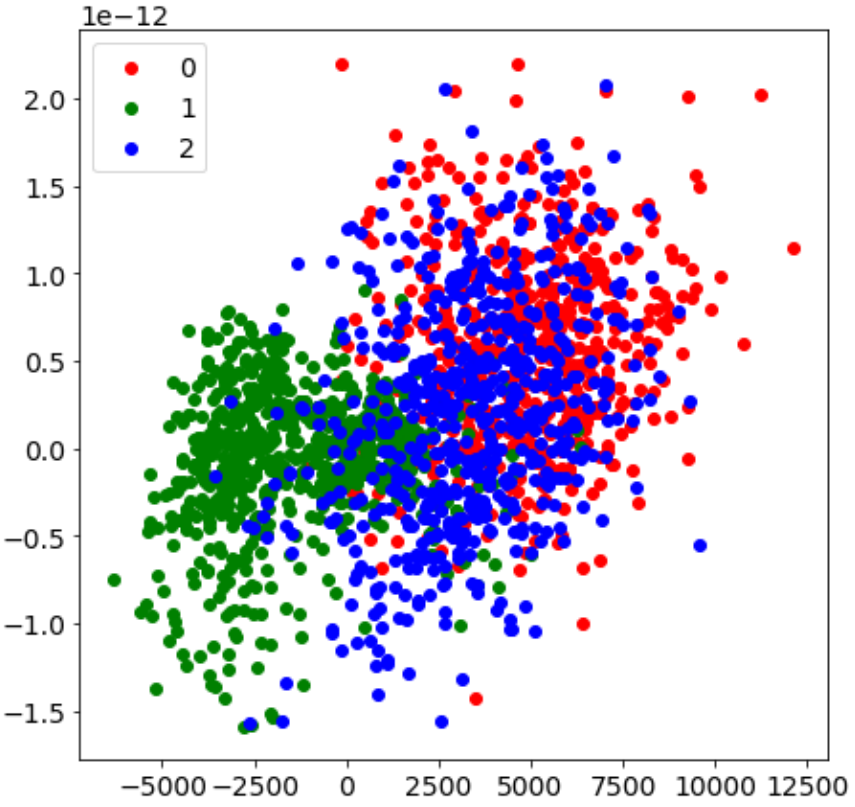
- With big data embedded in bigger dimensional space, can we efficiently do dimension reduction?
  - e.g., we want to preserve the pairwise distance between data (MDS)

- We may use PCA (or equivalently MDS) for small k
  - Finding basis requires O(#dim x #data x #data) FLOPS
    - For both SVD & power iteration under assuming naïve matrix multiplication
  - In addition, we need to perform additional matrix product for projecting original data to the found subspace
- Can we do this more efficiently?

Naïve idea

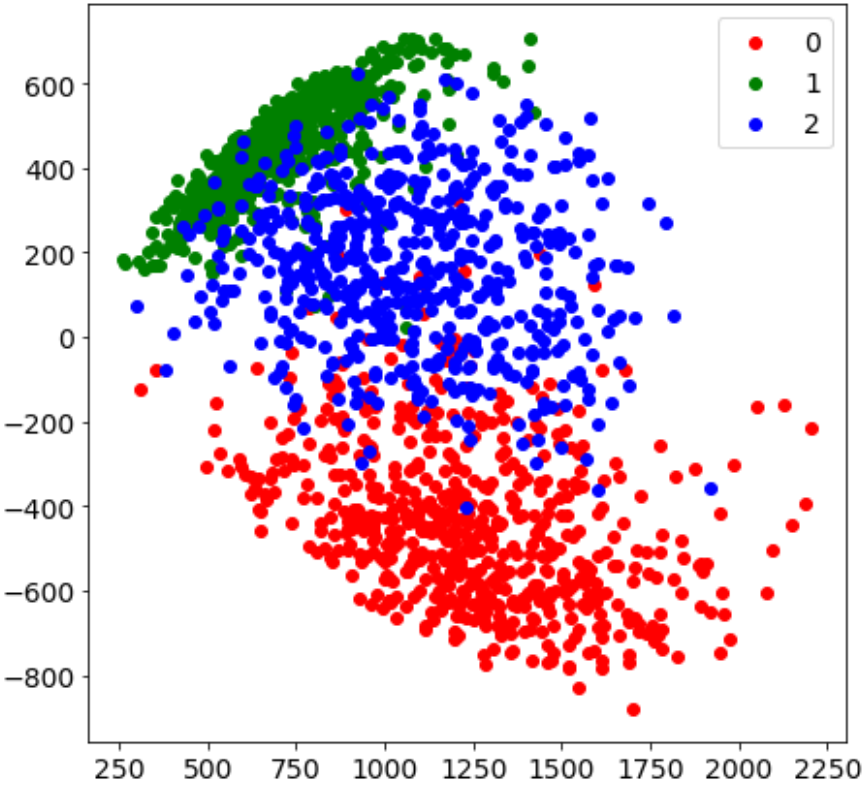
- We can choose random subspace and project data
  - Subsampled MNIST data with labels 0, 1, 2 (#samples ~ 1900)
  - Left: original PCA, Right: projection to random subspace (dim=2)

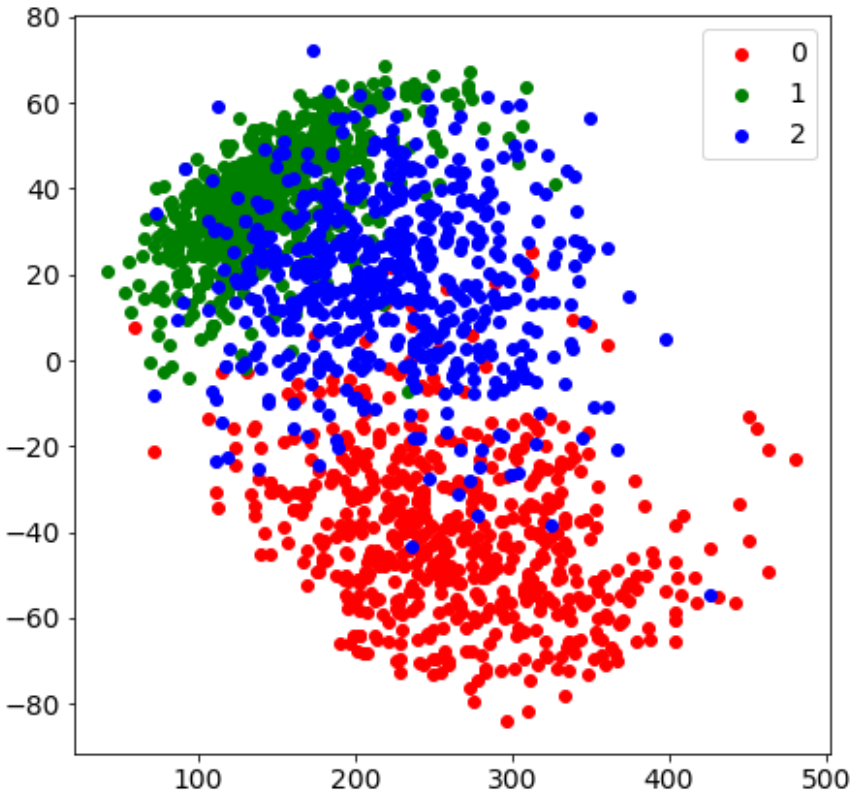




Naïve idea

- We can also do random projection first, and apply PCA
  - Left: original PCA (dim=28x28=784 -> 2)
  - Right: PCA after random projection (dim=784 -> 78 -> 2)





Johnson-Lindenstrauss lemma

- Why such a naïve idea works?
  - Mathematical explanation: Johnson-Lindenstrauss lemma

For any $\varepsilon > 0$, $x_1, \ldots, x_n \in \mathbb{R}^p$, and an integer $k > (8 \log n)/\varepsilon^2$, there exists a linear map $f : \mathbb{R}^p \to \mathbb{R}^k$ such that for any i, j

$$(1 - \varepsilon)\|x_i - x_j\| \leq \|f(x_i) - f(x_j)\| \leq (1 + \varepsilon)\|x_i - x_j\|$$

Johnson-Lindenstrauss lemma

- Why such a naïve idea works?
  - Mathematical explanation: Johnson-Lindenstrauss lemma

For any $\varepsilon > 0$, $x_1, \ldots, x_n \in \mathbb{R}^p$, and an integer $k > (8 \log n)/\varepsilon^2$, there exists a linear map $f : \mathbb{R}^p \to \mathbb{R}^k$ such that for any i, j

$$(1 - \varepsilon)\|x_i - x_j\| \leq \|f(x_i) - f(x_j)\| \leq (1 + \varepsilon)\|x_i - x_j\|$$

- If $n = 1900$, then $8 \log n \approx 26.23 \Longrightarrow \varepsilon \leq 1.725$ with $k = 78$
- Theory often provides a vacuous bound but this bound is indeed tight up to a constant multiplicative factor!

