Multivariate Analysis with PCA
CS1D6: Introduction to data and statistics

Dr. Fayyaz Minhas

Department of Computer Science

University of Warwick

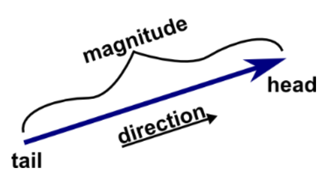
Contents

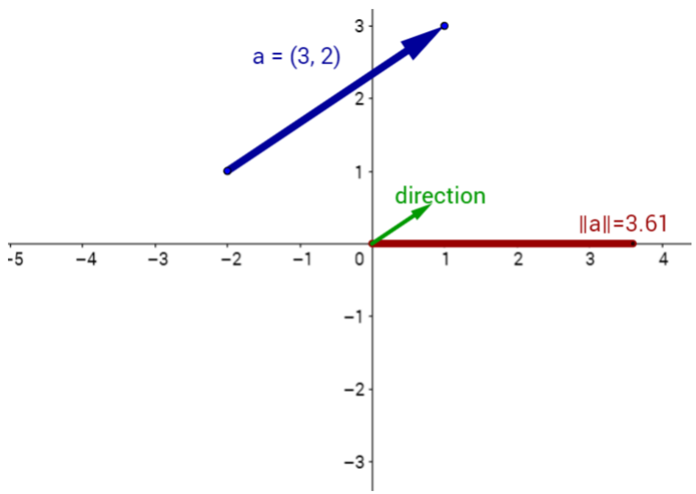
- What is the relationship between the following?
  - Distance
  - Norm
  - Dot product
  - Correlation
  - Covariance

- Basics
  - Vectors
  - Matrices
  - Dot Product and Projection
  - Eigen Vectors
  - Correlation
  - Covariance
- Covariance Matrix
- Principal Component Analysis (PCA)

Vectors

- We can measure single quantities
- But to represent multiple quantities associated with an object, we use vectors
- Example
  - We can represent an individual by their weight and height as a vector
  - Or a position on a map
  - A direction
- $\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$





Determining Similarity

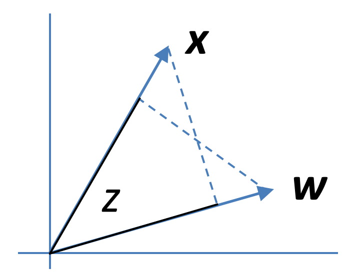
- Using distance
  - $d(\boldsymbol{u},\boldsymbol{v}) = \|\boldsymbol{u}-\boldsymbol{v}\| = \sqrt{(u_1-v_1)^2+(u_2-v_2)^2}$
    - Measures how far away one vector is “from” another
    - Norm/Magnitude
      - Length of a vector: $d(\boldsymbol{u},\boldsymbol{v}) = \|\boldsymbol{u}-\boldsymbol{v}\| = \|\boldsymbol{u}\| = \sqrt{{u_1}^2+{u_2}^2}$
- Using dot product
  - $\boldsymbol{u}\cdot\boldsymbol{v} = \boldsymbol{u}^T\boldsymbol{v} = \langle\boldsymbol{u},\boldsymbol{v}\rangle = u_1v_1+u_2v_2$
    - Measures how much one vector is “along” another
- Relationship between the two?

$$\|\boldsymbol{u}-\boldsymbol{v}\|^2 = (u_1-v_1)^2+(u_2-v_2)^2 = u_1^2+v_1^2-2u_1v_1+u_2^2+v_2^2-2u_2v_2 = u_1^2+u_2^2+v_1^2+v_2^2-2(u_1v_1+u_2v_2) = \|\boldsymbol{u}\|^2+\|\boldsymbol{v}\|^2-2\boldsymbol{u}^T\boldsymbol{v} = \boldsymbol{u}^T\boldsymbol{u}+\boldsymbol{v}^T\boldsymbol{v}-2\boldsymbol{u}^T\boldsymbol{v}$$

Dot Products and Projections

- One vector can be projected onto a vector by taking its dot-product

- $z = \boldsymbol{w}^T\boldsymbol{x}$
  - Projection of x in the direction of w



Some notes on representations

- **Preliminaries**
  - Love dot products (and learn to spot them!)

$$\boldsymbol{q} = \begin{bmatrix} b \\ d \\ f \end{bmatrix}$$

$$ab + cd + ef = \begin{bmatrix} a & c & e \end{bmatrix} \begin{bmatrix} b \\ d \\ f \end{bmatrix} = \boldsymbol{p}^T \boldsymbol{q} = \boldsymbol{q}^T \boldsymbol{p} = \boldsymbol{q} \cdot \boldsymbol{p}$$

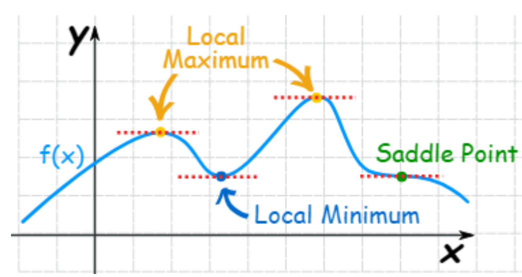
$$a^2 + c^2 + e^2 = \boldsymbol{p}^T \boldsymbol{p} = \|\boldsymbol{p}\|^2$$

$$\boldsymbol{p} = \begin{bmatrix} a \\ c \\ e \end{bmatrix}$$

  - Love matrix-vector products (and learn to spot them)

$$\begin{aligned} ab + cd + ef &= u \\ ag + ch + ek &= v \end{aligned} \qquad \begin{bmatrix} b & d & f \\ g & h & k \end{bmatrix} \begin{bmatrix} a \\ c \\ e \end{bmatrix} = \begin{bmatrix} u \\ v \end{bmatrix}$$

  - Love derivatives (and learn to solve them!)
    - Allow us to find minima or maxima



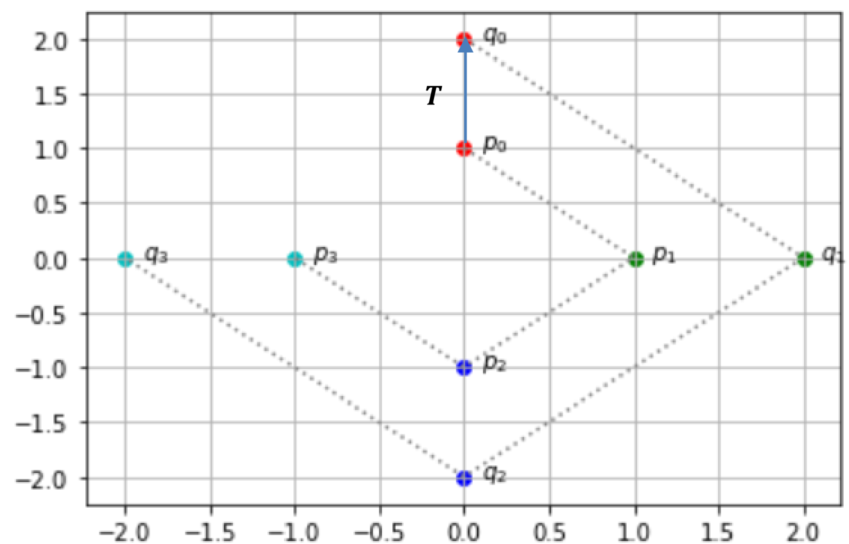
Operations on Vectors

- Using matrices
  - One way of thinking about matrices is that they are collection of vectors
  - For example, we can represent the data set for a given problem as a data matrix
    - Each row is a vector representation of a single example or data point
- Matrices as operators

