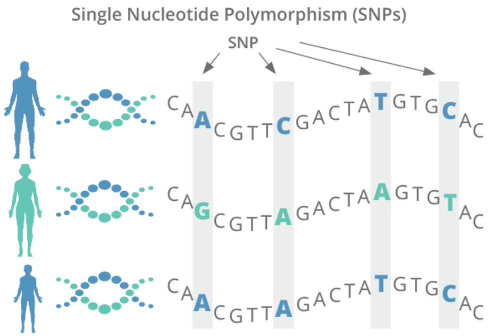
Unsupervised Dimensionality Reduction via PCA

David I. Inouye

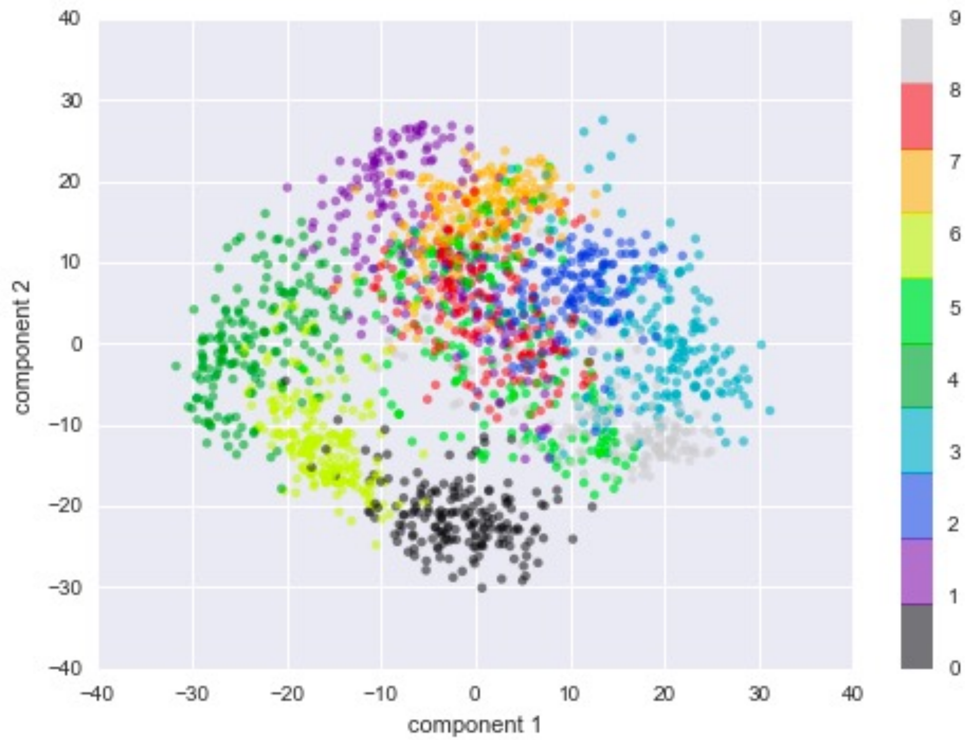
Very high-dimensional data is becoming ubiquitous

- Images (1 million pixels)
- Text (100k unique words)
- Genetics (4 million SNPs)
- Business data (12 million products)



Why dimensionality reduction? Visualization

- Allows 2D scatterplot visualizations even of high-dimensional data (2D projection of digits)

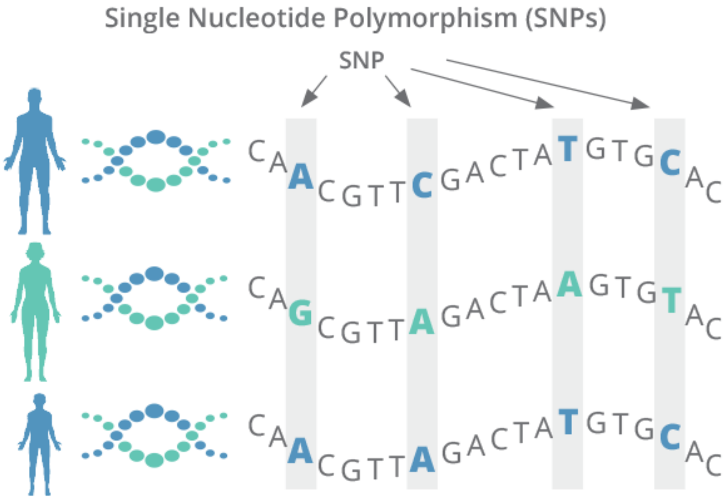


https://jakevdp.github.io/PythonDataScienceHandbook/05.09-principal-component-analysis.html