Proof sketch

Lemma. Let $x \in \mathbb{R}^p$ and M be a $k \times p$ random matrix whose entries are i.i.d. $N(0,1)$. Then,

$$\mathbb{P}\left((1-\varepsilon)\|x\|^2 \leq \left\| \frac{1}{\sqrt{k}} Mx \right\|^2 \leq (1+\varepsilon)\|x\|^2 \right) \geq 1 - 2e^{-k(\varepsilon^2 - \varepsilon^3)/4}$$

Proof sketch

Lemma. Let $x \in \mathbb{R}^p$ and M be a $k \times p$ random matrix whose entries are i.i.d. $N(0,1)$. Then,

$$\mathbb{P}\left((1-\varepsilon)\|x\|^2 \leq \left\| \frac{1}{\sqrt{k}} Mx \right\|^2 \leq (1+\varepsilon)\|x\|^2 \right) \geq 1 - 2e^{-k(\varepsilon^2-\varepsilon^3)/4}$$

Proof sketch. This is from standard concentration inequality (i.e., law of large numbers) with the following observation

$$\mathbb{E}\left[\left\| \frac{1}{\sqrt{k}} Mx \right\| \right] = \frac{1}{k} \sum_{i=1}^{k} \sum_{j=1}^{p} \mathbb{E}\left[(M_{ij} x_j)^2 \right] = \frac{1}{k} \sum_{i=1}^{k} \sum_{j=1}^{p} (x_j)^2$$

$$= \frac{1}{k} \sum_{i=1}^{k} \|x\|^2 = \|x\|^2$$

Proof sketch

Proof sketch of JL lemma. Let $N = M/\sqrt{k}$ and assume $\varepsilon < \frac{1}{2}$

$$\mathbb{P}\left(\exists i, j \text{ s.t. } \|N(x_i - x_j)\|^2 \notin (1 \pm \varepsilon)\|x_i - x_j\|^2\right)$$

$$\leq \sum_{i>j} \mathbb{P}\left(\|N(x_i - x_j)\|^2 \notin (1 \pm \varepsilon)\|x_i - x_j\|^2\right)$$

$$\leq \frac{n(n-1)}{2} \cdot 2e^{-k(\varepsilon^2 - \varepsilon^3)/4}$$

$$< n^2 e^{-k\varepsilon^2/8}$$

$$\leq 1 \quad \text{under } k \geq \frac{16 \log n}{\varepsilon^2}$$

Johnson-Lindenstrauss transform

- JL lemma only guarantees the existence of good linear maps

- **Q.** Can we find such a good linear map?
- **A.** Yes (with high probability)
  - Choose a realization of the random matrix used in the proof!

$$x \mapsto \frac{1}{\sqrt{k}} M x$$

Complexity of random projection

- Random projection (JL transformation) requires one matrix multiplication (matrix size: k x p, p x n)

- This is extremely fast: even if we find basis (subspace) via some optimization procedure (e.g. PCA), we still need such a projection step

- And as we observed before, RP can be done before PCA

Fast Johnson-Lindenstrauss transform

- Computational complexity of random projection?
  - Naïve matrix-vector product requires O(kp) FLOPS

Fast Johnson-Lindenstrauss transform

- Computational complexity of random projection?
  - Naïve matrix-vector product requires O(kp) FLOPS

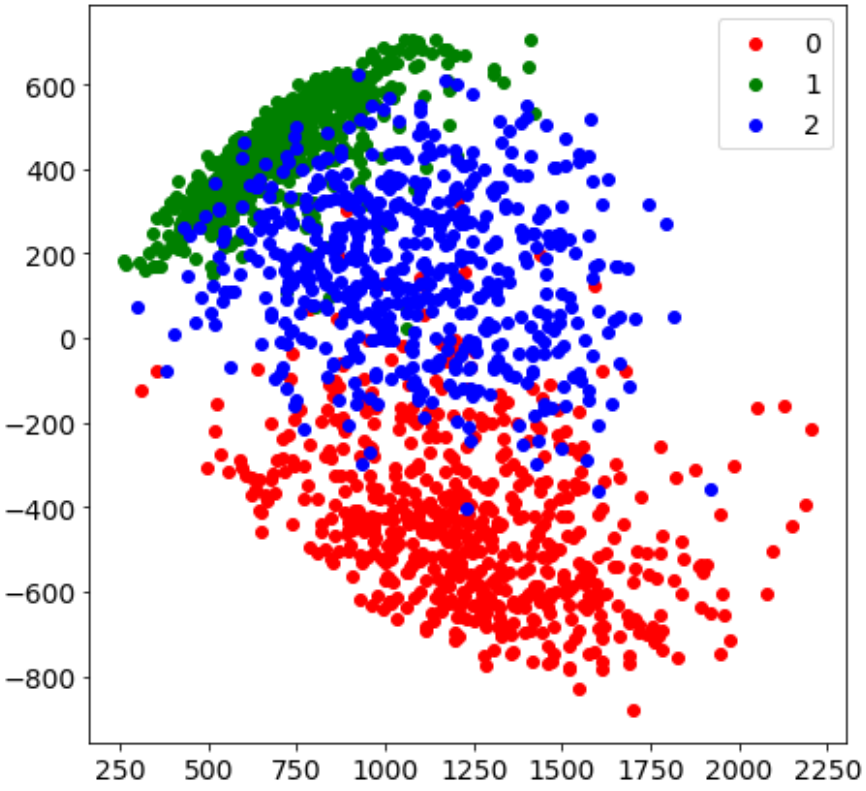
- **Q.** Can we do this even faster?
- **A.** Yes (with different random matrix)
  - This requires only $O(p \log p + k \log^2 n)$ FLOPS

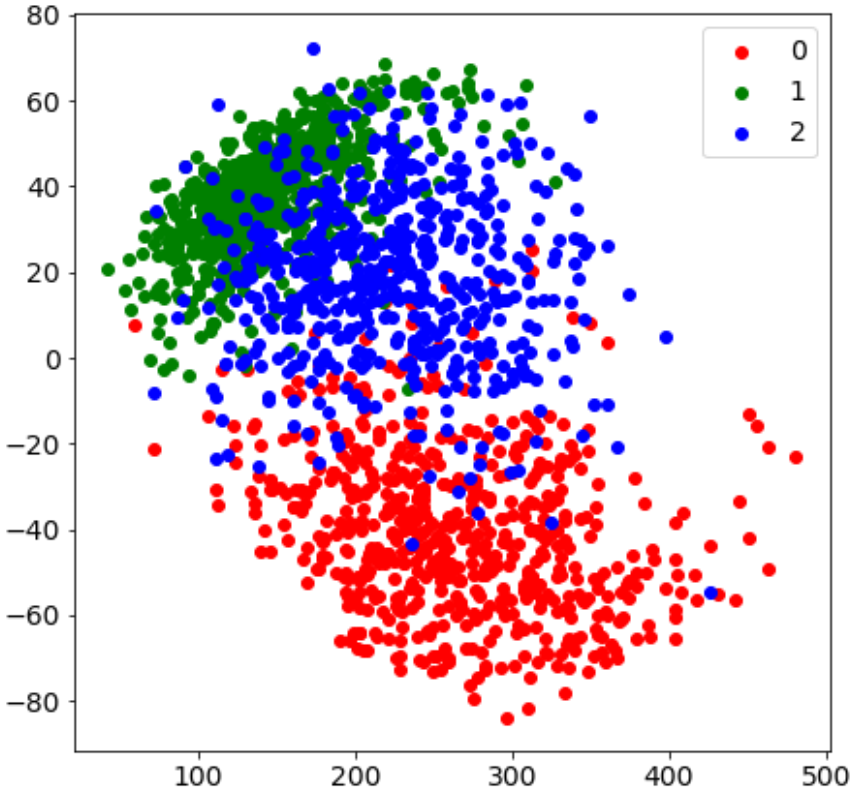
$$x \mapsto PHDx$$

  - $D \in \mathbb{R}^{p \times p}$: diagonal matrix
  - $H \in \mathbb{R}^{p \times p}$: matrix encoding FFT (FFT can be done in $O(p \log p)$ FLOPS)
  - $P \in \mathbb{R}^{k \times p}$: sparse matrix containing $\Theta(k \log^2 n)$ non-zero entries