Multiplication of a vector by a matrix

– Multiplication of a vector with a matrix can be viewed as a geometric transformation of the vector

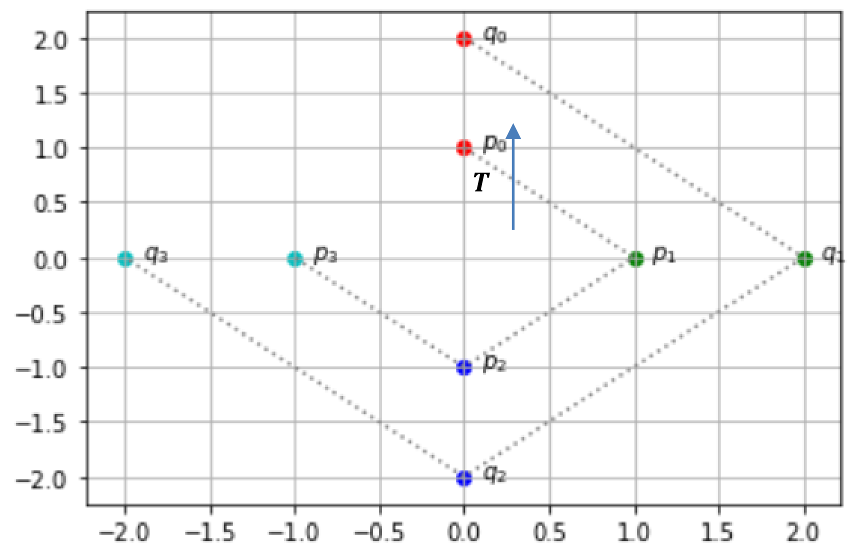
$$T = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
$$y = Tx = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$



Eigen Vectors

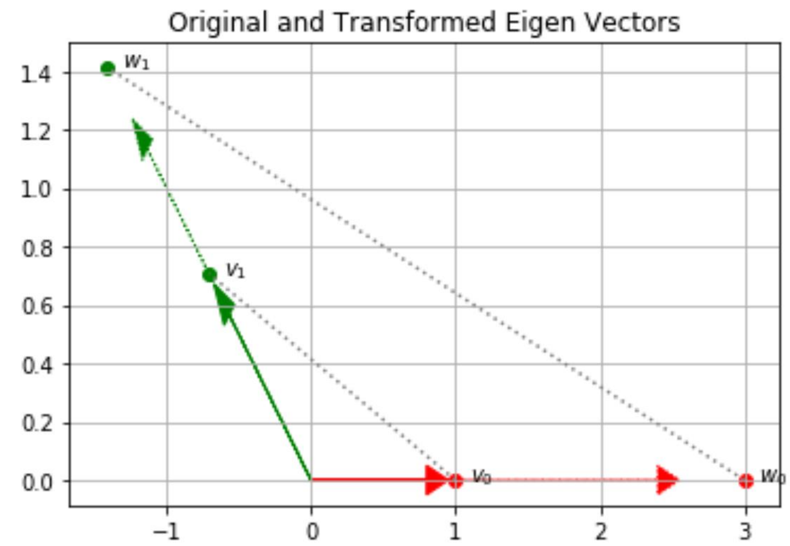
- Those points that are characteristic to a given matrix that undergo only a change in scale are called Eigen vectors $\boldsymbol{w} = \boldsymbol{T}\boldsymbol{v} = \lambda\boldsymbol{v}$
- How to find them: $(\boldsymbol{T} - \lambda\boldsymbol{I})\boldsymbol{v} = 0$ implies $|\boldsymbol{T} - \lambda\boldsymbol{I}| = 0$

$$\boldsymbol{T} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \boldsymbol{x} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
$$\boldsymbol{y} = \boldsymbol{T}\boldsymbol{x} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$



- See: https://github.com/foxtrotmike/PCA-Tutorial/blob/master/Eigen.ipynb

- $\boldsymbol{T} = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix}$



- Eigen Vector: $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$, Eigen Value: 3

$$\begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

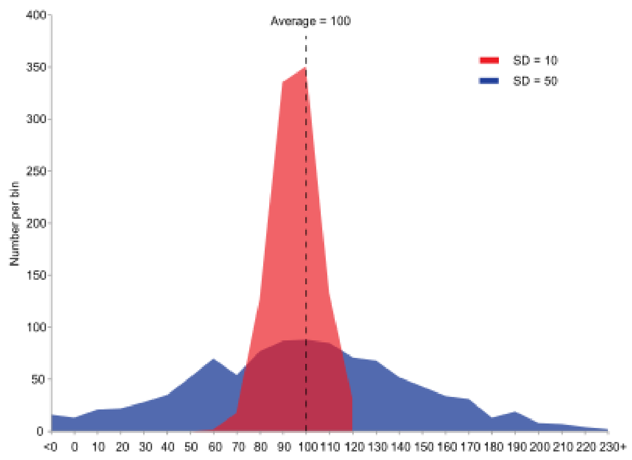
  – Note scaling only

- Eigen Vector: $\begin{bmatrix} -0.707 \\ 0.707 \end{bmatrix}$, Eigen Value: 2

$$\begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} -0.707 \\ 0.707 \end{bmatrix} = \begin{bmatrix} -1.414 \\ 1.414 \end{bmatrix} = 2 \begin{bmatrix} -0.707 \\ 0.707 \end{bmatrix}$$

Variance

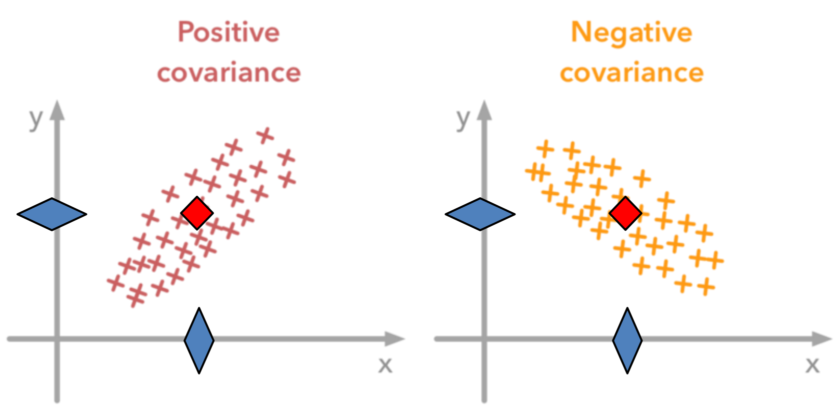
- Mean of the spread of a variable around its mean
- $var(z) = \frac{1}{N}\sum_{i=1}^{N}(z_i - \mu_z)^2 = \frac{1}{N}(\boldsymbol{z} - \mu_z)^T(\boldsymbol{z} - \mu_z)$
  - $\boldsymbol{z}$ is an N-dimensional vector composed of the values of all data points in the sample
- If mean is zero then $var(z) = \frac{1}{N}\boldsymbol{z}^T\boldsymbol{z} = \frac{1}{N}\|\boldsymbol{z}\|^2$
- $var(z) = E[(z - \mu_z)^2]$
- Variance as an information measure
  - How is variance related to information content?



