Dimensionality Reduction with PCA

CS 556

High Dimensional Data

- Nowadays data are high dimensional
- Example
  - 300x300 image, each pixel is a tuple (Red, Green, Blue)
  - House price datasets can contains tens or hundreds of features

Challenges of High Dimensional Data

- Hard to analyze
- Interpretation is difficult
- Impossible visualization
- Computationally expensive
- Lie on lower dimensional space

Statistical Concepts Review
Mean

Mean denoted by μ is the average value in a collection of numbers.

$$\boxed{\mu = \frac{1}{n}\sum_{i=1}^{n} x_i}$$

$$X = \{3,7,5\}$$

$$\mu = \frac{3+7+5}{3} = 5$$

Statistical Concepts Review
Variance

Variance denoted by σ^2 is a statistical measurement of the speed between the number in a data set.

$$\boxed{\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2, \ \mu = E[X]}$$

$$X = \{3,7,5\}$$

$$\mu = \frac{3+7+5}{3} = 5$$

$$\sigma^2 = \frac{1}{3}((3-5)^2 + (7-5)^2 + (5-5)^2) = \frac{8}{3}$$

Statistical Concepts Review
Covariance

Covariance is a statistical measure of the strength and sign of the linear relationship between two variables in the scale of the original data.

$$Cov[X, Y] = \frac{1}{n-1} \sum_{i=1}^{n} [(x - \mu_x)(y - \mu_y)]$$

Covariance Matrix

$$\begin{bmatrix} Var[X] & Cov[X,Y] \\ Cov[Y,X] & Var[Y] \end{bmatrix}$$

How to Construct Covariance Matrices

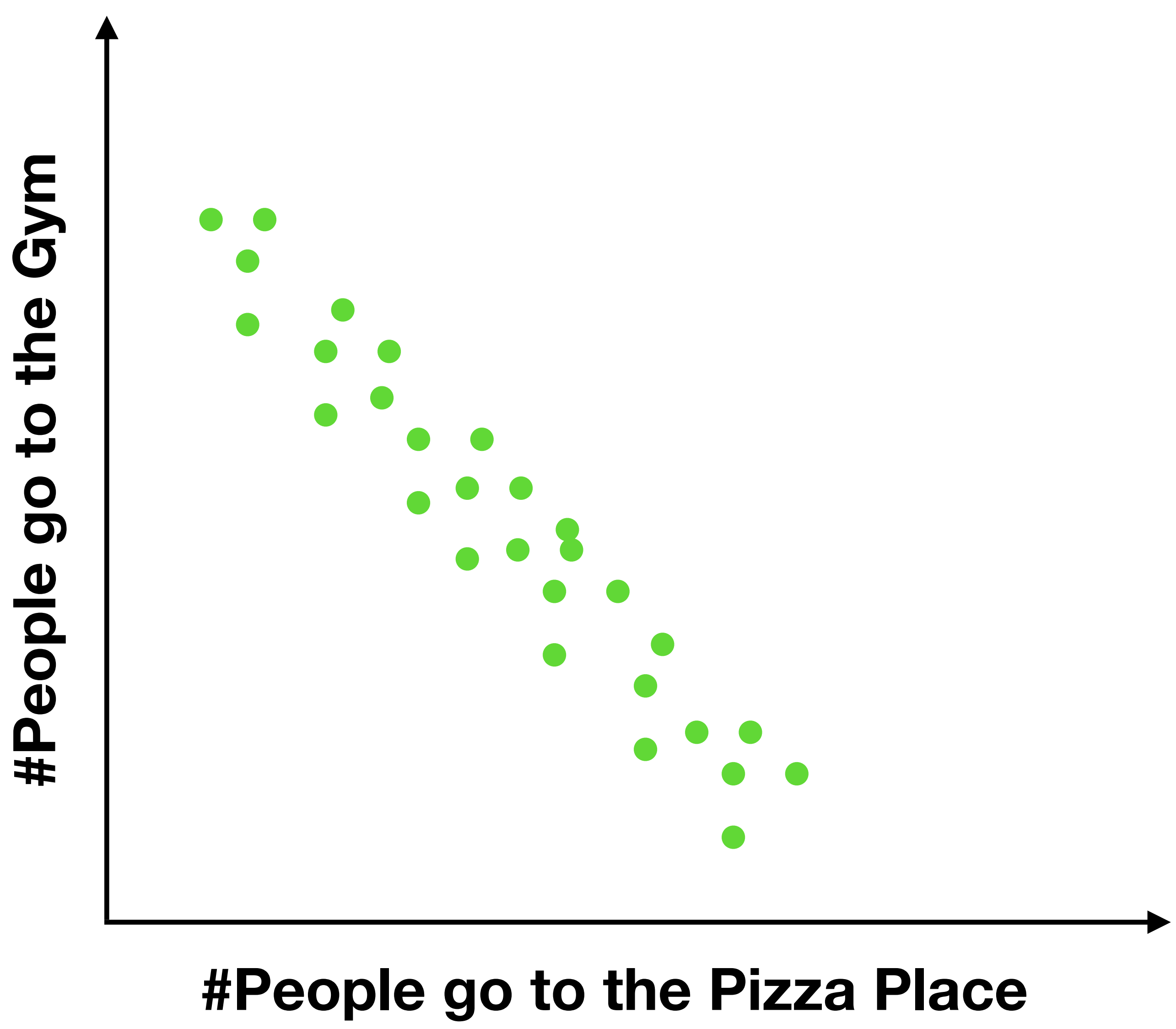
Assume we have the following dataset:

	Study Time(ST)	Exam Score (ES)
Student 1	10	90
Student 2	6	68
Student 3	20	95

$$X = \begin{bmatrix} 10 & 90 \\ 6 & 68 \\ 20 & 95 \end{bmatrix}$$

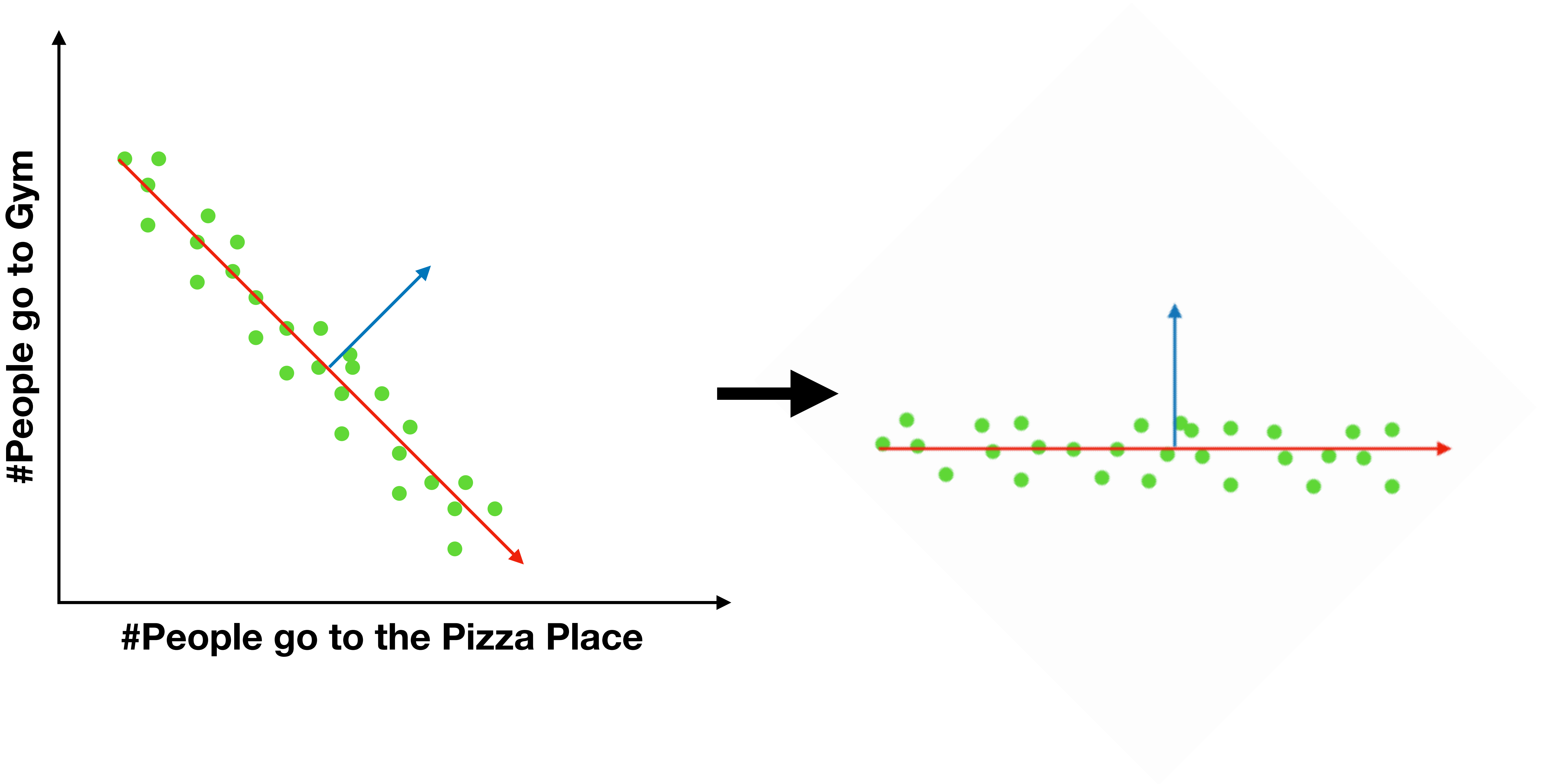
$$COV = \frac{1}{n-1} X^T X = \begin{bmatrix} 10 & 6 & 20 \\ 90 & 68 & 95 \end{bmatrix} \begin{bmatrix} 10 & 90 \\ 6 & 68 \\ 20 & 95 \end{bmatrix} = \begin{bmatrix} Var(ST) & Cov(ST, ES) \\ Cov(ES, ST) & Var(ES) \end{bmatrix}$$