Distance distortion

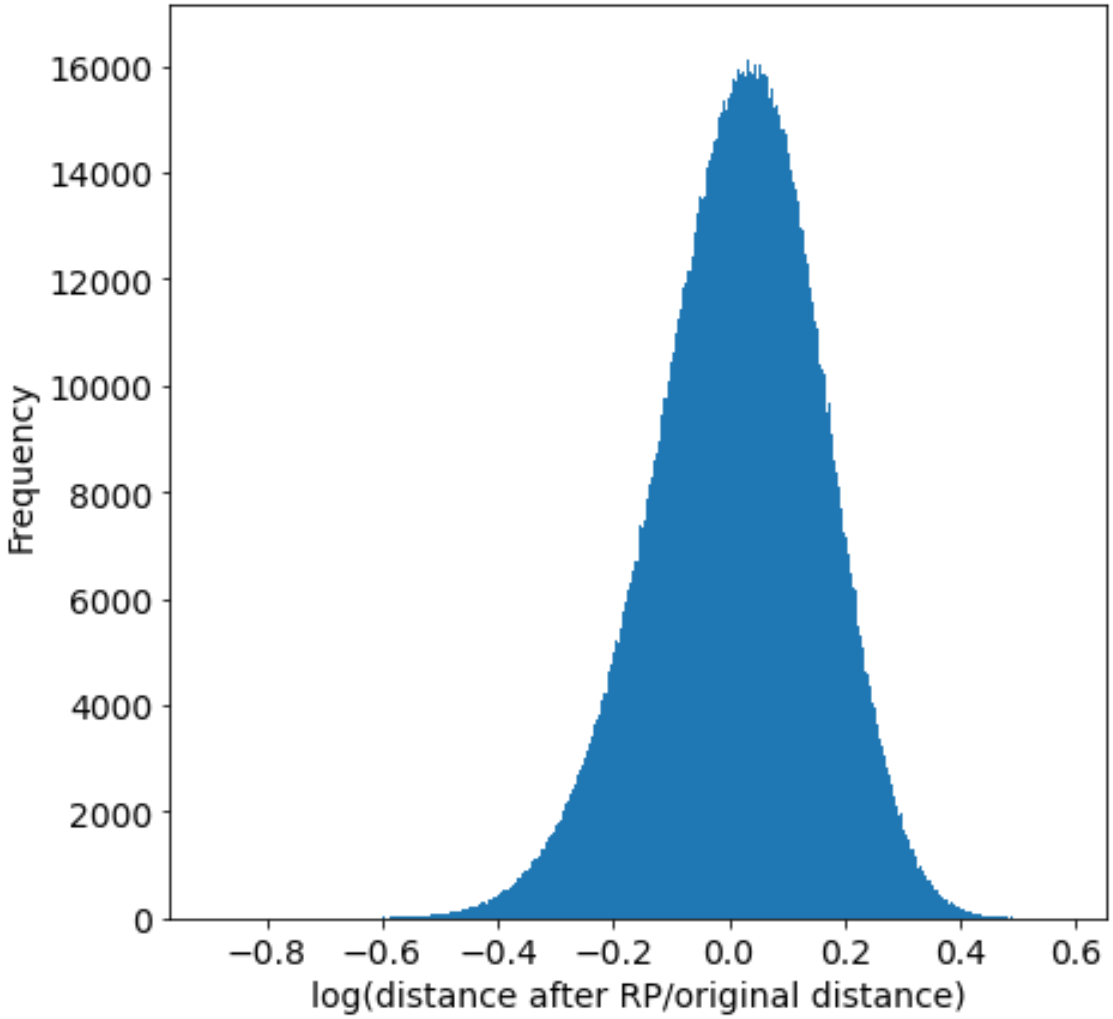
- #sample~1900, random projection from dim=784 to dim=74
  - Left: original PCA, Right: PCA after RP





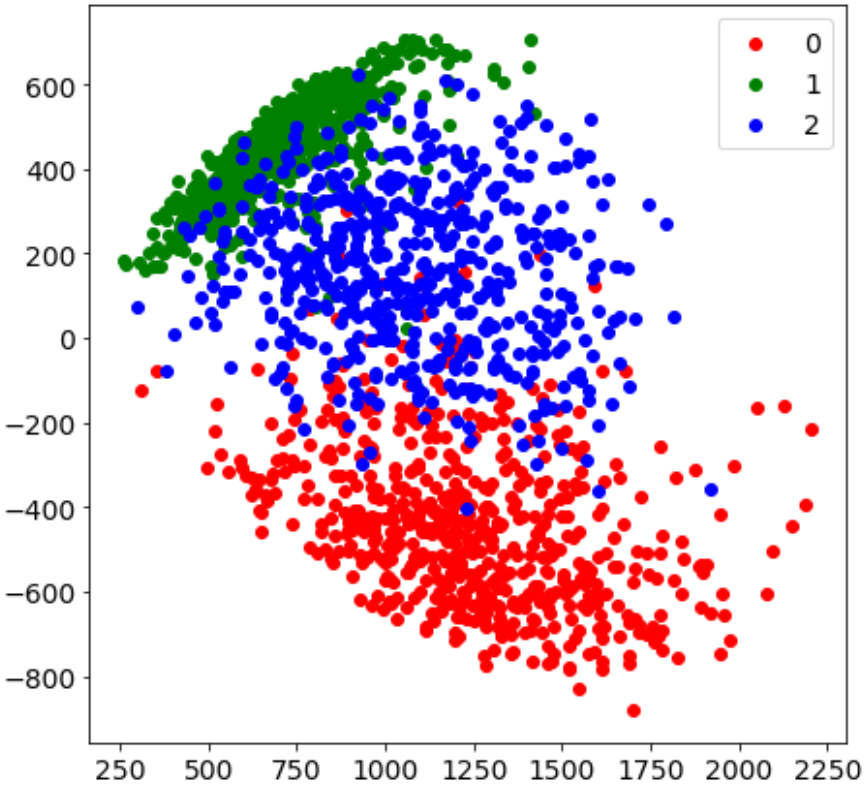
Distance distortion

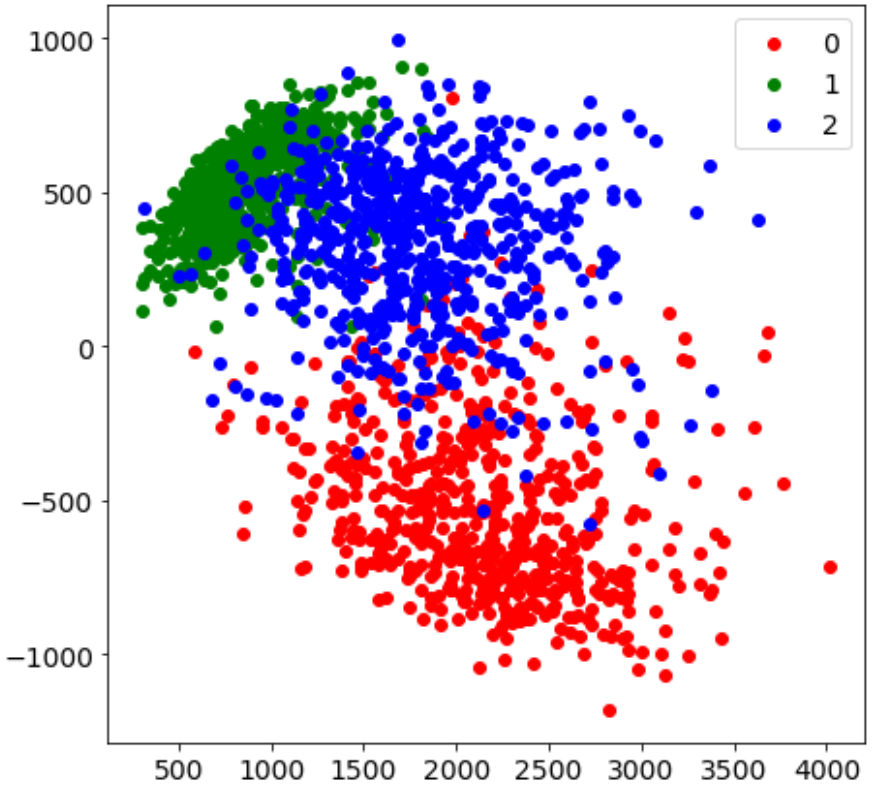
- #sample~1900, random projection from dim=784 to dim=74