https://www.khanacademy.org/math/ap-statistics/summarizing-quantitative-data-ap/more-standard-deviation/v/review-and-intuition-why-we-divide-by-n-1-for-the-unbiased-sample-variance

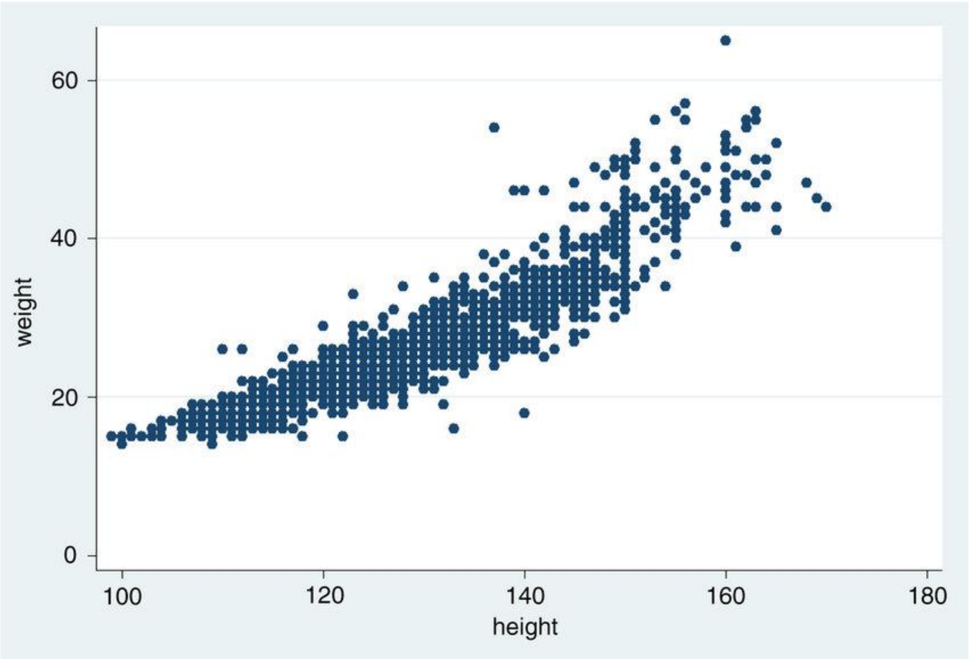
Covariance

- Co-Variance
  - Given two random variables, to what extent are they linearly related to each other
  - $cov(x, y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) = \frac{1}{N}(\boldsymbol{x} - \mu_x)^T(\boldsymbol{y} - \mu_y)$
    - Covariance is positive if, on average,
      - When one variable is above its mean then the other variable is above its mean too
      - When one variable is below its mean then the other variable is below its mean too
    - Covariance is negative if, on average,
      - When one variable is above the mean, the other is below its mean
  - Assume that the means are zero: $cov(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{N}\boldsymbol{x}^T\boldsymbol{y}$
    - Maximum when the vectors are co-linear or parallel
  - $cov(\boldsymbol{x}, \boldsymbol{y}) = E\left[(y - \mu_y)(x - \mu_x)\right]$
  - Thus, $var(z) = cov(z, z)$



Correlation

- What is the association between two random variables?
  - Example: How are height and weight associated with each other?



Quantifying Correlation

- We can quantify the degree of linear association between two random variables through correlation coefficient

covariance $\mathrm{cov}_{XY} = \sigma_{XY} = E[(X - \mu_X)(Y - \mu_Y)]$

correlation $\mathrm{corr}_{XY} = \rho_{XY} = E[(X - \mu_X)(Y - \mu_Y)]/(\sigma_X \sigma_Y)$

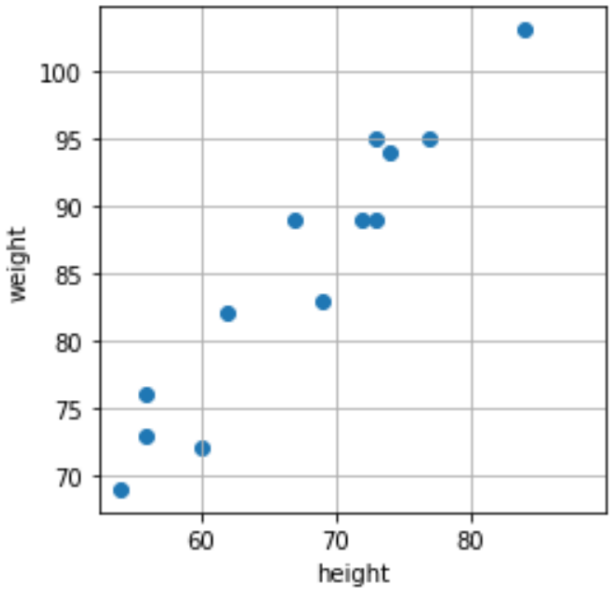
  – Pearson correlation

https://en.wikipedia.org/wiki/Covariance_and_correlation

Covariance Matrix of a dataset

- Matrix of all pairwise covariances of all variables
  - $\boldsymbol{C} = \begin{bmatrix} cov(y,y) & cov(z,y) \\ cov(y,z) & cov(z,z) \end{bmatrix}$

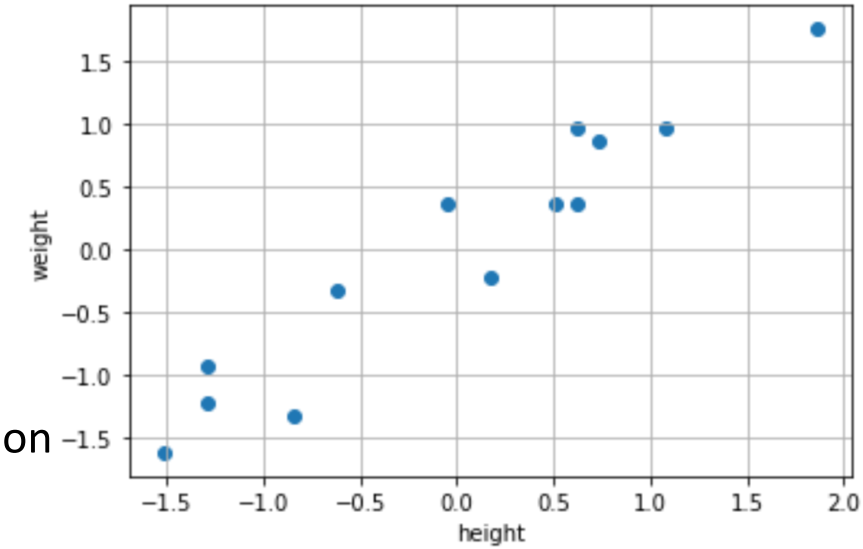
Covariance Matrix Example



$$\frac{h - \mu_h}{\sigma_h}$$

$$\frac{w - \mu_w}{\sigma_w}$$

Standardization



The mean is [67.46 85.31]
The standard deviation is: [8.86 10.06]
The variance is: [78.56 101.14]
The co-variance matrix is: $\begin{bmatrix} 78.56 & 85.55 \\ 85.55 & 101.14 \end{bmatrix}$