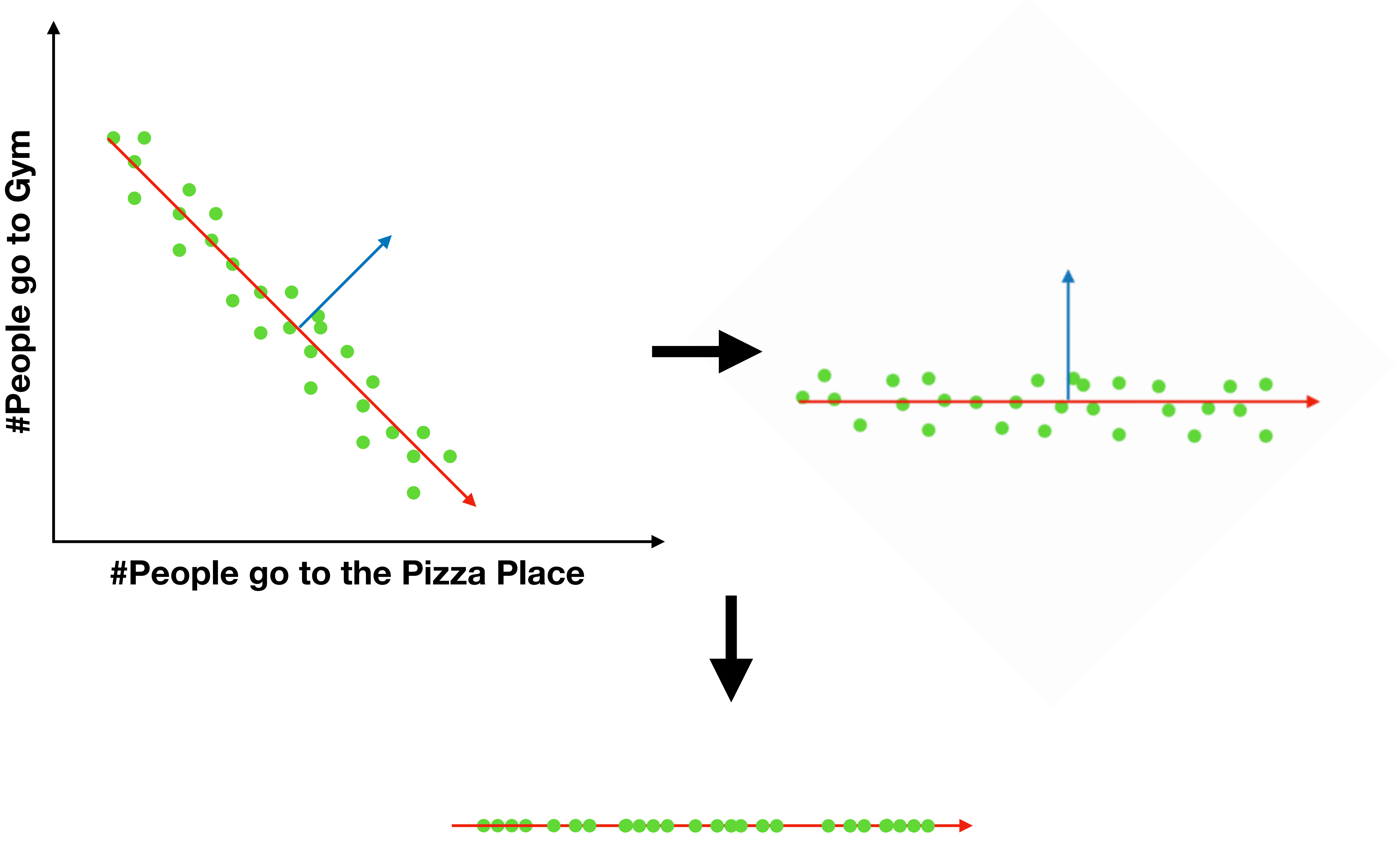

GYM


#People go to the Gym
#People go to the Pizza Place


#People go to Gym
#People go to the Pizza Place

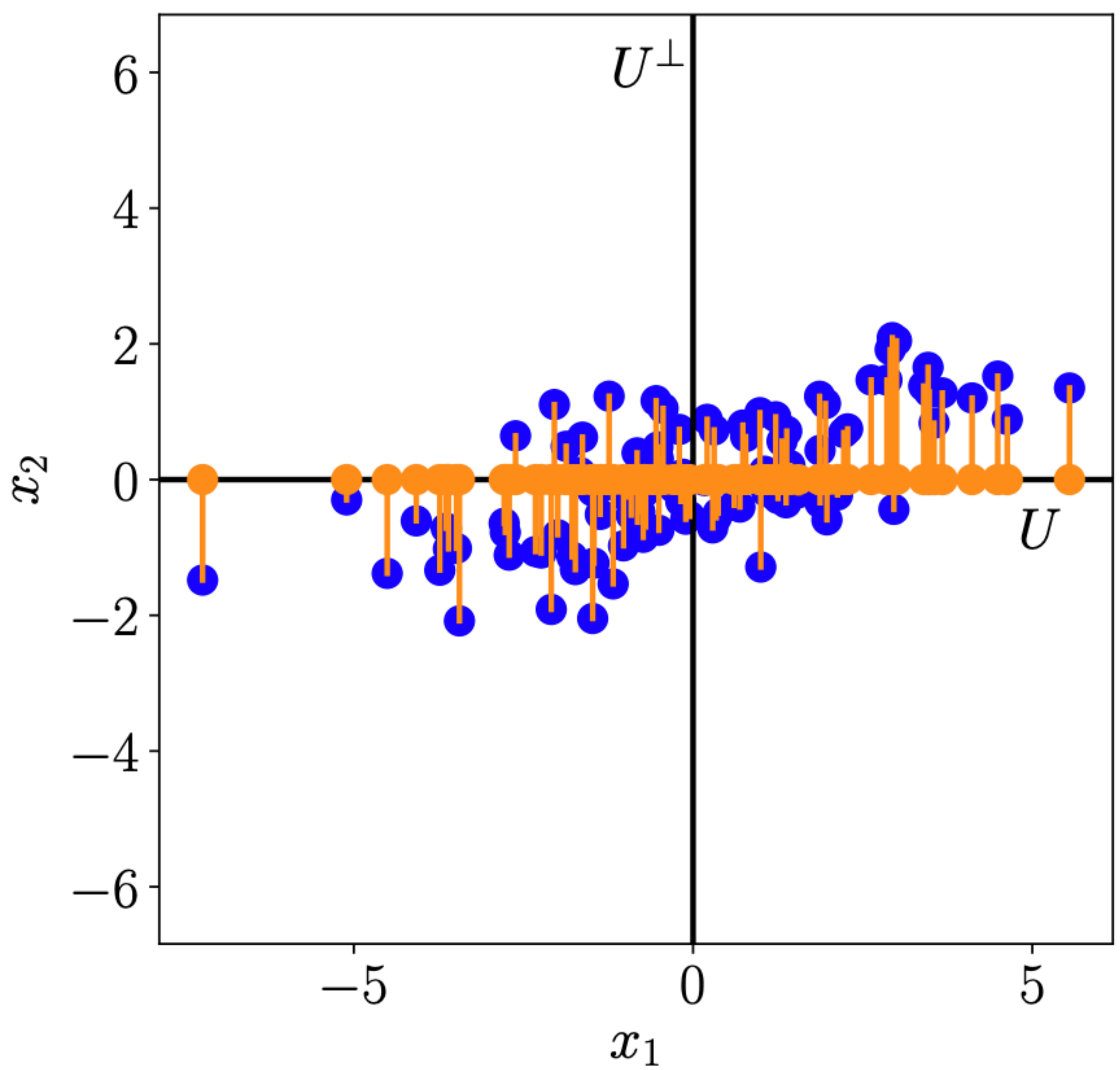

#People go to Gym
#People go to the Pizza Place


#People go to Gym
#People go to the Pizza Place


#People go to Gym
#People go to the Pizza Place

PCA: Key Idea

Use orthogonal projections to find lower dimensional representations of data that retain as much information as possible.



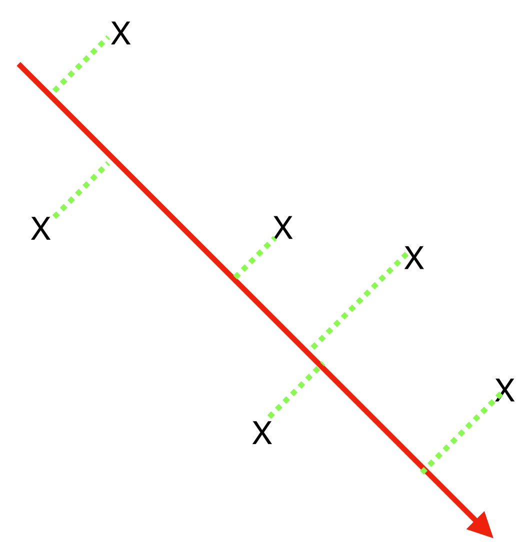
Why PCA?

