Video signals

FEATURES EXTRACTION

Hough Transform

Goal: recognize lines in images

Approach:

- For every point in the starting image plot the sinusoid on the dual plane (parameter space):

  $$\rho = x * \cos(\vartheta) + y * \sin(\vartheta)$$

  where x and y are fixed (the considered point coordinates) while ρ and ϑ are variables.
- The Hough Transform of an image with K lines is the sum of many sinusoids intersecting in K points.
- Maxima in the dual plane indicate the parameters of the k lines

Hough: implementation

Consider a discretization of the dual plane for the parameters (ρ,ϑ): it becomes a matrix whose raw and column indices correspond to the quantized values of ρ and ϑ.

The limits of ρ are chosen accordingly to the image size.

Usually: $-\rho_{max} \leq \rho \leq \rho_{max}$, $-\pi/2 \leq \vartheta \leq \pi/2$

Hough: implementazion

Clear the matrix H(m,n);

Fro every point P(x,y) of the image

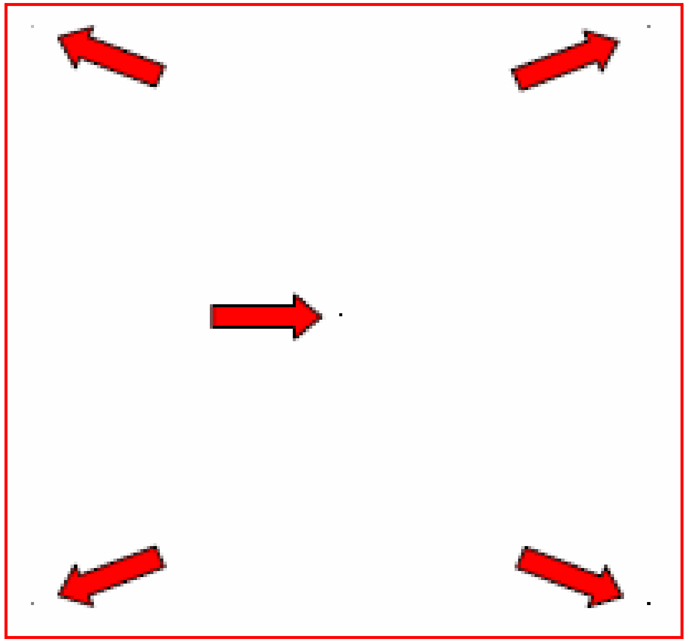
- 1. for ϑ_n that ranges from $-\pi/2$ to $\pi/2$ with step $d\vartheta$
  - 1. Evaluate $\rho(n)=x*\cos(\vartheta_n)+y*\sin(\vartheta_n)$
  - 2. find the index *m* corresponding to *ρ(n)*
  - 3. Increase H(m,n)
- 2. end