The mean is [0 0]
The standard deviation is: [1 1]
The variance is: [1 1]
Total variance: 1+1 = 2.0
The co-variance matrix is: $\begin{bmatrix} 1 & 0.96 \\ 0.96 & 1 \end{bmatrix}$

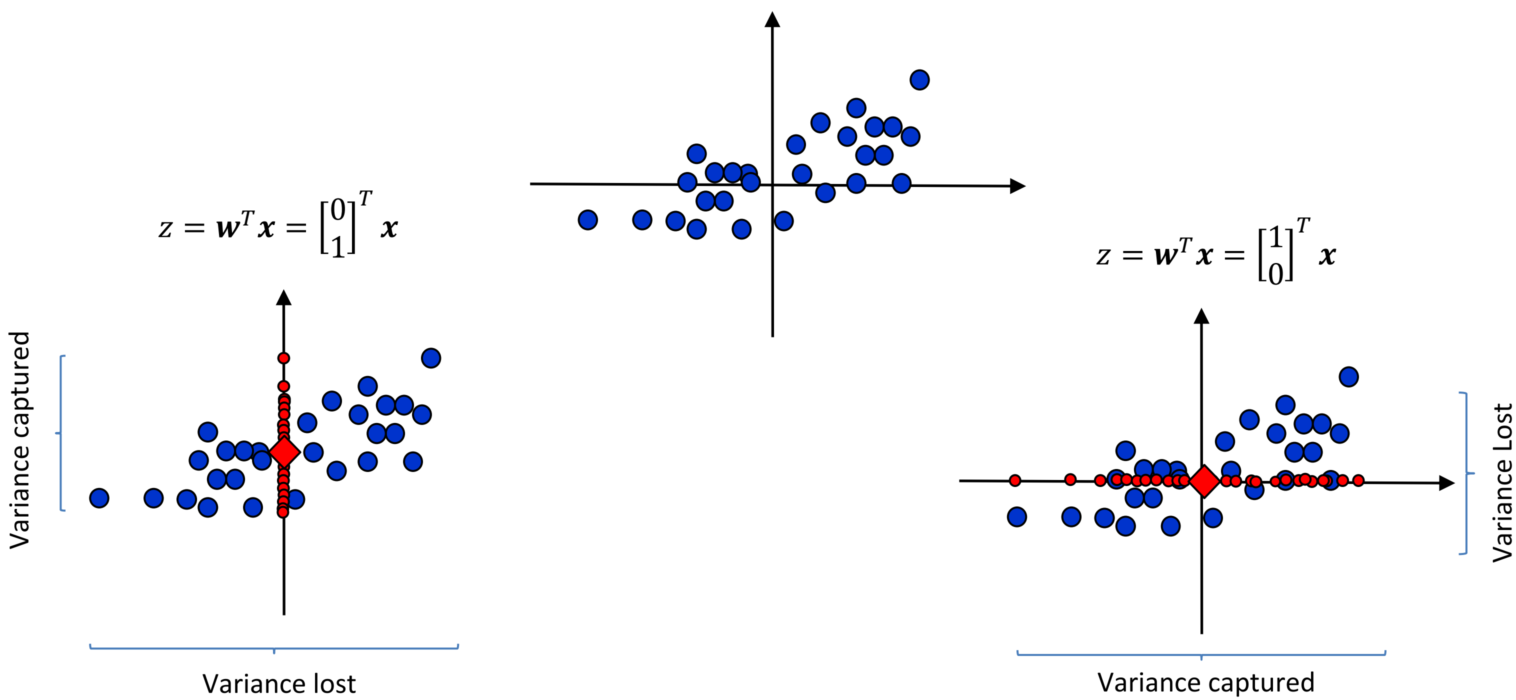
Other Correlation Coefficient

- Spearman Rank Correlation
  - Perform a rank transform and then calculate the correlation based on the ranks
    - Ignores the raw values
- Kendall Correlation

https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient
https://en.wikipedia.org/wiki/Kendall_rank_correlation_coefficient

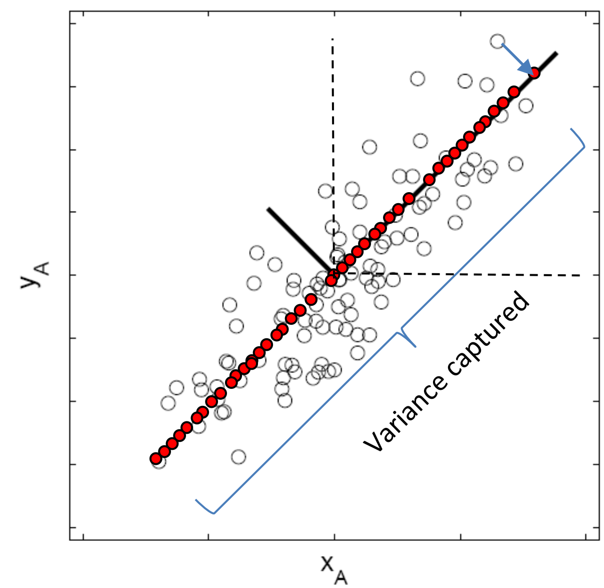
Data Dimensionality Reduction

- How can we reduce dimensions?
  - Drop features?
    - Equivalent to projecting data onto canonical axes
    - Loss in variance



Dimensionality Reduction as Projections

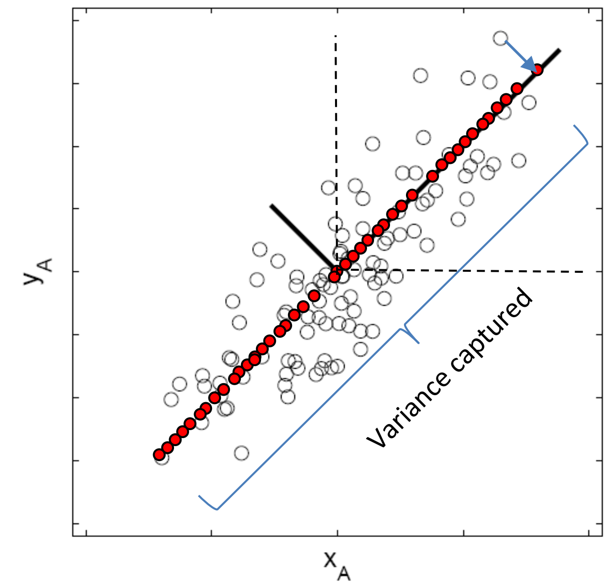
- Projections can be used for reducing dimensions
  - However, projecting data onto a vector loses information
  - We want to reduce the amount of information loss
  - Solution: Find and project along a direction along which information loss is minimum
    - A direction along which most of the variance is captured
  - How to do it?