- Visualize data in a lower-dimensional space
- Understand the sources of variability in the data
- Understand correlations between different coordinates of the data points

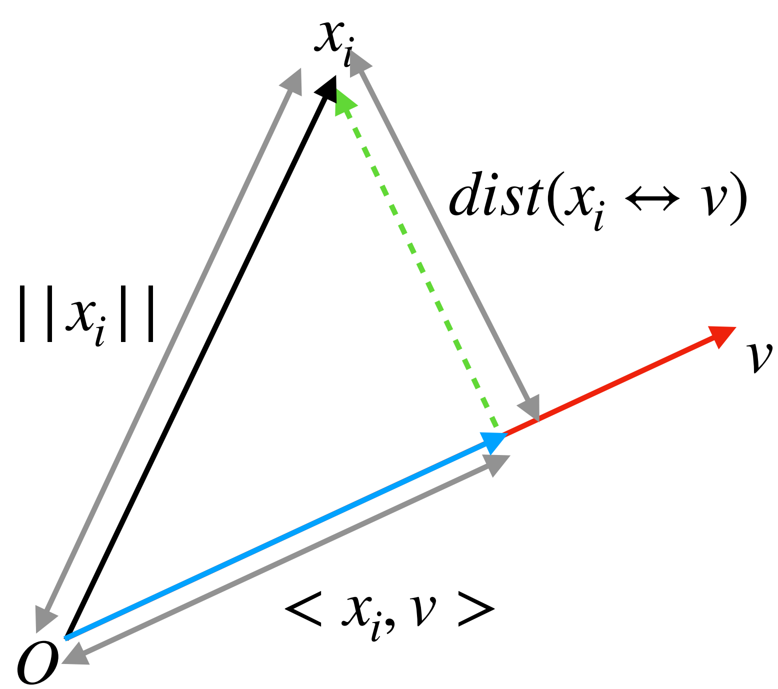
Objective Function

For a given data set and parameter k, the goal of PCA is to compute the **k-dimensional subspace** that minimizes the average squared distance between the points and the subspace.

$$\underset{k-dim\ spaces\ S}{argmin}\ \frac{1}{n}\sum_{i=1}^{n}((length\ of\ x_i's\ projection\ on\ S)^2)$$



Objective Function



$$\underset{\mathbf{v}\,:\,\|\mathbf{v}\|=1}{\operatorname{argmin}} \frac{1}{m} \sum_{i=1}^{m} \left((\text{distance between } \mathbf{x}_i \text{ and line spanned by } \mathbf{v})^2 \right)$$

$$(\operatorname{dist}(\mathbf{x}_i \leftrightarrow \text{line}))^2 + \langle \mathbf{x}_i, \mathbf{v} \rangle^2 = \|\mathbf{x}_i\|^2$$

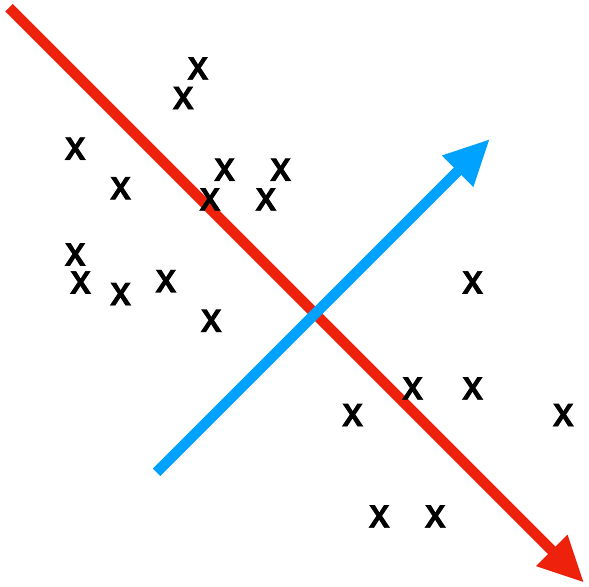
$$\underset{\mathbf{v}\,:\,\|\mathbf{v}\|=1}{\operatorname{argmax}} \frac{1}{m} \sum_{i=1}^{m} \langle \mathbf{x}_i, \mathbf{v} \rangle^2$$

Objective Function

Given $x_1, \ldots, x_m \in \mathbb{R}^D$ and a parameter $k \geq 1$, compute orthonormal vectors $v_1, \ldots, v_k \in \mathbb{R}^D$ to maximize:

$$\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{k} \langle x_i, v_j \rangle^2$$

The resulting k orthonormal vectors are called the top k principal components of the data.



Which is the best principle component?

How to find principle Components? (Summary of Preliminaries)

- Given a dataset $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m \mid \mathbf{x}_i \in \mathbb{R}^D\}$
- **Goal**: Represent each instance in space $\mathbb{R}^M$ such that M<D
- We usually `set` M before running the procedure.
- Mean: $\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$
- Covariance: $S = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x} - \overline{\mathbf{x}})(\mathbf{x} - \overline{\mathbf{x}})^T$
- Matrix *S* is symmetric and positive semi-definite (i.e., all eigenvalues are non-negative)

How to find principle Components? 1D Projection illustration

- Given a dataset $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m \mid \mathbf{x}_i \in \mathbb{R}^D\}$
- Consider some vector $\mathbf{u}_1 \in \mathbb{R}^D$ and a datapoint $\mathbf{x}_i \in \mathbb{R}^D$.
- We can project $\mathbf{x}_i$ onto $\mathbf{u}_1$ with the scalar $\mathbf{u}_i^T\mathbf{x}_i$ (by projection geometry)
- Similarly the *mean* projection is given by $\mathbf{u}_i^T\overline{\mathbf{x}}$
- We are only interested in the *direction* of $\mathbf{u}_i$ hence let $||\mathbf{u}_i|| = 1$
- Calculate variance of *projected* data.