Distance distortion

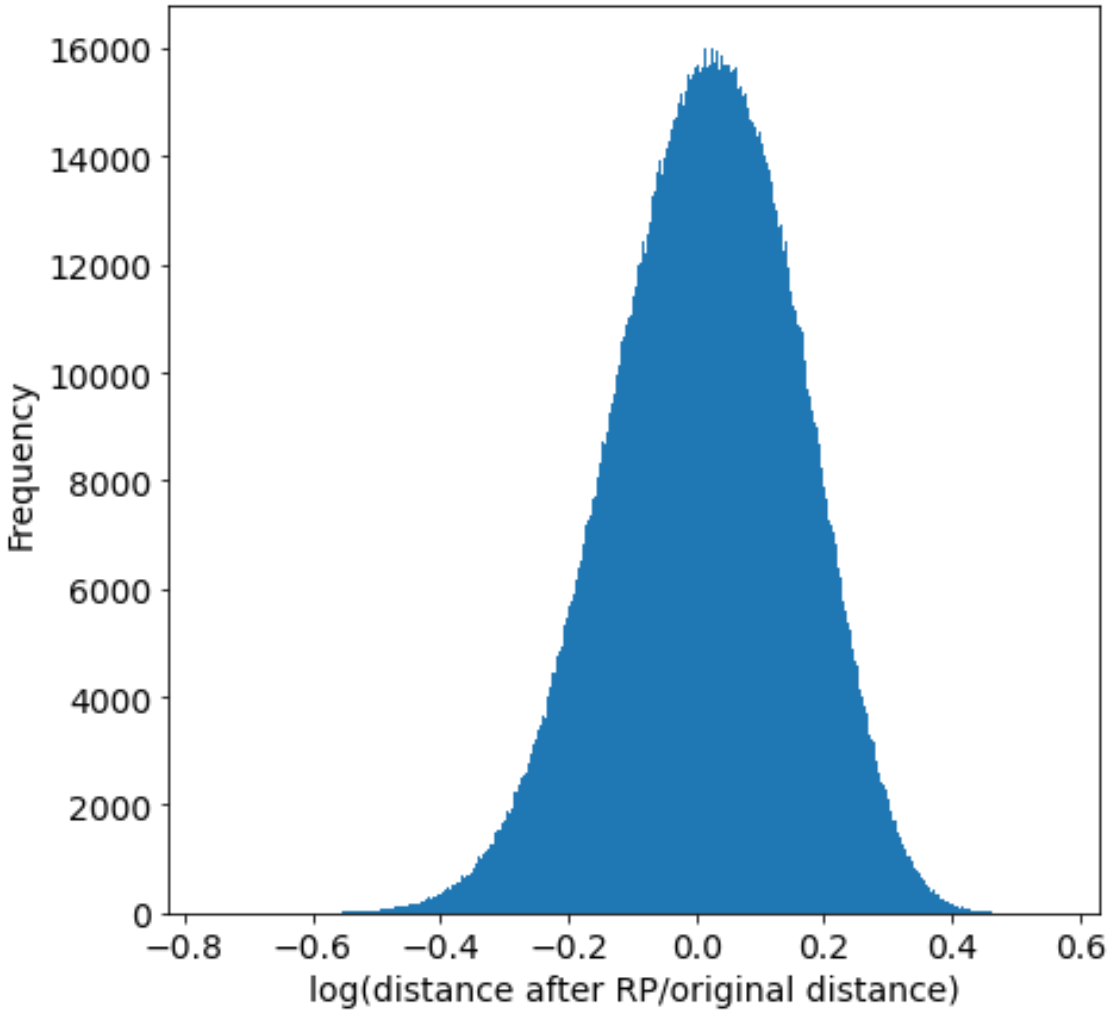
- #sample~1900, random projection from dim=784 to dim=100
  - Left: original PCA, Right: PCA after RP





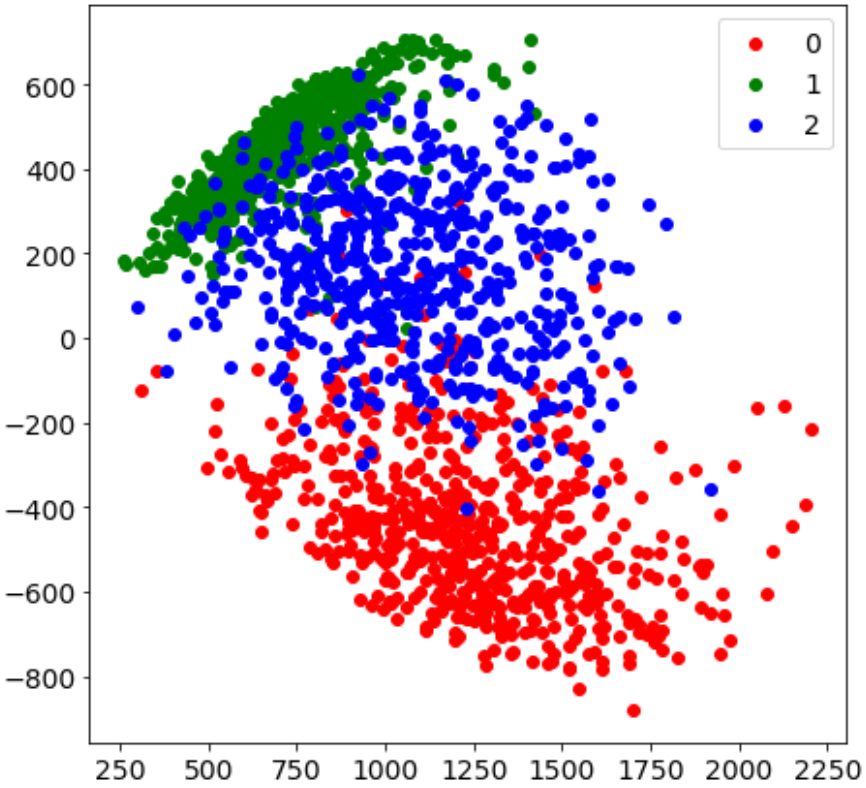
Distance distortion

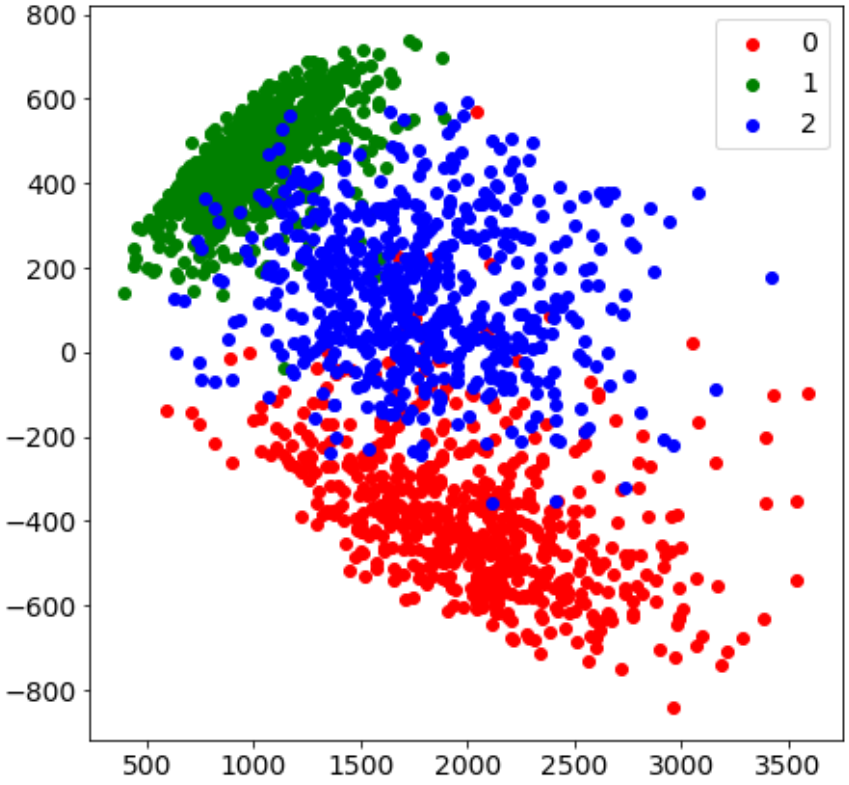
- #sample~1900, random projection from dim=784 to dim=100