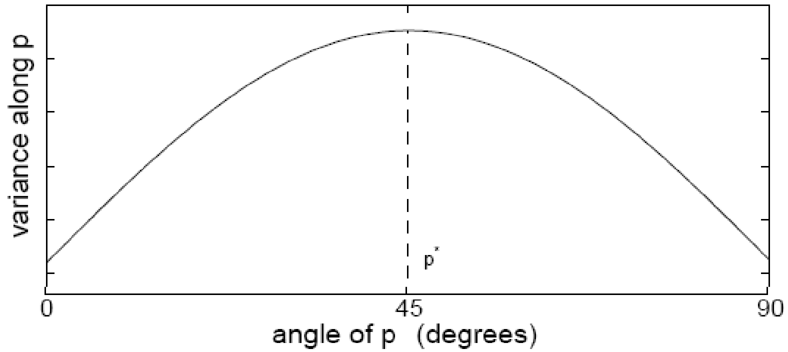


How to do it: Naïve Implementation

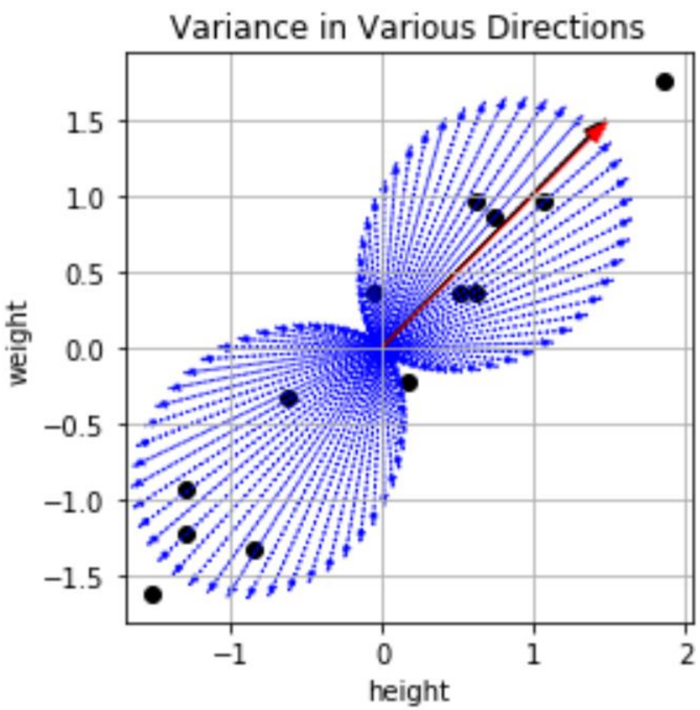
- Set p = 0
- For p from 0 to π in steps
  - Calculate projection vector
    - $\boldsymbol{w_p} = \begin{bmatrix} \cos(p) \\ \sin(p) \end{bmatrix}$
  - Project your data onto $z_i = \boldsymbol{w}_p^T \boldsymbol{x}_i$
  - Find the variance of the projected data
- Plot the variance across p
- Find the p that gives maximum variance

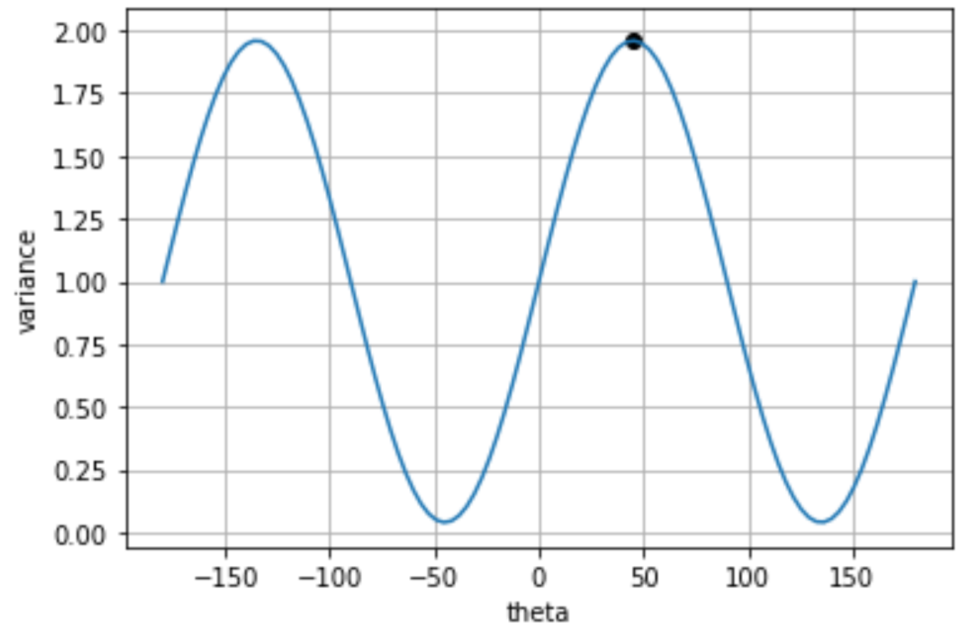
- Issues?





Using the naïve implementation

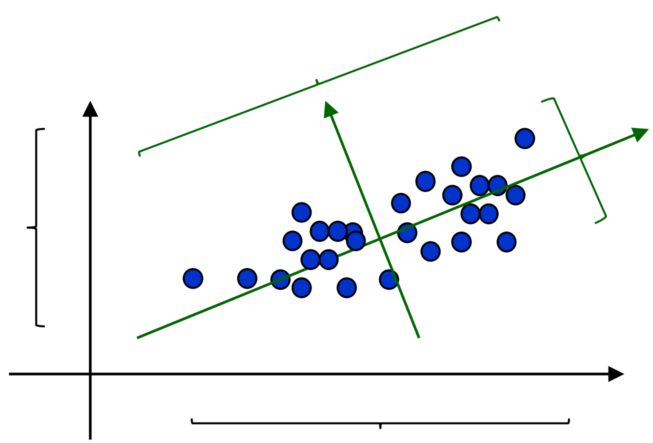





```
Direction of Maximum Variance: [0.70, 0.71]
```

So what is PCA?

- A method for transforming the data
  - Projecting the data onto orthogonal vectors such that the variance of the projected data is maximum
  - Projection of x on the direction of w: $z = w^T x$
  - Find w such that Var(z) is maximized



Principal Component Analysis

- Relation between variance of projection and covariance matrix

$$\begin{aligned}\mathrm{Var}(z) = \mathrm{Var}(\boldsymbol{w}^T\boldsymbol{x}) &= \mathrm{E}[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})^2] \\ &= \mathrm{E}[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})] \\ &= \mathrm{E}[\boldsymbol{w}^T(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{w}] \\ &= \boldsymbol{w}^T\,\mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T]\boldsymbol{w} = \boldsymbol{w}^T\,\mathbf{C}\,\boldsymbol{w}\end{aligned}$$

where $\mathrm{Cov}(x) = \mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T] = \mathbf{C}$

- If we know *w*, we can calculate the variance of the projected data along that direction

Principal Component Analysis

- We want to find a unit vector **w** that maximizes the variance along the projection
- Maximize Var(z_1)
- subject to $\mathbf{w}_1^T \mathbf{w}_1 = 1$
- Using the method to

$$\max_{\mathbf{w}_1} \mathbf{w}_1^T \boldsymbol{C} \mathbf{w}_1 - \alpha(\mathbf{w}_1^T \mathbf{w}_1 - 1)$$

- Taking the derivative of this with respect to w and substituting to zero, we get

$$\boldsymbol{C}\mathbf{w}_1 = \alpha \mathbf{w}_1$$

Method of Lagrange Multipliers

Constrained Optimization Problem

$$\max_{\mathrm{u}} f(u) \; s.t. \; g(u) = 0$$

$$\Updownarrow$$

$$\max_{\mathrm{u},\alpha} f(u) - \alpha g(u)$$

Unconstrained Optimization Problem

https://en.wikipedia.org/wiki/Lagrange_multiplier

Principal Component Analysis

- The direction of maximum variance is $\mathbf{w}_1$, given by:

$$\boldsymbol{C}\mathbf{w}_1 = \alpha\mathbf{w}_1$$

- $\mathbf{w}_1$ is the Eigen Vector Corresponding to the covariance matrix $\boldsymbol{C}$ with Eigen value α