$$\frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}_1^T\mathbf{x}_i - \mathbf{u}_1^T\overline{\mathbf{x}})^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}_1^T(\mathbf{x}_i - \overline{\mathbf{x}}))^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}_1^T(\mathbf{x}_i - \overline{\mathbf{x}})^2\mathbf{u}_1)$$

$$\frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}_1^T(\mathbf{x}_i - \overline{\mathbf{x}})^2\mathbf{u}_1) = \frac{1}{N}\sum_{i=1}^{N}\mathbf{u}_1^T(\mathbf{x}_i - \overline{\mathbf{x}})^T(\mathbf{x}_i - \overline{\mathbf{x}})\mathbf{u}_1 = \mathbf{u}_1^T\boxed{\left(\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})^T(\mathbf{x}_i - \overline{\mathbf{x}})\right)}\mathbf{u}_1$$

Covariance Matrix *S*

$$\mathbf{u}_1^T\left(\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})^T(\mathbf{x}_i - \overline{\mathbf{x}})\right)\mathbf{u}_1 = \mathbf{u}_1^T S\mathbf{u}_1$$

Goal: Find some $\mathbf{u}_1$ such that $\mathbf{u}_1^T S\mathbf{u}_1$ is maximized

Adapted from: UvA - Lectures and Slides for the UvA Master AI Course Machine Learning 1

How to find principle Components? 1D Projection illustration

- Goal: Find some $\mathbf{u}_1$ such that $\mathbf{u}_1^T S \mathbf{u}_1$ is maximized.

$$\mathbf{u}_1^T \left(\frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}_i - \overline{\mathbf{x}})^T (\mathbf{x}_i - \overline{\mathbf{x}}) \right) \mathbf{u}_1 = \mathbf{u}_1^T S \mathbf{u}_1$$

$$\text{argmax}_{\mathbf{u}_1} \mathbf{u}_1^T S \mathbf{u}_1 \quad \text{s.t.}\ \mathbf{u}_1^T \mathbf{u}_1 = 1 \qquad -(1)$$

- Solve (1) via the *method of Lagrange Multipliers*

$$\mathbf{L}(\mathbf{u}_1, \lambda_1) = \mathbf{u}_1^T S \mathbf{u}_1 \quad + \lambda_1 (\mathbf{u}_1^T \mathbf{u}_1 - 1) \quad -(2)$$

- To optimize (2), set derivative of **L** w.r.t $\mathbf{u}_1$ to zero.

$$\frac{\partial L(\mathbf{u}_1, \lambda_1)}{\partial \mathbf{u}_1} = S \mathbf{u}_1 - \lambda_1 \mathbf{u}_1 = 0$$

$$S \mathbf{u}_1 = \lambda_1 \mathbf{u}_1 \qquad -(3)$$

- Expression (3) implies that $\mathbf{u}_1$ and λ_1 are the *eigenvector* and corresponding *eigenvalue* respectively of $S \in \mathbb{R}^{D \times D}$

Principal component

Variance of projected data

$$\mathbf{u}_1^T S \mathbf{u}_1 = \lambda_1$$

Intuition

argmax expression in (1) implies that we search for the eigenvector $\mathbf{u}_1$ with the largest eigenvalue

How to find principle Components? k-D Projection illustration

- Repeat same procedure for *M* components to get $U_M = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M\}$

- **PCA Procedure**: Compute $\bar{\mathbf{x}} \in \mathbb{R}^{D\times 1}$, *S* and eigen-decomposition of *S* to get $\mathbf{U}_M \in \mathbb{R}^{M\times M}$

- **Projection**: For some new data point $\mathbf{x}_i \in \mathbb{R}^{D\times 1}$, $\mathbf{x}_i^{proj} = U_M^T(\mathbf{x} - \bar{\mathbf{x}})$ where $\mathbf{x}_i^{proj} \in \mathbb{R}^{M\times 1}$

- The *M* eigenvectors of *S* in $\mathbf{U}_M$ are the **principal components** and are ordered in decreasing order of eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M$

- Total variance of projected data $\sum_{i=1}^{M} \lambda_i$

- In practice, eigen-decomposition is $O(D^3)$ hence we employ SVD which is $O(MD^2)$ to obtain U_M .

Finding principal components

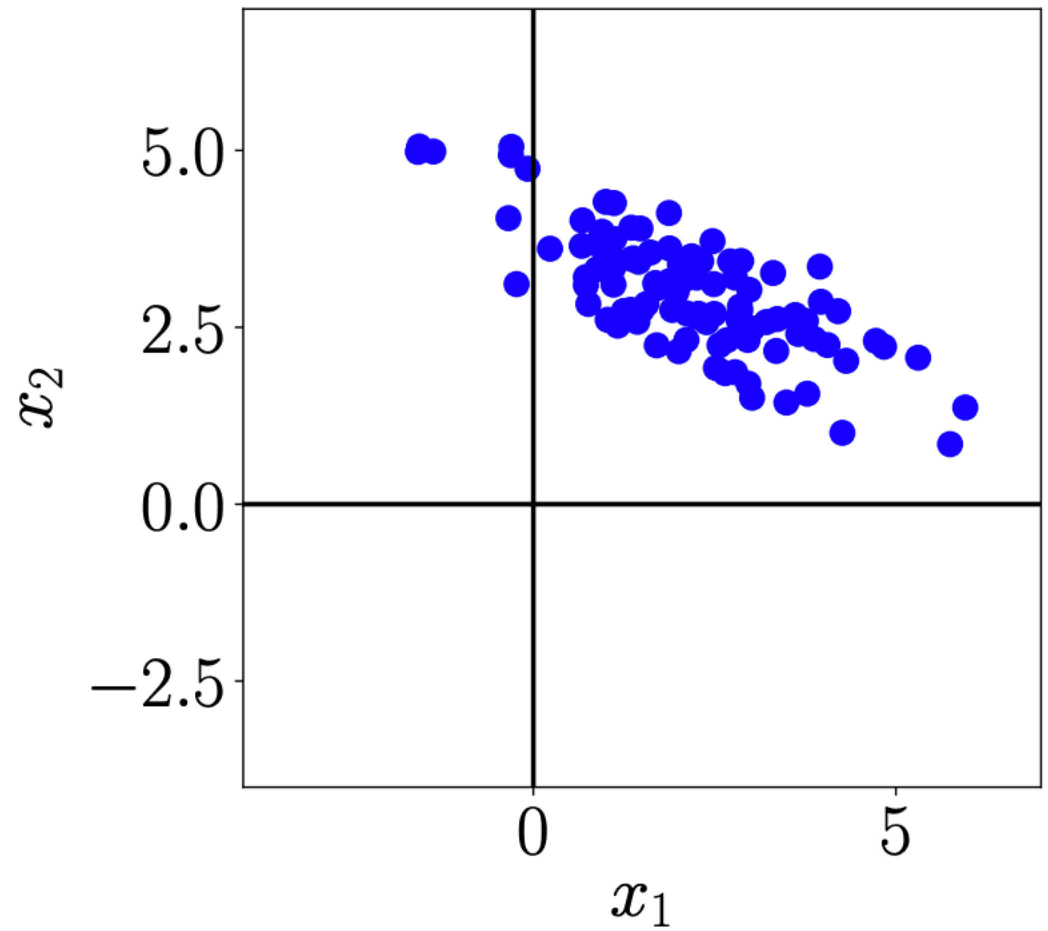
1. Calculate the eigenvalues and unit eigenvectors of the covariance matrix and order the eigenvectors in descending order with respect to the corresponding eigenvalues.