Why dimensionality reduction? Lower computation costs

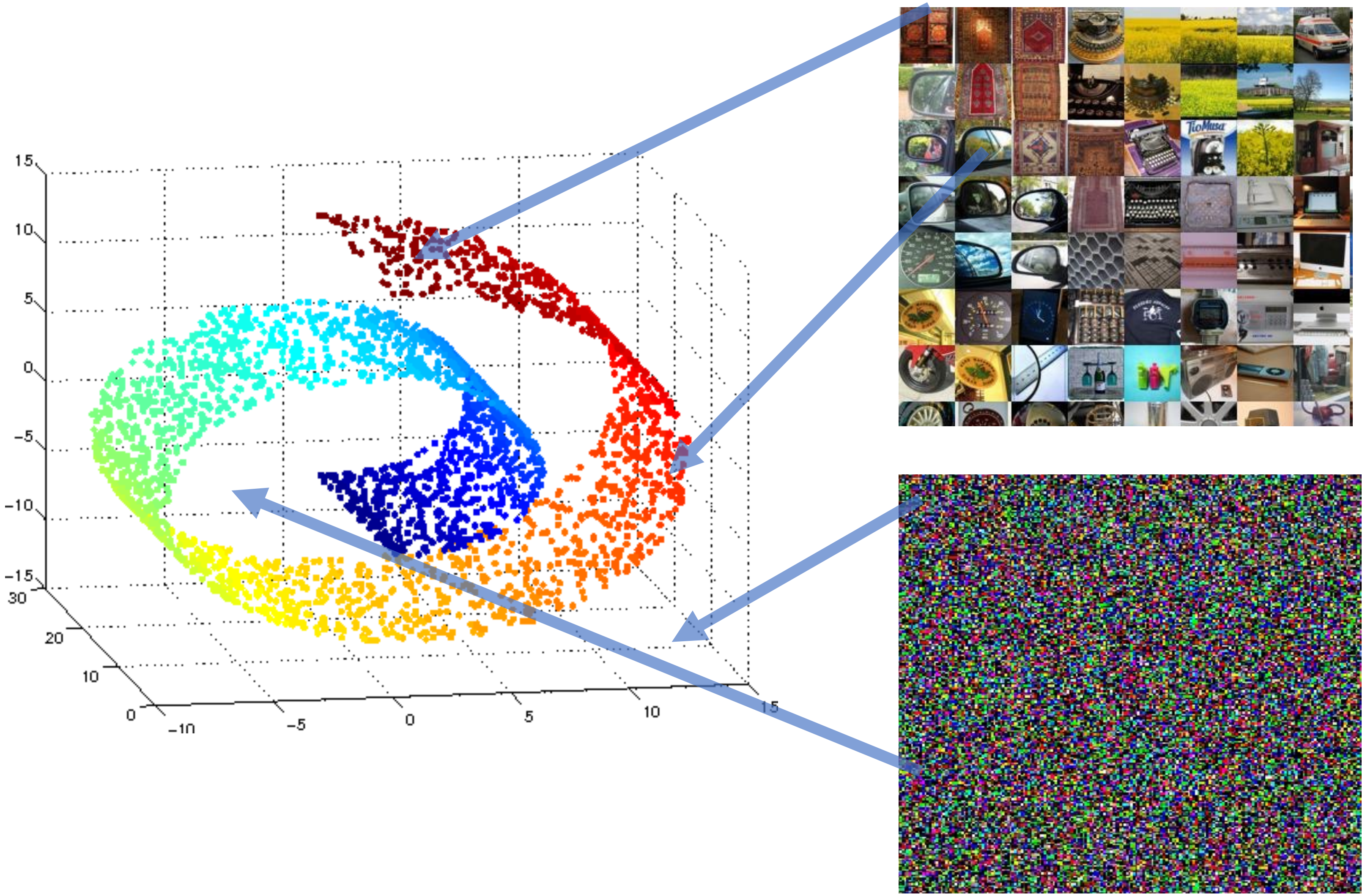
- Suppose original dimension is large like $\mathrm{d}=100000$ (e.g., images, DNA sequencing, or text)

- If we reduce to $k=100$ dimensions, the training algorithm can be sped up by $1000\times$



4-5 million SNPs in human genome.
https://www.diagnosticsolutionslab.com/tests/genomicinsight