An algorithmic view of how PCA Works

- **Input**: $\boldsymbol{X}_{N\times d}$
- **Output**: A transformation matrix **W** which can be used for dimensionality reduction
- **Parameters**: Selection of principal components
  - Proportion of variance
  - Number of principal components (k)
  - Which principal components to retain

- **Internal Working**
  - Normalize data
    - Calculate feature wise mean and standard deviation and normalize data to zero mean and unit standard deviation
  - Find Covariance Matrix
  - Find Principal Components (Eigen Value Problem)
  - Select Principal Components
    - Using Scree Graph
    - Intuition
  - Reduce dimensionality by Projection along selected components

Let's see for our data

The co-variance matrix is: $\begin{bmatrix} 1 & 0.96 \\ 0.96 & 1 \end{bmatrix}$

Eigen vector 1:

$$w_1 = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}, \ \alpha_1 = 1.96$$

Variance of data after projecting along w_1: 1.96

Eigen vector 2:

$$w_2 = \begin{bmatrix} -0.7071 \\ 0.7071 \end{bmatrix}, \ \alpha_2 = 0.04$$

Variance of data after projecting along w_1: 0.04

Fraction of variance captured along each PC:

Using PC-1: 1.96/2 = 0.98

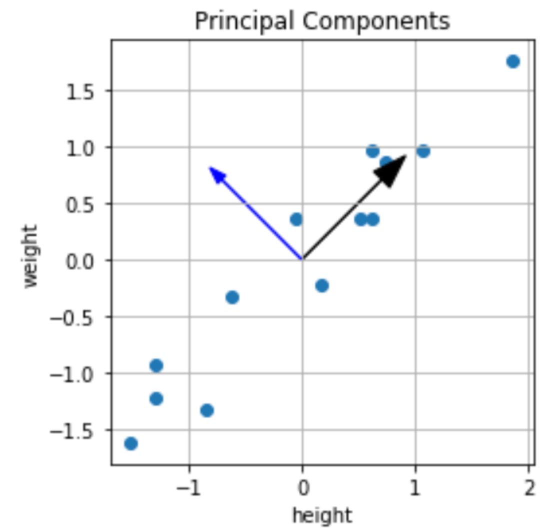
Using PC-1 and PC-2: (1.96+0.04)/2 = 1.0

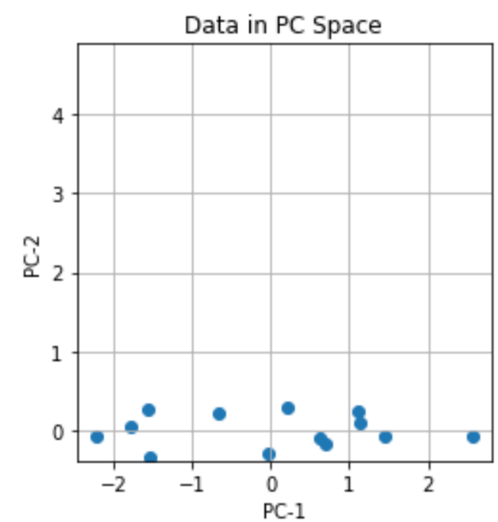
The two PC vectors are orthogonal to each other $w_1^T w_2 = 0$

The PC Matrix is $\mathrm{W} = \begin{bmatrix} 0.7071 & -0.7071 \\ 0.7071 & 0.7071 \end{bmatrix}$

The inverse of W is: $\mathrm{W}^{-1} = \begin{bmatrix} 0.7071 & 0.7071 \\ -0.7071 & 0.7071 \end{bmatrix} = W^T$

Thus, $\mathrm{W^T W = I}$



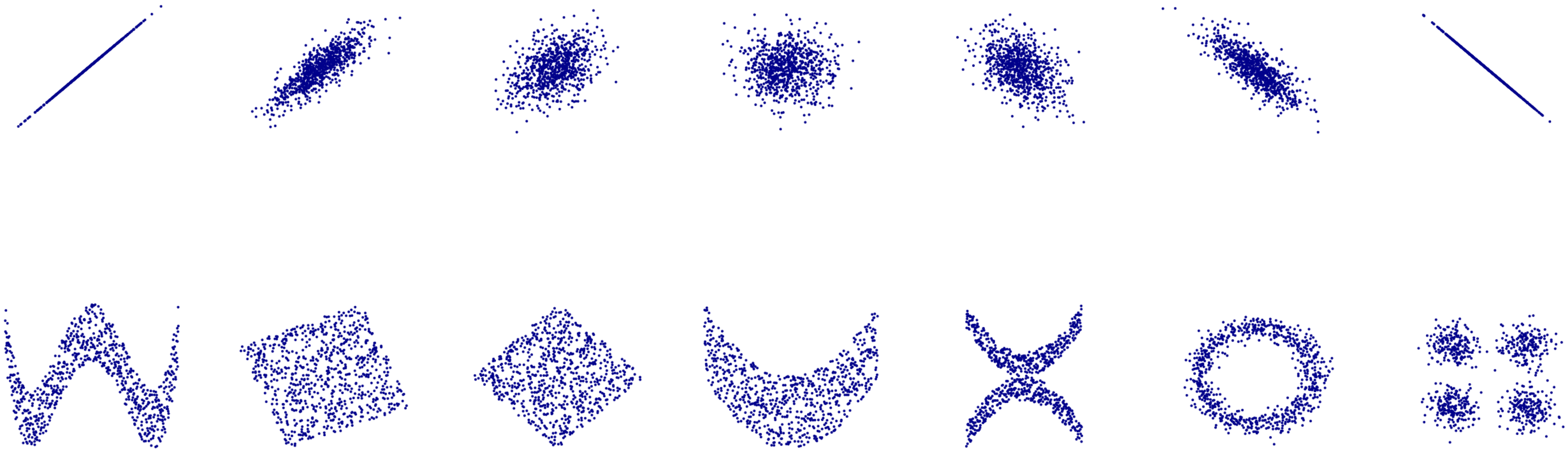


Things to note

- There are two principal components: The one with the largest variance (eigen value) is called the first principal component whereas the other one is called the second principal component.
- The variance along the first principal component is higher in comparison to the second.
- The variance along the first projected direction is higher than the variance along original features which is 1.0 after normalization. Thus, the principal component is a direction that captures more information than any of the original features alone.
- The norm of each of the principal components is 1.0.
- The two principal components are orthogonal to each other.
- The principle component matrix and its transpose are inverses of each other, i.e., $\mathbf{W}^{\mathbf{T}}\mathbf{W} = \mathbf{I}$
- The eigen values correspond to the amount of captured variance: The fraction of variance captured along a direction is exactly equal to the fraction of eigen values. Thus, the first principal component corresponds to the largest eigen value and so on.
- The plot of the fraction of captured variance up to k principal components (called the scree plot) can be used to select how many principal components to retain when reducing dimensionality. For the original data used in this example, upto 98% variance is along the first principal component. Therefore, if the second principal component is dropped, the loss of information will be only ~2%.