Distance distortion

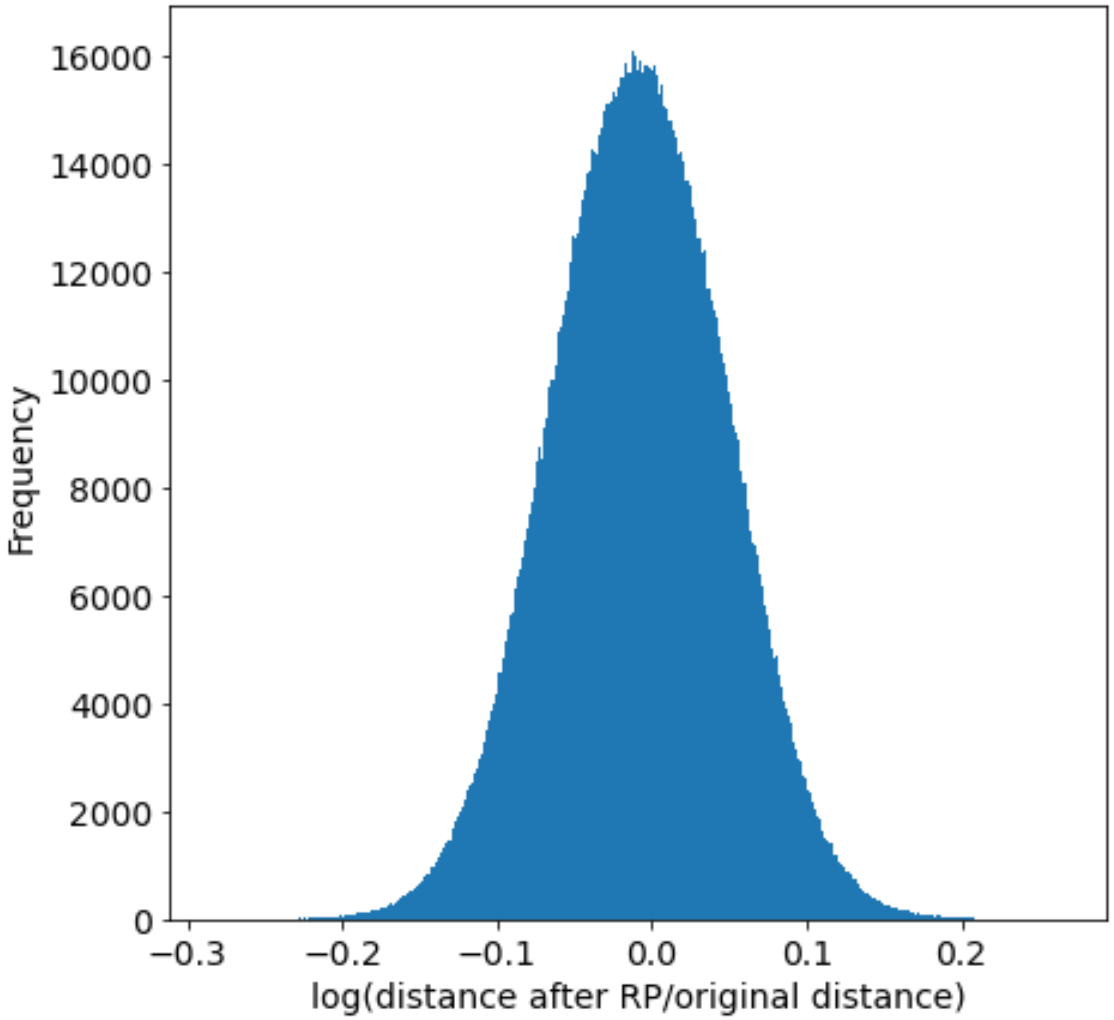
- #sample~1900, random projection from dim=784 to dim=500
  - Left: original PCA, Right: PCA after RP