Why dimensionality reduction?
Underlying phenomena is on lower dimensional space



Outline of Principal Components Analysis (PCA)

1. Motivation for dimensionality reduction
2. Formal PCA problem: Min reconstruction
3. Derive PCA formulation for 1D
    - Least error 1D projection is orthogonal
    - Sum over all data points
4. Solution is based on truncated SVD
5. Equivalent problem: Max variance

Math: <u>Principal Component Analysis (PCA)</u> can be formalized as minimizing the *linear* reconstruction error of the data using only $k \leq d$ dimensions

- PCA can be formalized as

$$\min_{Z \in \mathbb{R}^{n \times k}, W \in \mathbb{R}^{d \times k}} \|X_c - ZW^T\|_F^2 \quad \text{s.t. } W^T W = I_k$$

  - where

$$\mathrm{X_c} = \mathrm{X} - \mathbf{1}_n \mu_x^T \in \mathbb{R}^{n \times d} \quad \text{(centered input data)}$$

Review of linear algebra and introduction to numpy Python library

- See Jupyter notebook, which can be opened and run in Google Colab

Math: Principal Component Analysis (PCA) can be formalized as minimizing the linear reconstruction error of the data using only $k \leq d$ dimensions

- PCA can be formalized as

$$\min_{Z \in \mathbb{R}^{n \times k}, W \in \mathbb{R}^{d \times k}} \|X_c - ZW^T\|_F^2 \ \text{ s.t. } W^T W = I_k$$

  - where

$$\mathrm{X_c} = \mathrm{X} - \mathbf{1}_n \mu_x^T \in \mathbb{R}^{n \times d} \quad \text{(centered input data)}$$

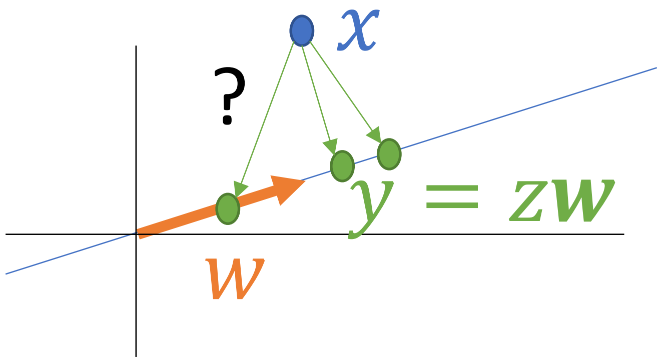
Math: Principal Component Analysis (PCA) can be formalized as minimizing the linear reconstruction error of the data using only $k \leq d$ dimensions

$$\min_{Z \in \mathbb{R}^{n \times k}, W \in \mathbb{R}^{d \times k}} \|X_c - ZW^T\|_F^2 \;\; \text{s.t. } W^T W = I_k$$

- Let's stare at this equation some more ☺
- Why is this dimensionality reduction?
- What does the orthogonal constraint mean?
- Why minimize the squared Frobenius norm?
- $\|X_c - ZW^T\|_F^2 = \sum_{i=1}^n \left\|\boldsymbol{x}_i^T - \boldsymbol{z}_i^T W^T\right\|_2^2 = \sum_{i=1}^n \|\boldsymbol{x_i} - W\boldsymbol{z}_i\|_2^2$
- For analysis, let's simplify to a single dimension (i.e., $k = 1$)
  - $\sum_{i=1}^n \|\boldsymbol{x_i} - z_i \boldsymbol{w}\|_2^2$ where z_i is a scalar

What is the best projection given a fixed subspace (line in 1D case)?

- If we are given $\boldsymbol{w}$, what is the best z (i.e. minimum reconstruction error) for a given $\boldsymbol{x}$?
  - $\min_z \|\boldsymbol{x} - z\boldsymbol{w}\|_2^2$



- The orthogonal projection!
  - $z = \boldsymbol{x}^T \boldsymbol{w} = \|\boldsymbol{x}\| \|\boldsymbol{w}\| \cos\theta = \|\boldsymbol{x}\| \cos\theta$
  - $z = \|\boldsymbol{x}\| \cos\theta = \mathrm{hyp} \cdot \frac{\mathrm{adj}}{\mathrm{hyp}} = \mathrm{adj}$
  - $z\boldsymbol{w}$ is a scaled vector along the line defined by $\boldsymbol{w}$