2. The unit eigenvectors of the covariance matrix represent the principle components of the data. The corresponding eigenvalues give the variance of the principle components.

Theorem

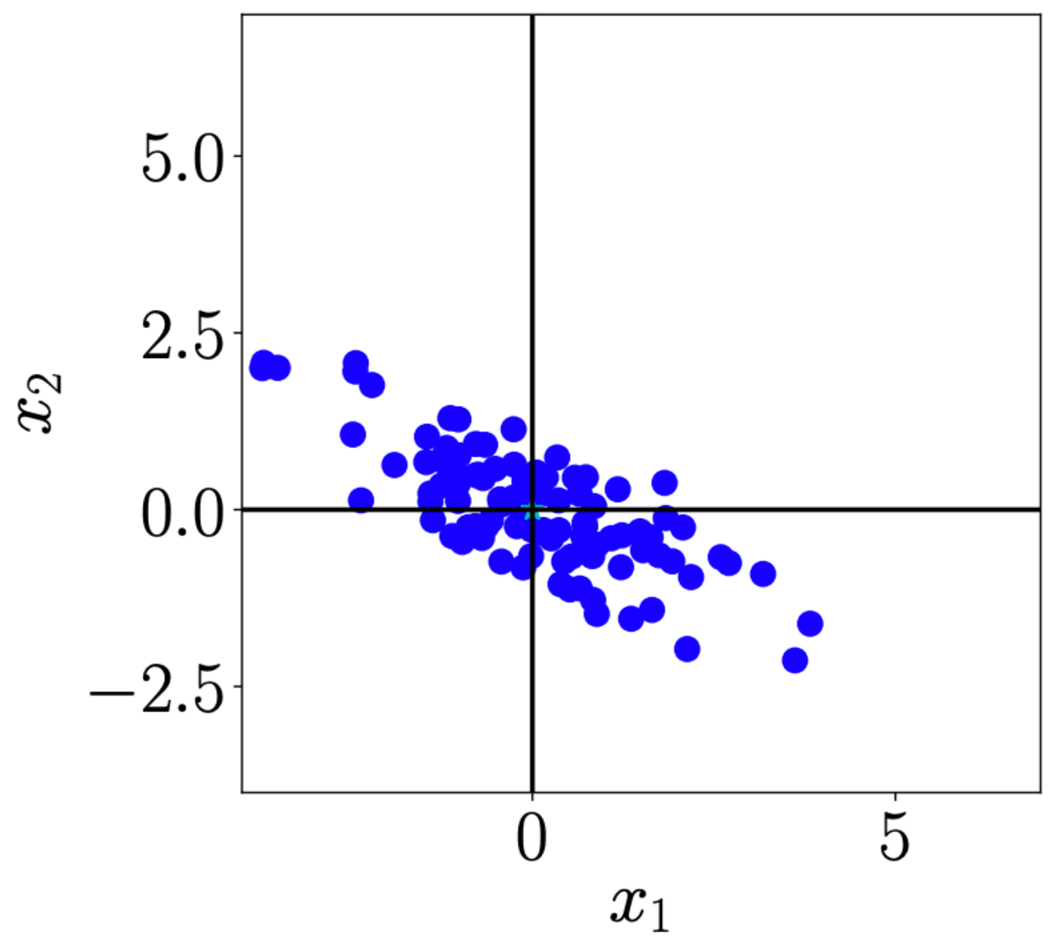
Let A be a covariance matrix, then A is orthogonally diagonalizable and Q is the diagonalizing matrix formed by the unit eigenvectors u_1, u_2, .., u_n of A as rows. Furthermore, the coordinates y of each observation x with respect to u_1, u_2, .. , u_n is give by y = Qx.