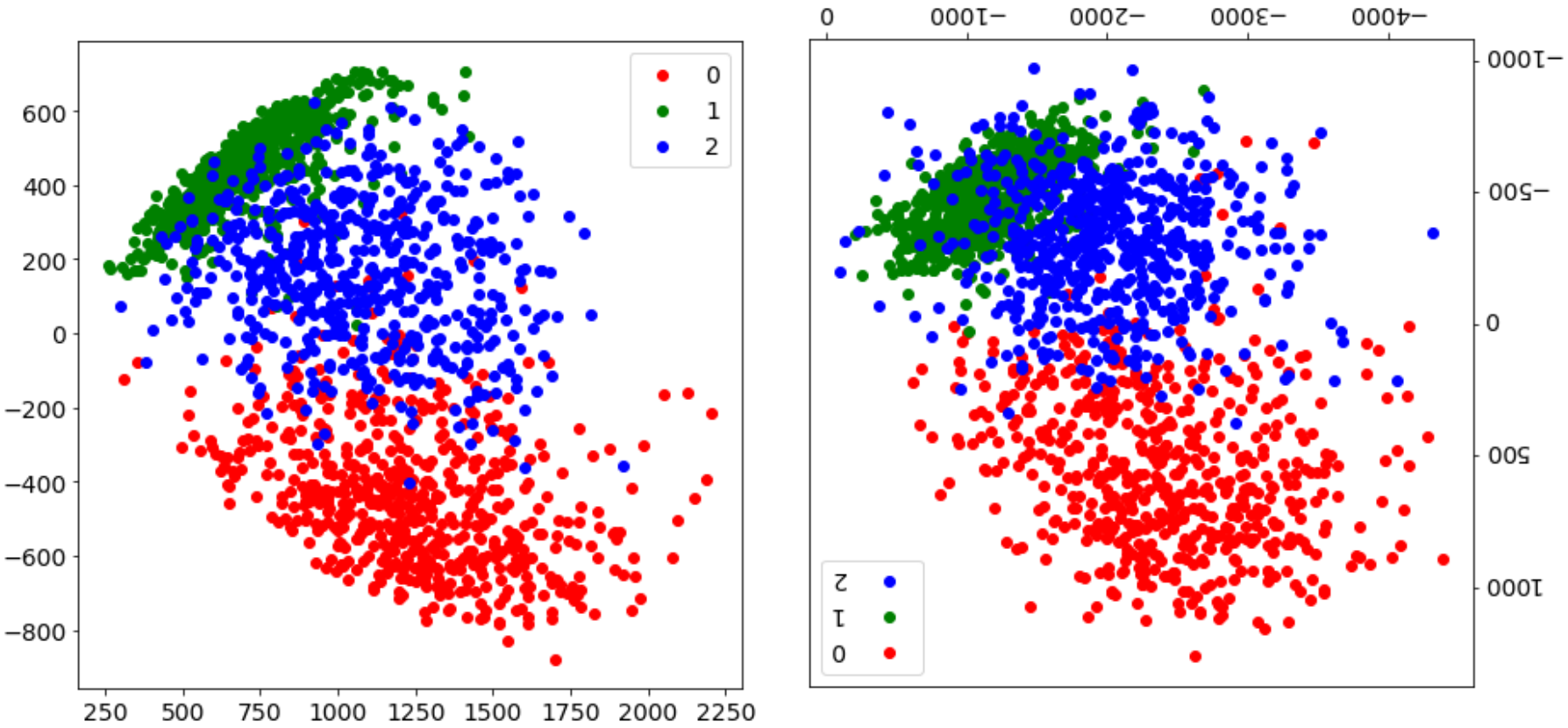
Distance distortion

- #sample~1900, random projection from dim=784 to dim=500



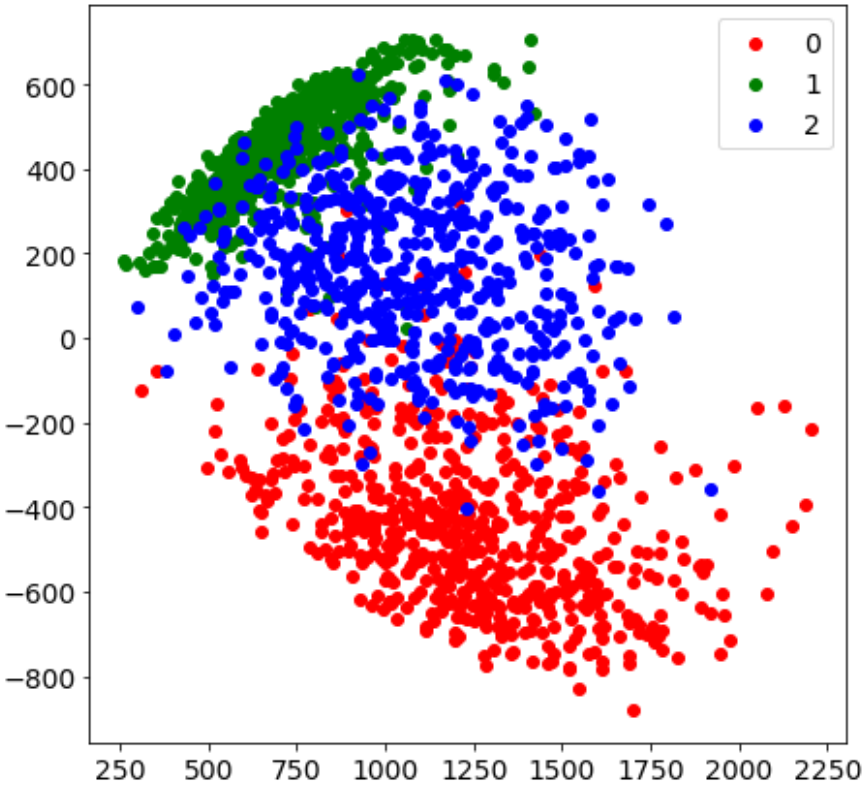
Distance distortion

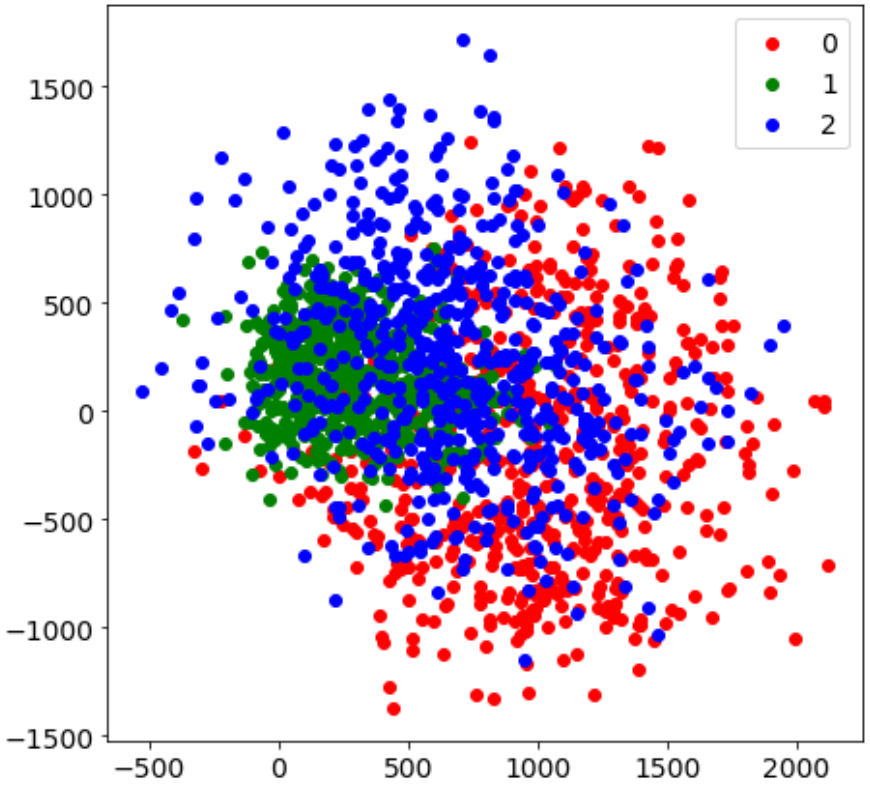
- #sample~1900, random projection from dim=784 to dim=30
  - Left: original PCA, Right: PCA after RP



Distance distortion

- #sample~1900, random projection from dim=784 to dim=10
  - Left: original PCA, Right: PCA after RP