Thus, we can simplify to only minimizing over W

$$\min_{\boldsymbol{z},\boldsymbol{w}:\|\boldsymbol{w}\|_2=1} \sum_{i=1}^{n} \|\boldsymbol{x_i} - z_i \boldsymbol{w}\|_2^2 = \min_{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1} \sum_{i=1}^{n} \left\|\boldsymbol{x_i} - \left(\boldsymbol{x}_i^T \boldsymbol{w}\right)\boldsymbol{w}\right\|_2^2$$

- Now we can return to the Frobenius norm:

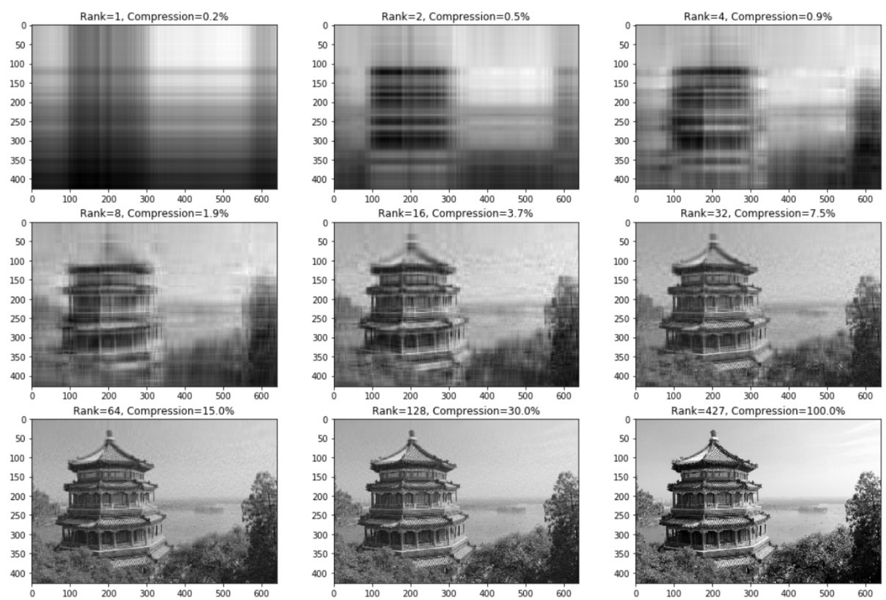
$$\min_{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1} \left\|X_c - \boldsymbol{z}\boldsymbol{w}^{\boldsymbol{T}}\right\|_F^2 \text{ where } \boldsymbol{z} = X_c \boldsymbol{w}$$

- What is $\boldsymbol{z}\boldsymbol{w}^{\boldsymbol{T}}$? Have we seen something like this before?
- This is the best rank-1 approximation to X_c, which is given by the SVD!
  - $\boldsymbol{w} = \boldsymbol{v}_1$ and $\boldsymbol{z} = \sigma_1 \boldsymbol{u}_1$, where $\sigma_1, \boldsymbol{u}_1, \boldsymbol{v}_1$ are the first singular value, left singular vector and right singular vector respectively.