PCA in Practice



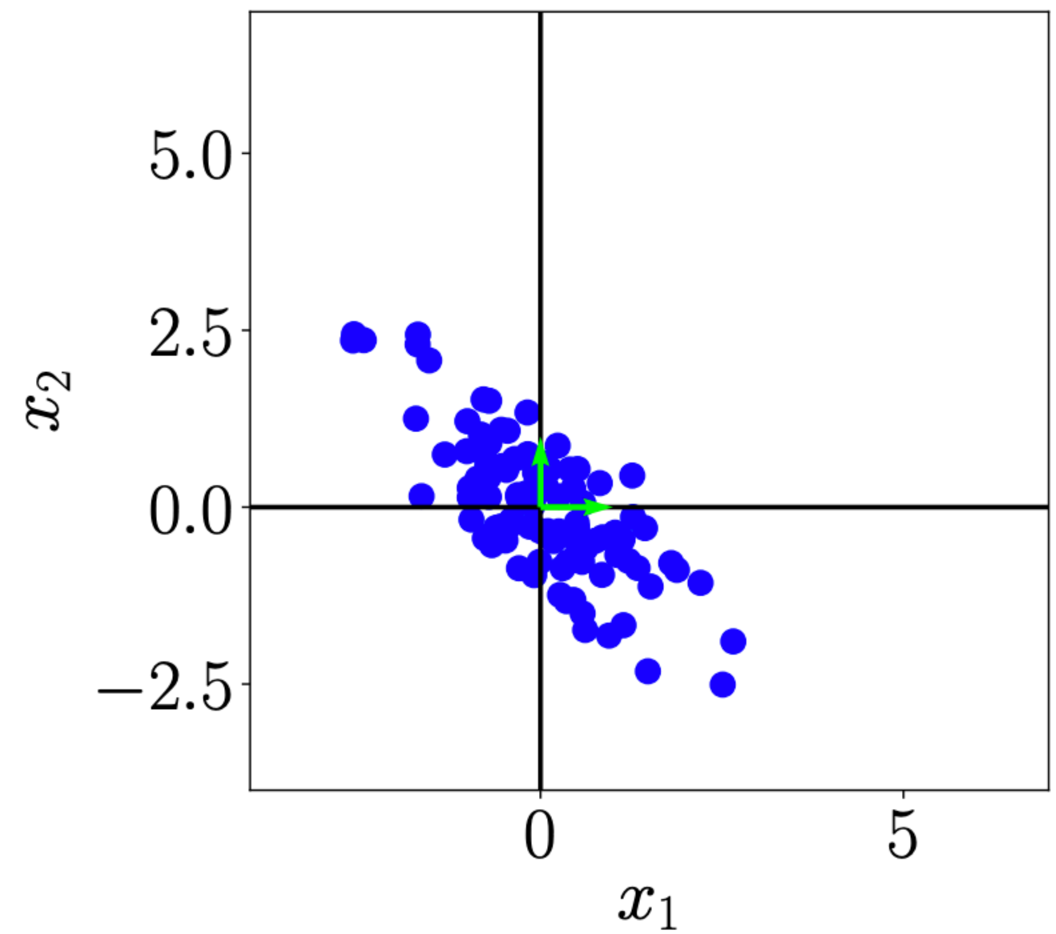
(a) Original dataset.

PCA in Practice



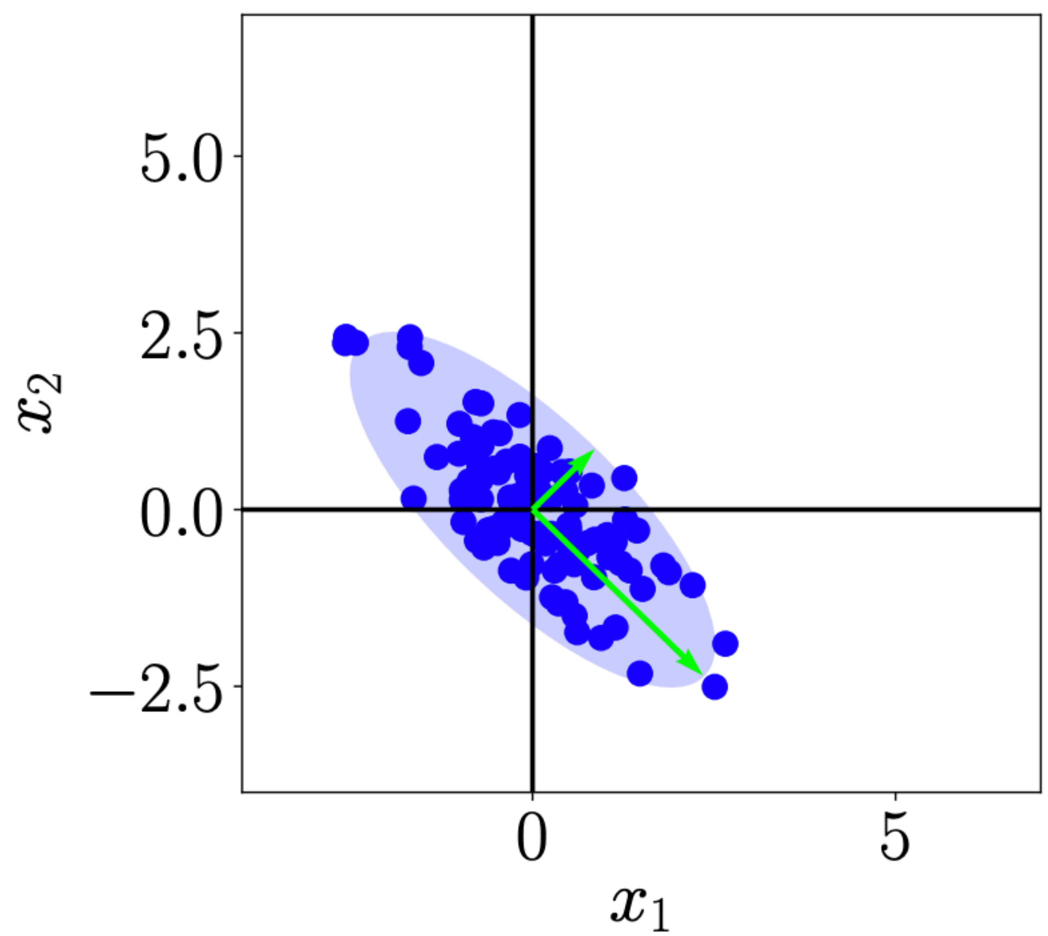
(b) Step 1: Centering by subtracting the mean from each data point.