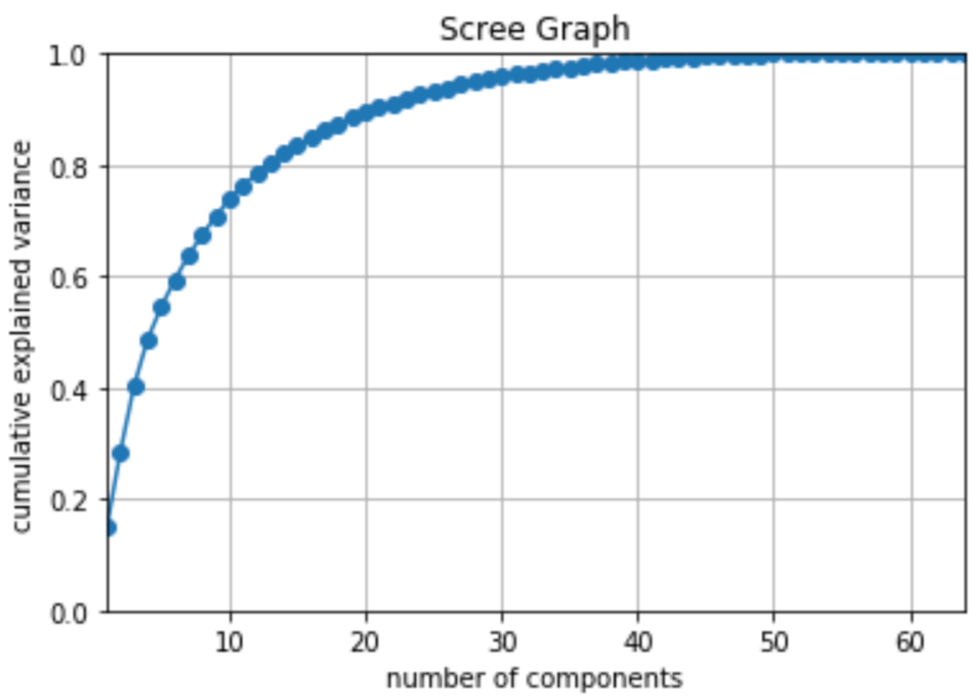
Quiz Time: Find Principal Components

- What are the principal components for each data set below?



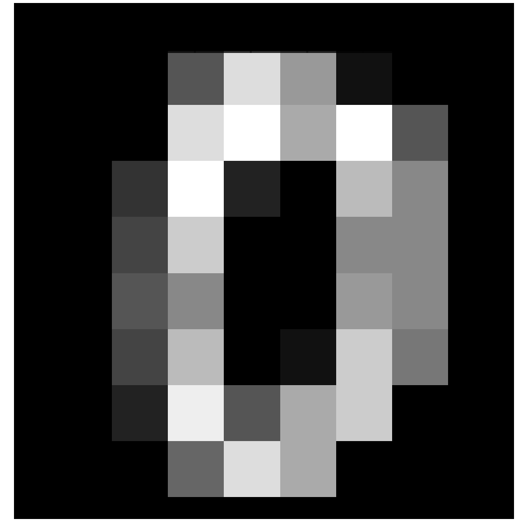
How many principal components?

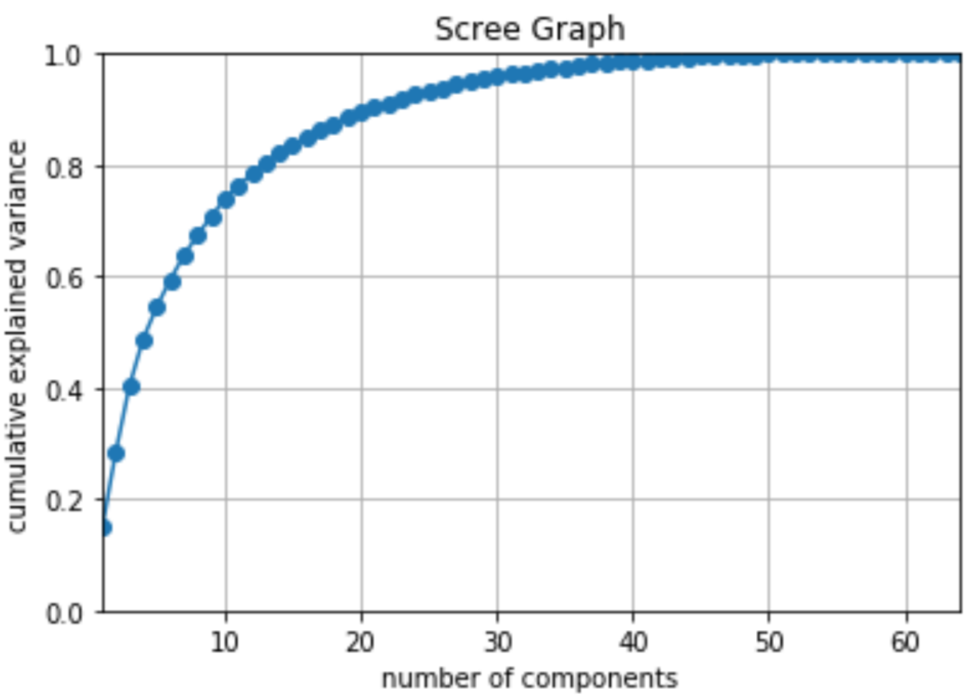
- Scree Graph
  - Plot the proportion of variance that is captured by incorporating more and more principal components