For $k \geq 1$, the PCA solution is the top k right singular vectors

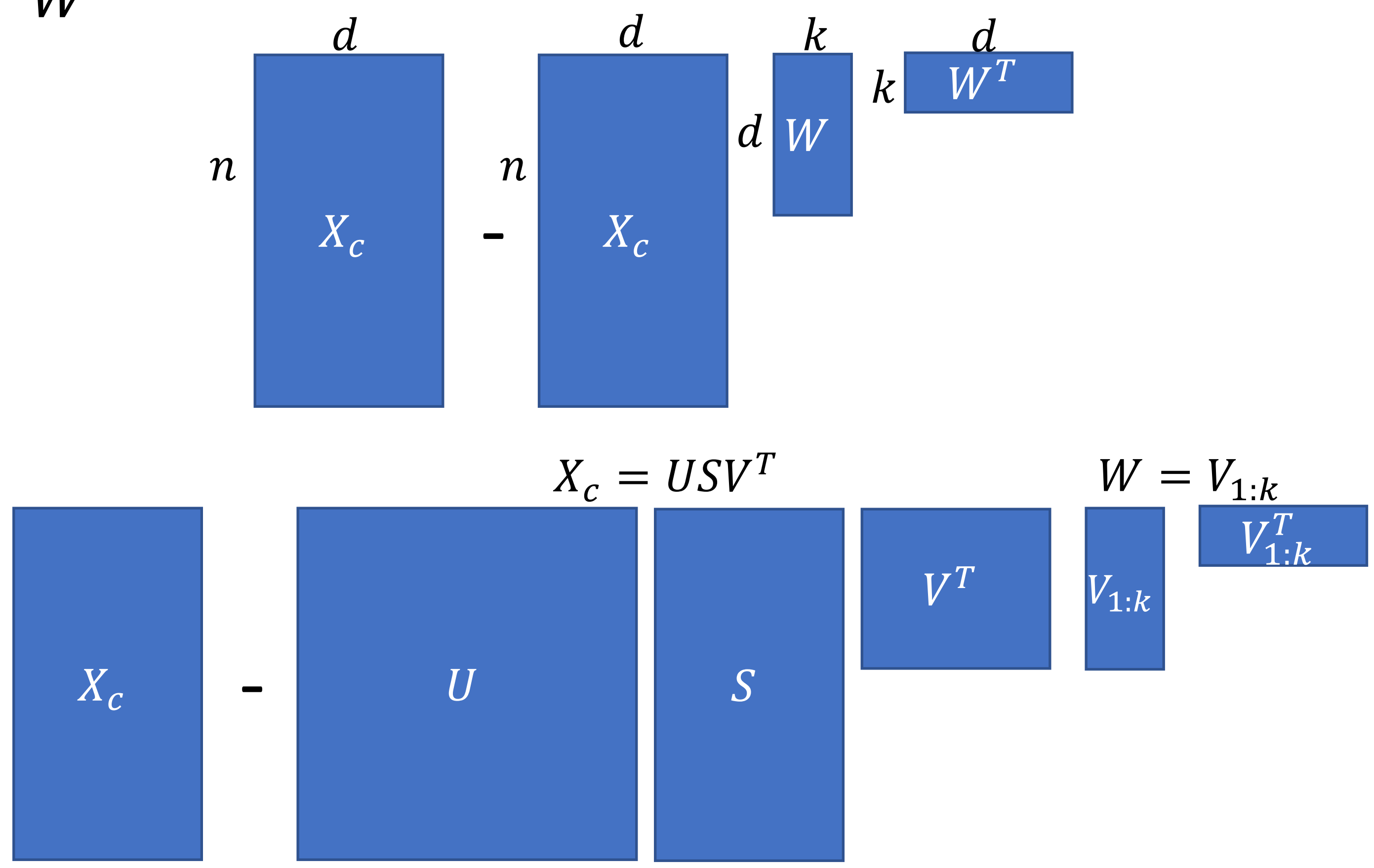
- If $X_c = USV^T$, then the general solution is
$$W^* = V_{1:k}$$