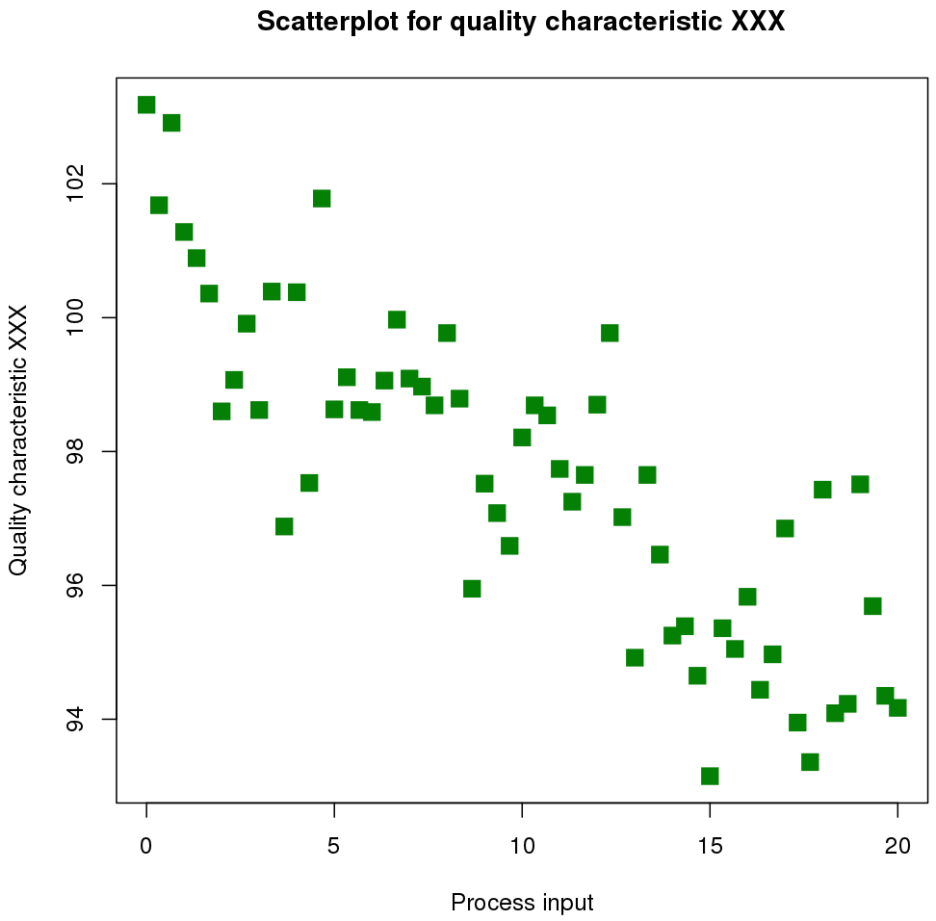




Linear dimensionality reduction with labeled data

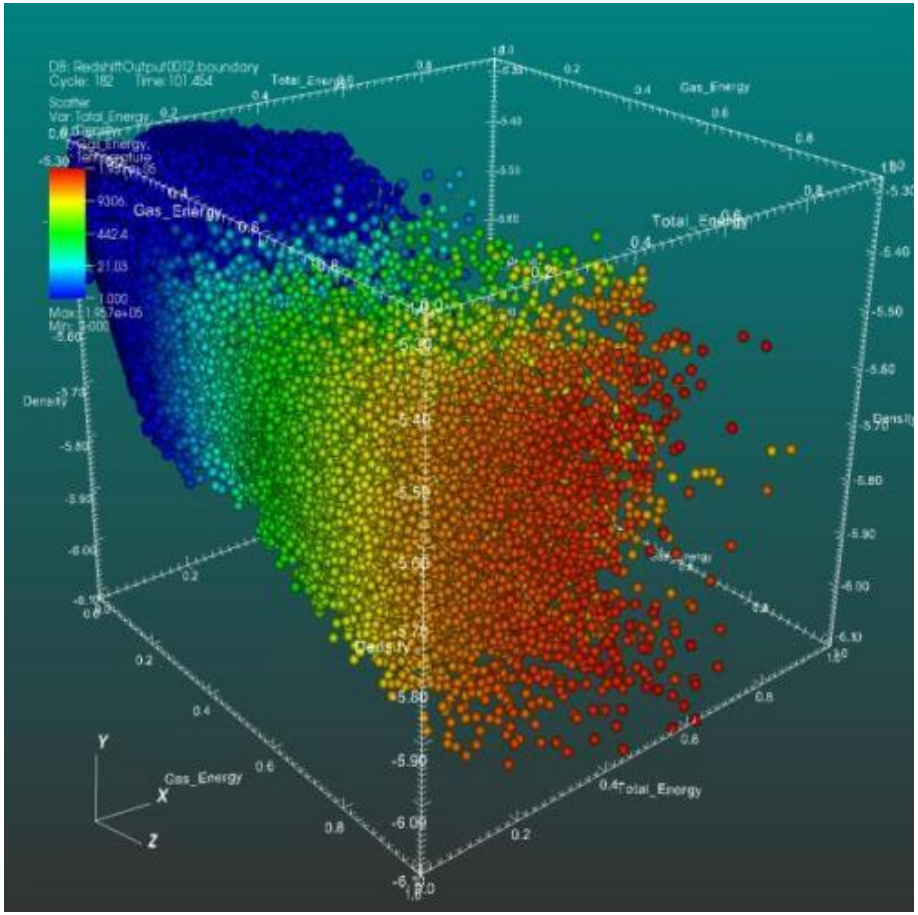
Labeled data

- Given labeled data, can we see a good scatter plot via PCA?
  - PCA does not use “label information”



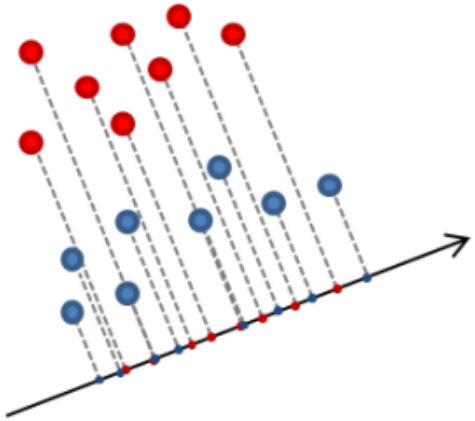
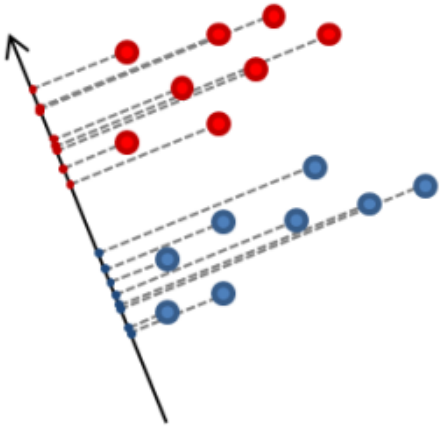
Labeled data

- Given labeled data, can we see scatter plot using PCA?
  - PCA does not use “label information”



Labeled data

- Given labeled data, can we see scatter plot using PCA?
  - PCA does not use “label information”

Labeled data	PCA projection: Maximizing variance	We may want this!
		

Sufficient dimension reduction

- Dimensionality reduction method considering classification and regression setups

- **Goal:** find the best low-dimensional representation of inputs preserving **information** of data
  - In PCA, we aim to minimize the error between the original data and reconstructed data from low-dimensional representation

Sufficient dimension reduction

- As in typical classical methods, we first consider
  - Linear transformation
  - Linear model in both classification and regression setups
  - Gaussian noise

Sufficient dimension reduction

- As in typical classical methods, we first consider
  - Linear transformation
  - Linear model in both classification and regression setups
  - Gaussian noise

- Suppose that input can be represented by low-dimensional representation without label information loss
  - X: input RV, Y: label RV, P: projection matrix

$$Y \perp X \,|\, P^\top X$$

Uniqueness of projection matrix

- Such a projection matrix P is not unique in general
  - We can use a different basis for the column space of P
  - i.e., for any invertible matrix M

$$Y \perp X \mid P^\top X \iff Y \perp X \mid MP^\top X$$

Uniqueness of projection matrix

- Such a projection matrix P is not unique in general
  - We can use a different basis for the column space of P

$$Y \perp X \,|\, P^\top X \iff Y \perp X \,|\, Q^\top X$$

- So we may consider the subspace spanned by the column space of P

$$\mathcal{S}(P) = \mathrm{span}(p_1, \ldots, p_k)$$

$$P = [p_1, \ldots, p_k] \in \mathbb{R}^{p \times k}$$

Dimension reduction subspace

- S is a **dimension reduction subspace** if its basis $b_1, \ldots, b_k$ satisfies

$$Y \perp X \mid b_1^\top X, \ldots b_k^\top X$$

  - We are particularly interested in minimum DRS (DRS with minimum dim.)

Dimension reduction subspace

- S is a **dimension reduction subspace** if its basis $b_1, \ldots, b_k$ satisfies

$$Y \perp X \,|\, b_1^\top X, \ldots b_k^\top X$$

  - We are particularly interested in minimum DRS (DRS with minimum dim.)

- Minimum DRS may not be unique
  - X follows uniform distribution on $\{(x_1, x_2) \in \mathbb{R}^2 : x_1^2 + x_2^2 = 1\}$
  - $y|x = x_1^2 + \varepsilon$

$$Y \perp X \,|\, x_1^2,\ Y \perp X \,|\, x_2^2$$

Dimension reduction subspace

- S is a **dimension reduction subspace** if its basis $b_1, \ldots, b_k$ satisfies

$$Y \perp X \mid b_1^\top X, \ldots b_k^\top X$$

  - We are particularly interested in minimum DRS (DRS with minimum dim.)

- A unique minimum DRS exists (called **central subspace**) if the support of X is open and convex

- We want to approximate the central subspace

Inverse regression and PCA

- Now, consider the following **inverse model**
  - Want to estimate an input given a label
  - $\mu \in \mathbb{R}^p, U \in \mathbb{R}^{p \times k}, \beta_y \in \mathbb{R}^k, \varepsilon \sim N(0, I)$
  - Assume $\sum_y \beta_y = 0$ and columns of U are orthonormal $(U^\top U = I)$

$$(X|Y = y) = \mu + U\beta_y + \varepsilon$$

Inverse regression and PCA

- Observe the distribution of $Y|X$

$$
\begin{aligned}
f_{Y|X}(y|x) &\propto f_{X|Y}(x|y) f_Y(y) \\
&\propto \exp\left(-\frac{1}{2}\|x-\mu-U\beta_y\|^2\right) f_Y(y) \\
&\propto \exp\left(-\frac{1}{2}(\beta_y^\top \beta_y - 2\beta_y^\top U^\top (x-\mu))\right) f_Y(y)
\end{aligned}
$$

Inverse regression and PCA

- Observe the distribution of $Y|U^\top X$

$$
\begin{aligned}
f_{Y|X}(y|U^\top x) &\propto f_{U^\top X|Y}(U^\top x|y) f_Y(y) \\
&\propto \exp\left(-\frac{1}{2}\|U^\top x - U^\top \mu - \beta_y\|^2\right) f_Y(y) \\
&\propto \exp\left(-\frac{1}{2}(\beta_y^\top \beta_y - 2\beta_y^\top U^\top (x-\mu))\right) f_Y(y)
\end{aligned}
$$

Inverse regression and PCA

- Observe the distribution of $Y|U^\top X$

$$
\begin{aligned}
f_{Y|U^\top X}(y|U^\top x) &\propto f_{U^\top X|Y}(U^\top x|y) f_Y(y) \\
&\propto \exp\left(-\frac{1}{2}\|U^\top x - U^\top \mu - \beta_y\|^2\right) f_Y(y) \\
&\propto \exp\left(-\frac{1}{2}(\beta_y^\top \beta_y - 2\beta_y^\top U^\top (x-\mu))\right) f_Y(y)
\end{aligned}
$$

- This implies that there is no information loss in the projection
  - Under this specific model at least, i.e., $Y \perp X \,|\, U^\top X$