PCA in Practice



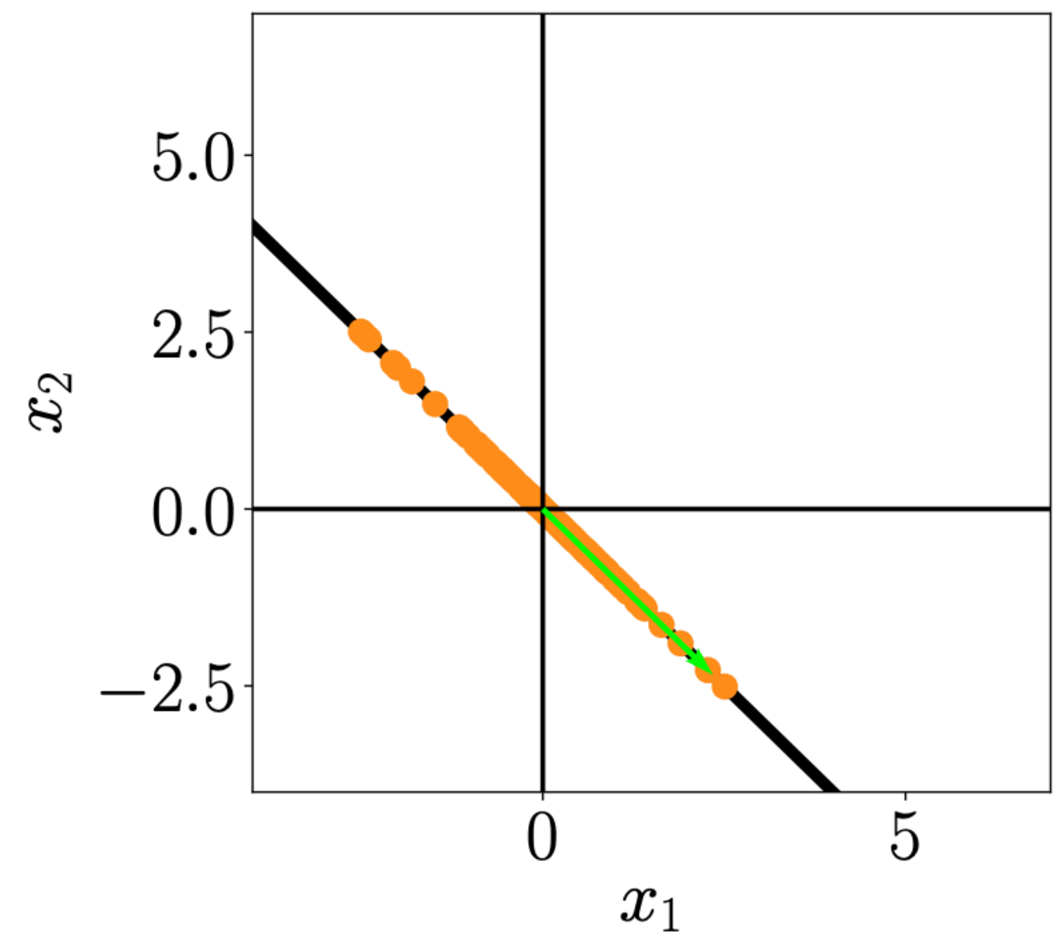
(c) Step 2: Dividing by the standard deviation to make the data unit free. Data has variance 1 along each axis.

PCA in Practice



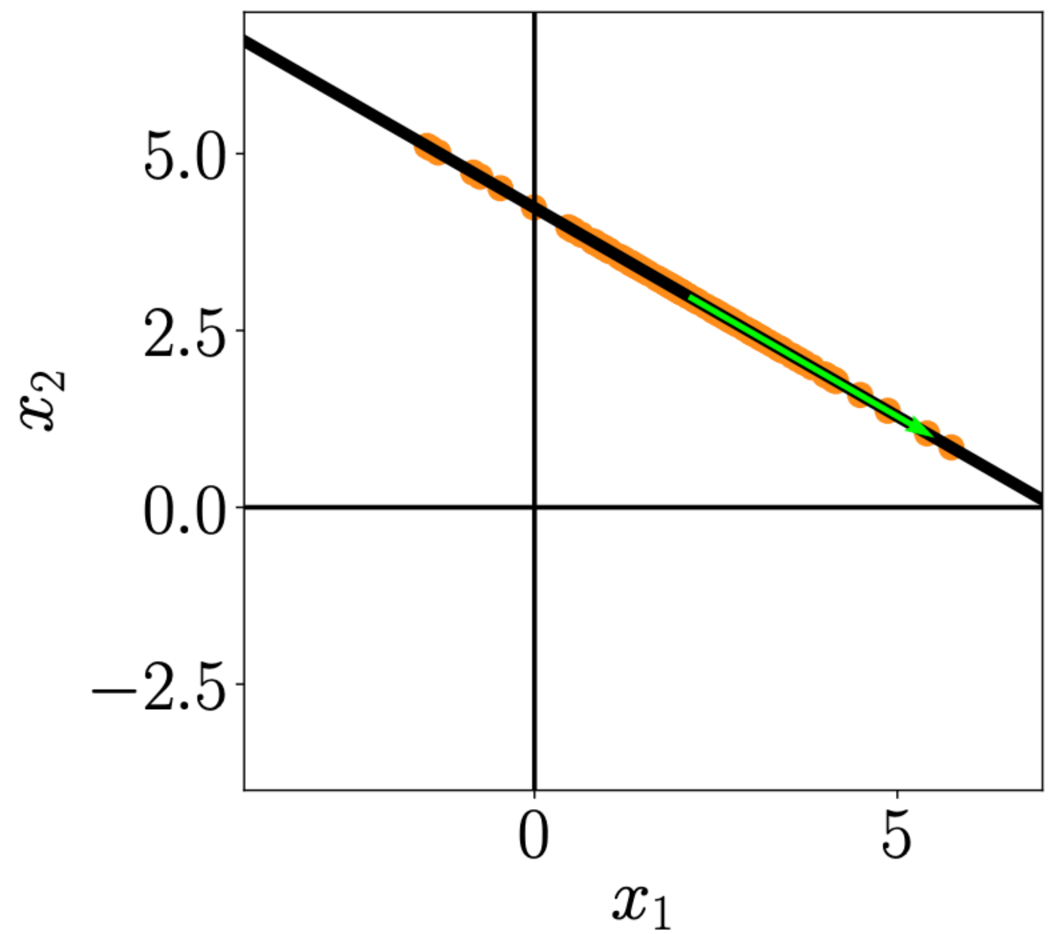
(d) Step 3: Compute eigenvalues and eigenvectors (arrows) of the data covariance matrix (ellipse).

PCA in Practice



(e) Step 4: Project data onto the principal subspace.

PCA in Practice



(f) Undo the standardization and move projected data back into the original data space from (a).

Successful Applications

- Novembre, John, et al.
  "Genes mirror geography within Europe."
  Nature 456.7218 (2008): 98-101.

- Turk, Matthew, and Alex Pentland.
  "Eigenfaces for recognition."
  Journal of cognitive neuroscience 3.1 (1991): 71-86.

Failure Cases

- Wrong scaling/normalization
- Non linear structure in your data
- Non orthogonal structure

Extra Materials

- https://web.stanford.edu/class/cs168/l/l7.pdf
- https://web.stanford.edu/class/cs168/l/l8.pdf
- https://www.youtube.com/watch?v=g-Hb26agBFg&t=1121s