- Remember: SVD is best k dim. approximation



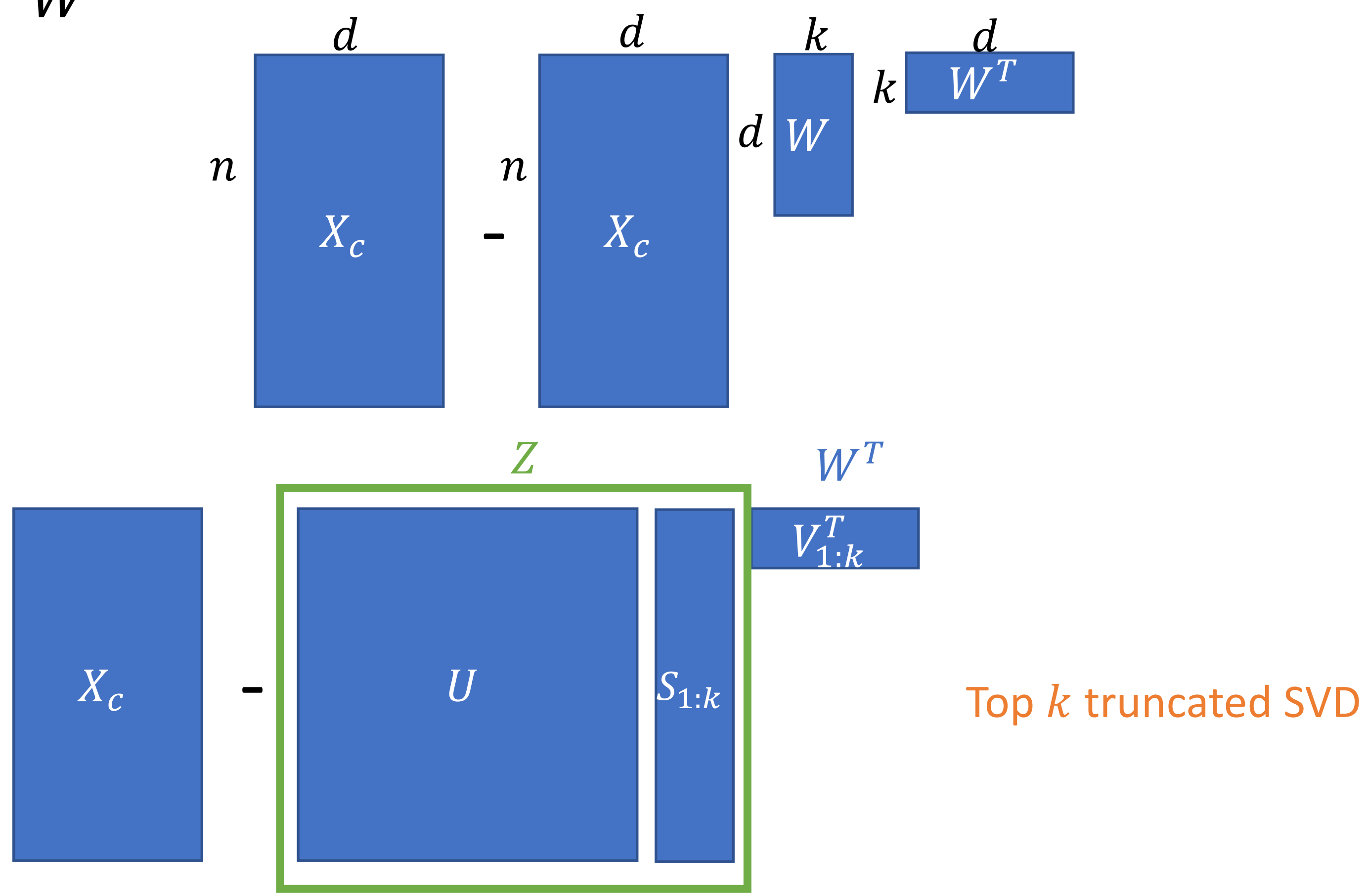
Check: The solution reveals the truncated SVD as best approximation

$$\min_W \|X_c - (X_c W)W^T\|_F^2 \quad \text{s.t.} \quad W^T W = I$$



Check: The solution reveals the truncated SVD as best approximation

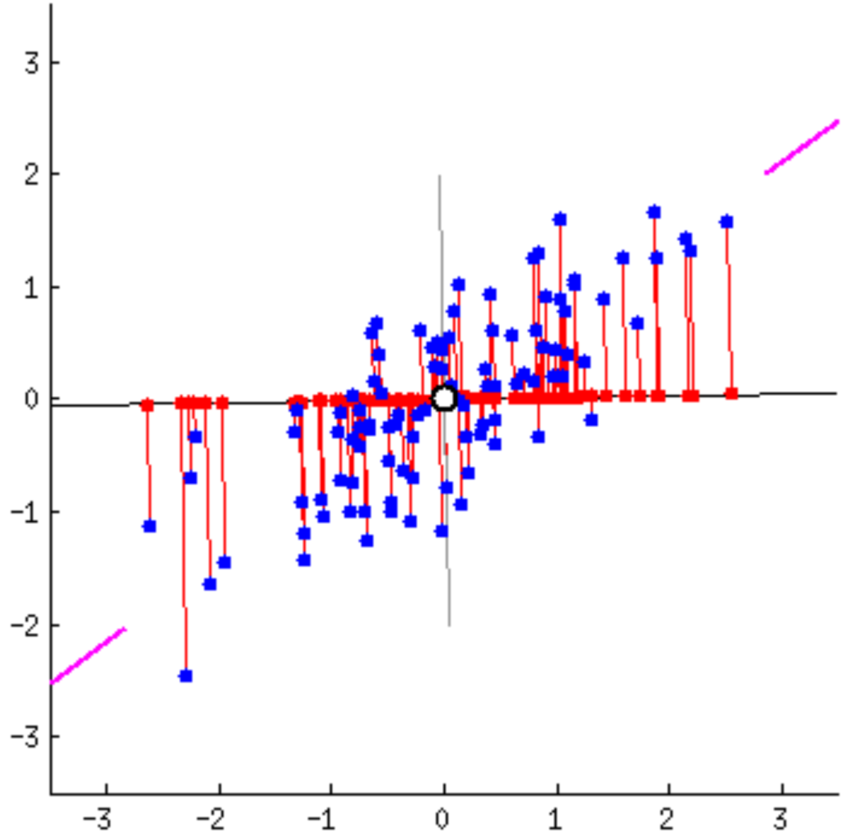
$$\min_W \|X_c - (X_c W)W^T\|_F^2 \;\; \text{s.t.} \;\; W^T W = I$$