Sliced inverse regression

- Now, consider a more general setup
  - ε: noise independent of X (e.g., additive Gaussian)

$$Y = g(v_1^\top X, \ldots, v_k^\top X, \varepsilon)$$

- Here, we want to estimate the central subspace

$$\text{span}(v_1, \ldots, v_k)$$

Sliced inverse regression

- We make one assumption
  - Inverse regression model satisfies this assumption

Assumption. For any $b \in \mathbb{R}^p$, $\mathbb{E}[b^\top x | v_1^\top x, \ldots, v_k^\top x]$ is linear in $v_1^\top x, \ldots, v_k^\top x$, i.e., there exist $c_0, \ldots, c_k$ such that

$$\mathbb{E}[b^\top x | v_1^\top x, \ldots, v_k^\top x] = c_0 + c_1 v_1^\top x + \cdots + v_k^\top x$$

Sliced inverse regression

- Under the assumption, it holds that
  - See [Sliced Inverse Regression for Dimension Reduction (Li, 1991)]

$$\mathbb{E}[X|Y=y] - \mathbb{E}[X] = \mu_y - \mu \in \text{span}(\Sigma v_1, \ldots, \Sigma v_k)$$

- This can be interpreted in a different way
  - For $Z = \Sigma^{-1/2}(X - \mu)$, i.e., whitened data

$$\mathbb{E}[Z|Y=y] \in \text{span}(\Sigma^{1/2} v_1, \ldots, \Sigma^{1/2} v_k)$$

- This implies that eigenvectors of $\text{Cov}(\mathbb{E}[Z|Y])$ is in the span
  - And vectors orthogonal to the span cannot be the eigenvectors

Sliced inverse regression

- Sliced inverse regression
  - Compute $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i,\ \hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})(x_i - \hat{\mu})^\top$
  - Whiten data: $z_i = \hat{\Sigma}^{-1/2}(x_i - \hat{\mu})$
  - Choose c disjoint intervals $\mathcal{I}_1, \ldots, \mathcal{I}_c$ of the support of y
    - n_j: #samples in $\mathcal{I}_j$
  - Compute $\hat{\nu}_j = \frac{1}{n_j}\sum_{i:x_i \in \mathcal{I}_j} x_i$, $\hat{\Phi} = \frac{1}{n}\sum_{j=1}^{c} n_j \hat{\nu}_j \hat{\nu}_j^\top$
  - Compute top-k eigenvectors $u'_1, \ldots, u'_k$ of $\hat{\Phi}$
  - Compute $\hat{U}_k = [\Sigma^{-1/2}u'_1, \ldots, \Sigma^{-1/2}u'_k]$

Sliced inverse regression

- Sliced inverse regression
  - Compute $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i,\ \hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})(x_i - \hat{\mu})^\top$
  - Whiten data: $z_i = \hat{\Sigma}^{-1/2}(x_i - \hat{\mu})$
  - Choose c disjoint intervals $\mathcal{I}_1, \ldots, \mathcal{I}_c$ of the support of y
    - n_j: #samples in $\mathcal{I}_j$
  - Compute $\hat{\nu}_j = \frac{1}{n_j}\sum_{i:x_i \in \mathcal{I}_j} x_i$, $\hat{\Phi} = \frac{1}{n}\sum_{j=1}^{c} n_j \hat{\nu}_j \hat{\nu}_j^\top$
  - Compute top-k eigenvectors $u'_1, \ldots, u'_k$ of $\hat{\Phi}$
  - Compute $\hat{U}_k = [\Sigma^{-1/2}u'_1, \ldots, \Sigma^{-1/2}u'_k]$

Sliced inverse regression

- **Remark**
  - the sliced inverse regression can only approximate a subspace of the central subspace
  - Typical choices of slices $\mathcal{I}_1, \ldots, \mathcal{I}_c$ are intervals containing a similar #samples

Fisher’s linear discriminant analysis

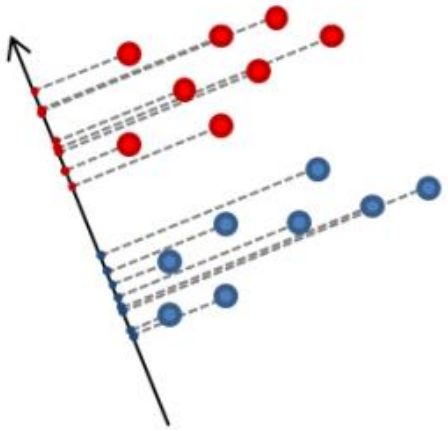
- Now, consider the binary classification setup: y = 1, 2
  - n_j: #sample of class j

Fisher's linear discriminant analysis

- Now, consider the binary classification setup: y = 1, 2
  - n_j: #sample of class j
- **Objective of Fisher's linear discriminant analysis**

$$\arg \max_{u:\|u\|=1} \frac{u^\top(\hat{\mu}_1 - \hat{\mu}_2)}{u^\top \hat{\Sigma}_1 u + u^\top \hat{\Sigma}_2 u}$$

$$\hat{\mu}_j = \frac{1}{n_j} \sum_{i:y_i=j} x_i, \; \frac{1}{n_j} \sum_{i:y_i=j} (x_i - \hat{\mu}_j)(x_i - \hat{\mu}_j)^\top$$

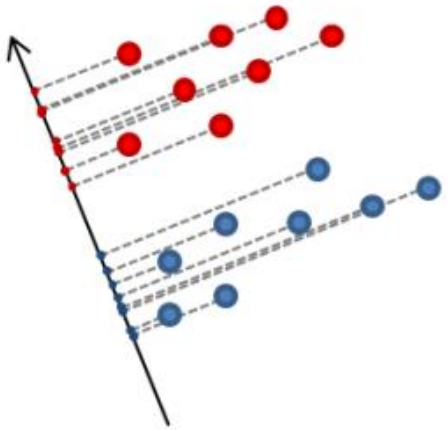


Fisher's linear discriminant analysis

- Now, consider the binary classification setup: y = 1, 2
  - n_j: #sample of class j
- **Objective of Fisher's linear discriminant analysis**

$$\arg \max_{u:\|u\|=1} \frac{u^\top(\hat{\mu}_1 - \hat{\mu}_2)}{u^\top \hat{\Sigma}_1 u + u^\top \hat{\Sigma}_2 u}$$

$$\hat{\mu}_j = \frac{1}{n_j} \sum_{i:y_i=j} x_i, \; \frac{1}{n_j} \sum_{i:y_i=j} (x_i - \hat{\mu}_j)(x_i - \hat{\mu}_j)^\top$$



- **Solution:** $u \propto (\hat{\Sigma}_1 + \hat{\Sigma}_2)^{-1}(\hat{\mu}_1 - \hat{\mu}_2)$

Fisher's linear discriminant analysis

- Extension to general classification setup: y = 1, …, c
  - n_j: #sample of class j
- **Objective**

$$\arg \max_{u:\|u\|=1} \frac{u^\top \Sigma_B u}{u^\top \hat{\Sigma} u}$$

$$\hat{\Sigma}_B = \frac{1}{c} \sum_{j=1}^{c} (\hat{\mu}_j - \hat{\mu})(\hat{\mu}_j - \hat{\mu})^\top$$

Fisher's linear discriminant analysis

- Extension to general classification setup: y = 1, …, c
  - n_j: #sample of class j
- **Objective**

$$\begin{aligned}\arg\max_{u:\|u\|=1} \frac{u^\top \Sigma_B u}{u^\top \hat{\Sigma} u} &\Longleftrightarrow \arg\max_{v:\|v\|=1} \frac{v^\top \Sigma^{-1/2}\Sigma_B\Sigma^{-1/2} v}{v^\top \hat{\Sigma}^{-1/2}\hat{\Sigma}\hat{\Sigma}^{-1/2} v} \\ &= \arg\max_{v:\|v\|=1} v^\top \Sigma^{-1/2}\Sigma_B\Sigma^{-1/2} v \\ &= \arg\max_{v:\|v\|=1} v^\top \hat{\Phi} v\end{aligned}$$

$$\hat{\Phi} = \frac{1}{c}\sum_{j=1}^{c} \left(\Sigma^{-1/2}(\hat{\mu}_j - \hat{\mu})\right)\left(\Sigma^{-1/2}(\hat{\mu}_j - \hat{\mu})\right)^\top$$

Fisher's linear discriminant analysis

- Multiclass LDA reduces to sliced inverse regression with the same #sample for all classes
  - $n_j = n_{j'}$

- Hence, it also generalizes for choosing a subspace rather than a single vector in the vanilla FDA for binary classification