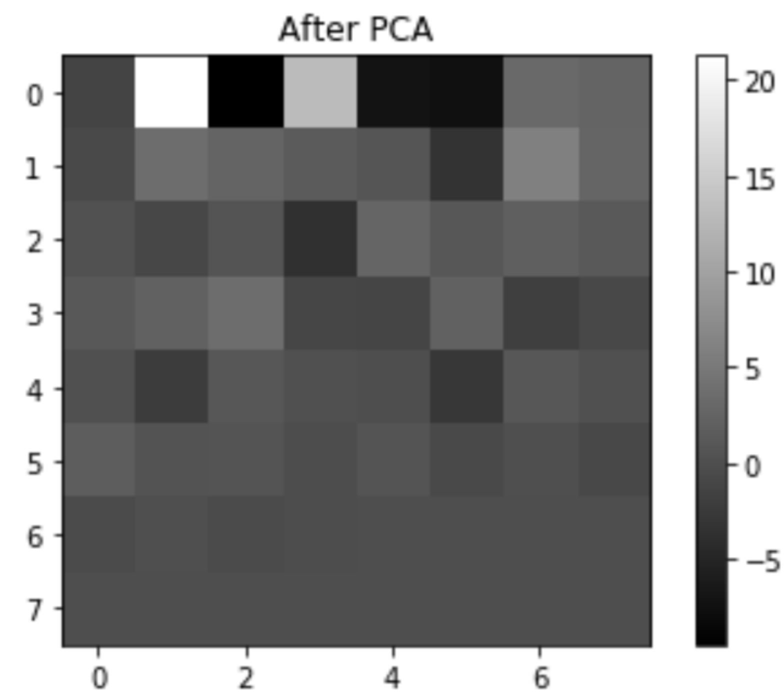


Example

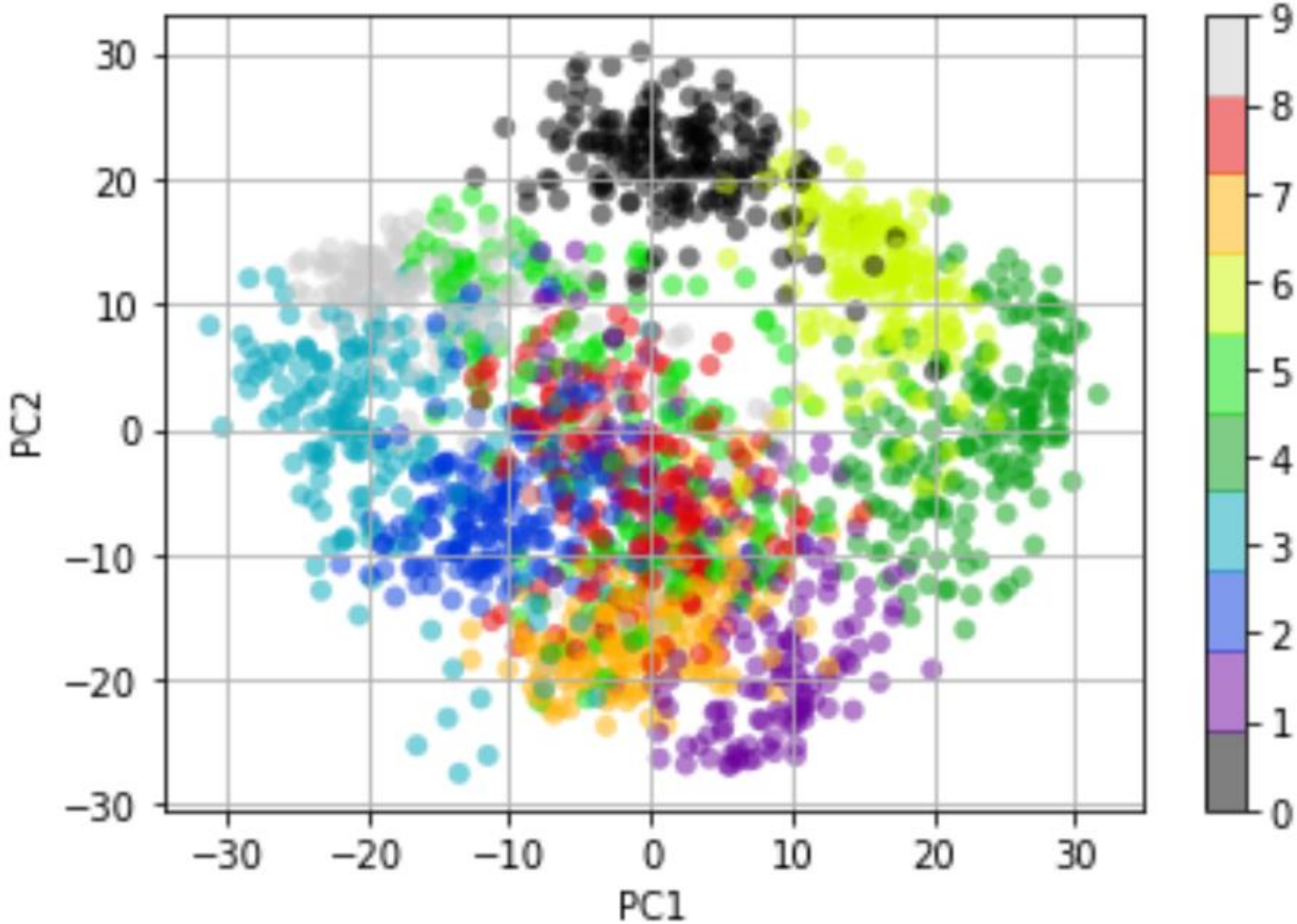
- MNIST visualization
- X: 1797x64







Visualization

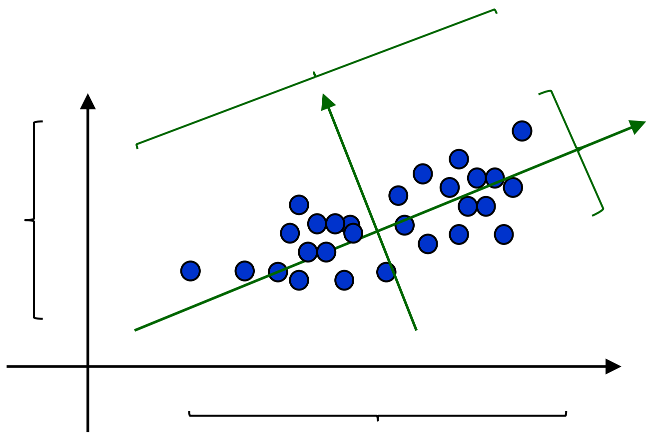


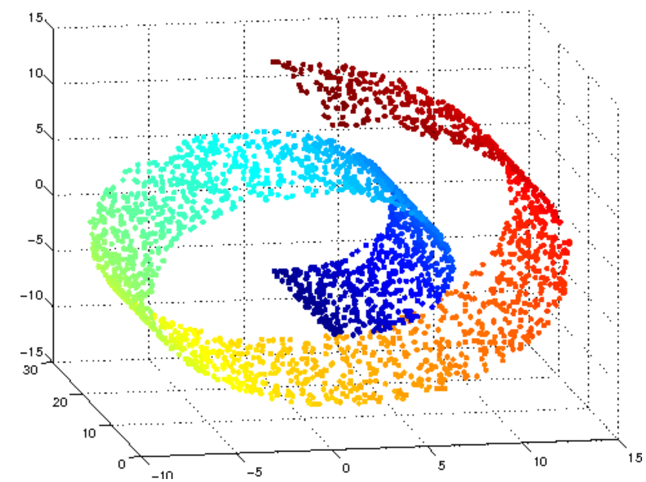
How to code?

- **Fitting PCA to training data**
  - `from sklearn.decomposition import PCA`
  - `pca = PCA(n_components=4)`
  - `pca.fit(X) #rows are samples, columns are features`
- **Projection**
  - `Z = pca.transform(X)`
- **Visualization**
- **Screen Graph**
  - `plt.plot(np.cumsum(pca.explained_variance_ratio_),'o-’)`
- **Reconstruction**
  - `Xr = pca.inverse_transform(Z)`

Important Conceptual Note

- A number of variables can be correlated in real datasets
- Thus, the effective dimensionality of the dataset can be lower than what you see in terms of number of features
- Thus, data lives on a “manifold”
- That is why dimensionality reduction models help
- Sometimes these manifolds may not be linear





Other ways of dimensionality reduction

- Multi-dimensional scaling
  - Reduce the number of dimensions in the data while preserving pairwise distances between points
- t-distributed Stochastic Neighbor Embedding (t-SNE)
  - Reduce data dimensionality while preserving the local probability distribution of the data
    - Used for visualization
- UMAP

https://scikit-learn.org/stable/modules/classes.html#module-sklearn.manifold

Notes and Exercise

- https://github.com/foxtrotmike/PCA-Tutorial/blob/master/Eigen.ipynb
- https://github.com/foxtrotmike/PCA-Tutorial/blob/master/pca-lagrange.ipynb

End of Lecture

We want to make a machine that will be proud of us.

- Danny Hillis