Intuition: Principal component analysis finds the **best linear projection** onto a lower-dimensional space

$$\min_{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1} \left\| X_c - \boldsymbol{z}\boldsymbol{w}^{\boldsymbol{T}} \right\|_F^2$$

$$\text{where } \boldsymbol{z} = X_c \boldsymbol{w}$$



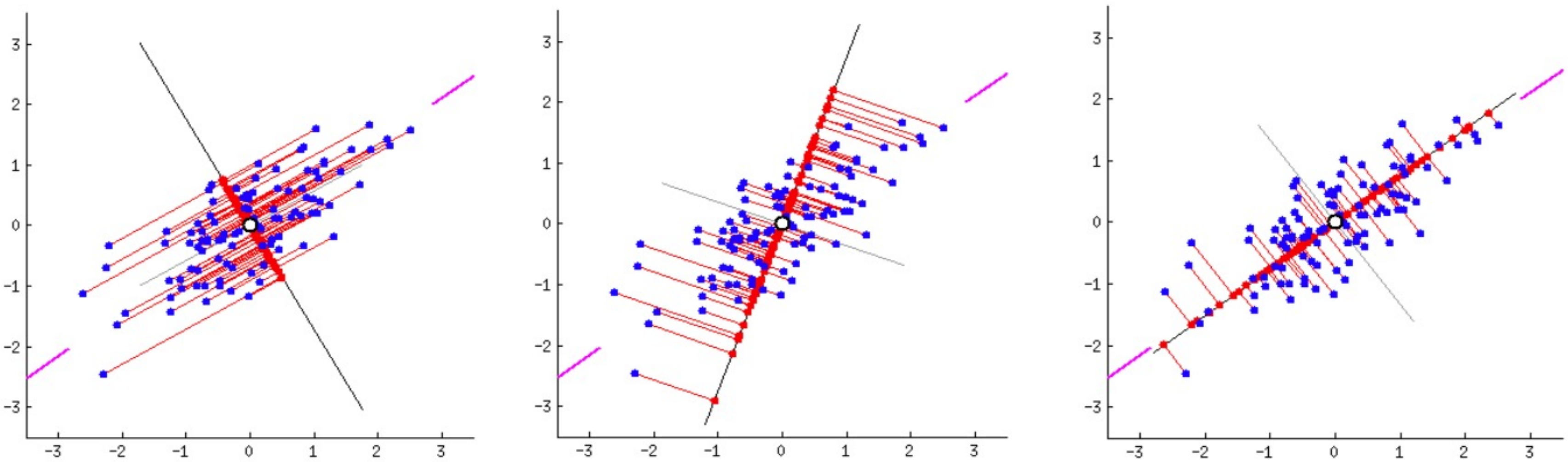
2D to 1D projection: Red lines show the projection error onto 1D lines. PCA finds the line that has the smallest projection error (in this example, when it aligns with the purple).

https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

Minimizing reconstruction error (red lines) is equivalent to maximizing the variance of projection (spread of red points)

Max reconstruction error
Min variance

Min reconstruction error
Max variance



$$\underset{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1}{\operatorname{argmax}} \left\|X_c - \boldsymbol{z}\boldsymbol{w}^T\right\|_F^2$$

$$= \underset{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1}{\operatorname{argmin}} \|z\|_2^2 = \underset{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1}{\operatorname{argmin}} \sigma_z^2$$

where $\boldsymbol{z} = X_c \boldsymbol{w}$

$$\underset{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1}{\operatorname{argmin}} \left\|X_c - \boldsymbol{z}\boldsymbol{w}^T\right\|_F^2$$

$$= \underset{\boldsymbol{w}:\|\boldsymbol{w}\|_2=1}{\operatorname{argmax}} \sigma_z^2$$

Derivation of equivalence will require 3 facts

1. Squared Frobenius norm is trace of matrix product (generalizes $\|x\|_2^2 = x^T x$):
   - $\|A\|_F^2 = \mathrm{Tr}(A^T A)$
2. If $A \in \mathbb{R}^{n \times d}$ is a *centered* data matrix, then the Frobenius norm is the scaled sum of 1D variances:
   - $\|A\|_F^2 = \mathrm{Tr}(A^T A) = n\mathrm{Tr}(\hat{\Sigma}) = n \sum_j \sigma_j^2$
   - where $\hat{\Sigma}$ is the empirical covariance matrix and σ_j^2 is variance for the j-th dimension
3. Optimization solutions are invariant when the objective is multiplied by **positive** constant or a constant is added,
   - $\operatorname*{argmin}_W f(W) = \operatorname*{argmin}_W af(W) + b, \quad \forall a > 0, b \in \mathbb{R}$

The PCA objective can be decomposed into the original variance minus the variance of projection

- Minimize reconstruction error
$$\min_{W:W^TW=I_k} \|X_c - (X_cW)W^T\|_F^2$$
- $\|X_c - X_cWW^T\|_F^2$
- $= \mathrm{Tr}[(X_c - X_cWW^T)^T(X_c - X_cWW^T)]$
- $= \mathrm{Tr}[(X_c^T - WW^TX_c^T)(X_c - X_cWW^T)]$
- $= \mathrm{Tr}[X_c^TX_c - WW^TX_c^TX_c - X_c^TX_cWW^T + WW^TX_c^TX_cWW^T]$
- $= \mathrm{Tr}[X_c^TX_c] - \mathrm{Tr}[WW^TX_c^TX_c] - \mathrm{Tr}[X_c^TX_cWW^T] + \mathrm{Tr}[WW^TX_c^TX_cWW^T]$
- $= \mathrm{Tr}[X_c^TX_c] - \mathrm{Tr}[W^TX_c^TX_cW] - \mathrm{Tr}[W^TX_c^TX_cW] + \mathrm{Tr}[W^TX_c^TX_cWW^TW]$
- $= \mathrm{Tr}[X_c^TX_c] - \mathrm{Tr}[W^TX_c^TX_cW] - \mathrm{Tr}[W^TX_c^TX_cW] + \mathrm{Tr}[W^TX_c^TX_cW]$
- $= \mathrm{Tr}[X_c^TX_c] - \mathrm{Tr}[(X_cW)^TX_cW]$
- $= \mathrm{Tr}[X_c^TX_c] - \mathrm{Tr}[Z^TZ]$
- $= n\sum_{j=1}^{d} \sigma_{x,j}^2 - n\sum_{j=1}^{k} \sigma_{z,j}^2$

Equivalence is derived by manipulating optimization problem

- $\underset{W:W^TW=I_k}{\operatorname{argmin}} \|X_c - (X_c W)W^T\|_F^2$
- $= \underset{W:W^TW=I_k}{\operatorname{argmin}} n\sum_{j=1}^{d} \sigma_{x,j}^2 - n\sum_{j=1}^{k} \sigma_{z,j}^2$
- $= \underset{W:W^TW=I_k}{\operatorname{argmin}} -\sum_{j=1}^{k} \sigma_{z,j}^2$
- $= \underset{W:W^TW=I_k}{\operatorname{argmax}} \sum_{j=1}^{k} \sigma_{z,j}^2$
- This last one is exactly maximizing the variance along the projected dimensions of $\boldsymbol{z}$

Equivalent solutions: The solution to both problems is the top k right singular vectors of X_c

- Minimize reconstruction error
$$\min_{W:W^TW=I_k} \|X_c - (X_cW)W^T\|_F^2$$
  - Singular value decomposition (SVD) of $X_c = USV^T$
  - Solution: $W^* = V_{1:k}$

- Maximize variance of latent projection (equivalent solution)
$$\max_{W:W^TW=I_k} \sum_{j=1}^{k} \sigma_{z,j}^2$$
  - Equivalent solution is the eigenvectors of $X_c^T X_c = n\hat{\Sigma}$
    - $X_c^T X_c = (USV^T)^T(USV^T) = (VSU^T)(USV^T) = VS(U^TU)SV^T = VS^2V^T = Q\Lambda Q^T$
  - Solution: $W^* = Q_{1:k} \equiv V_{1:k}$!

Recap: Principal Components Analysis (PCA)

1. Motivation for dimensionality reduction
2. Formal PCA problem: Min reconstruction
3. Derive PCA formulation for 1D
   - Least error 1D projection is orthogonal
   - Sum over all data points
4. Solution is based on truncated SVD
5. Alternative viewpoint: Max variance
   - Derive equivalence
   - Derive equivalent solutions

Demo of PCA via sklearn (time permitting)

- Random projections vs PCA projections
- Visualizations of
  - Minimum reconstruction error
  - Maximum variance
  - Explained variance based on k
- Code examples
  - Digits
  - Eigenfaces

Questions?