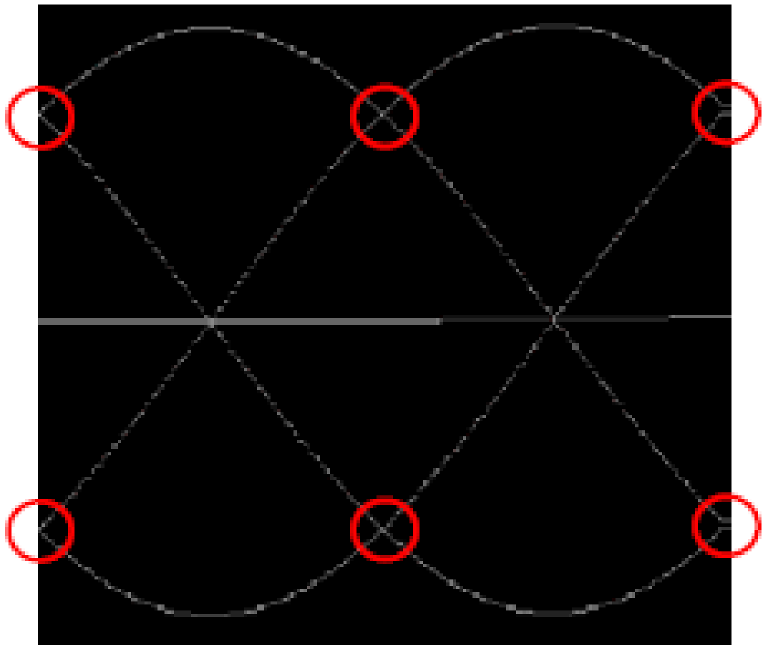
end

4. Find local maxima in H(.,.) that will corresponds to parameters of the founded lines

Hough Trasform

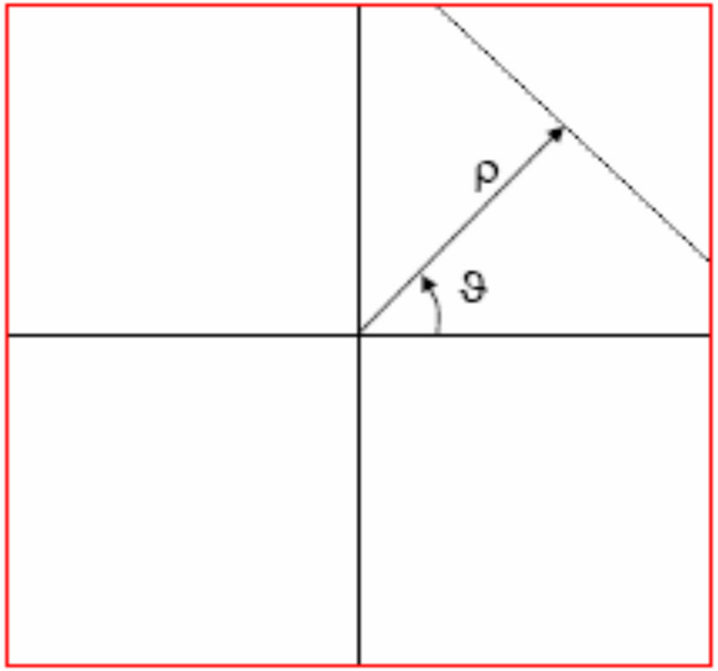
5 points

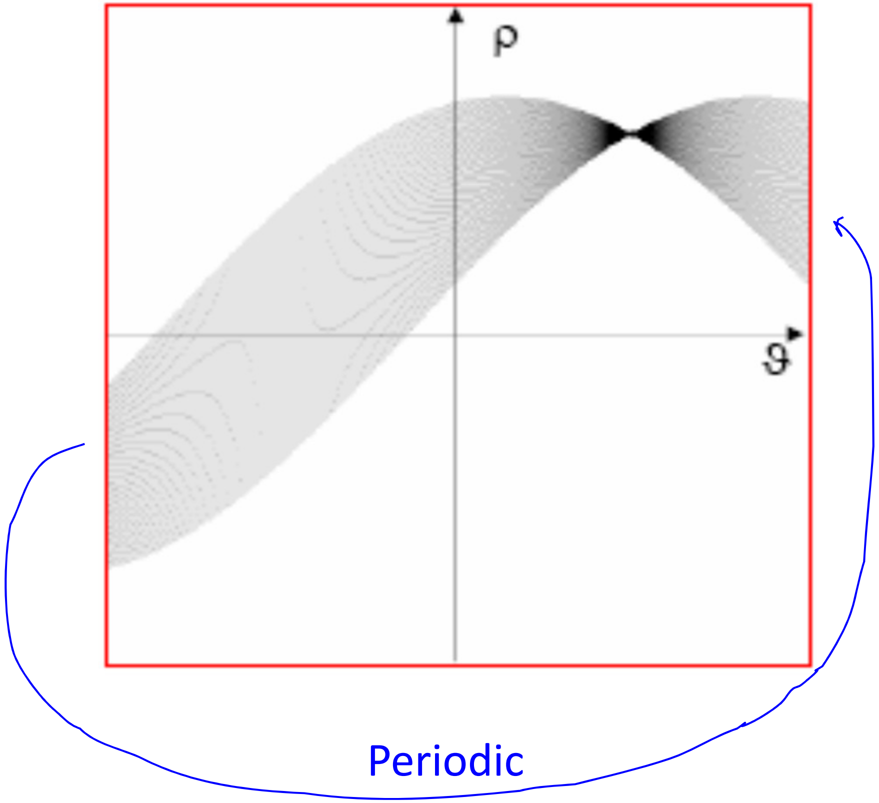




Hough Trasform

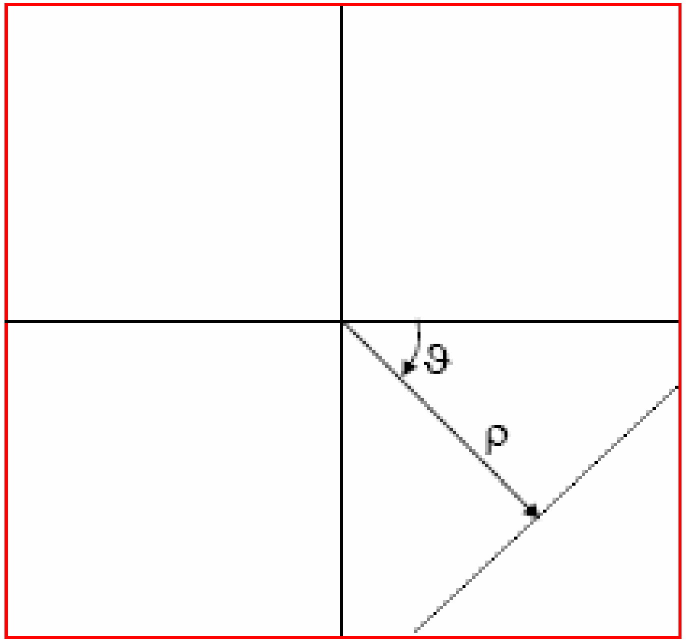
line $\rho>0, \theta>0$

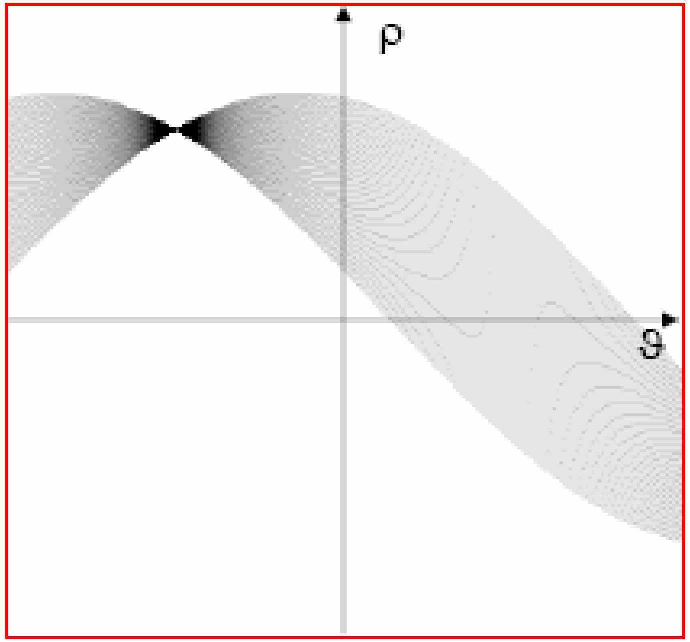




Hough Trasform

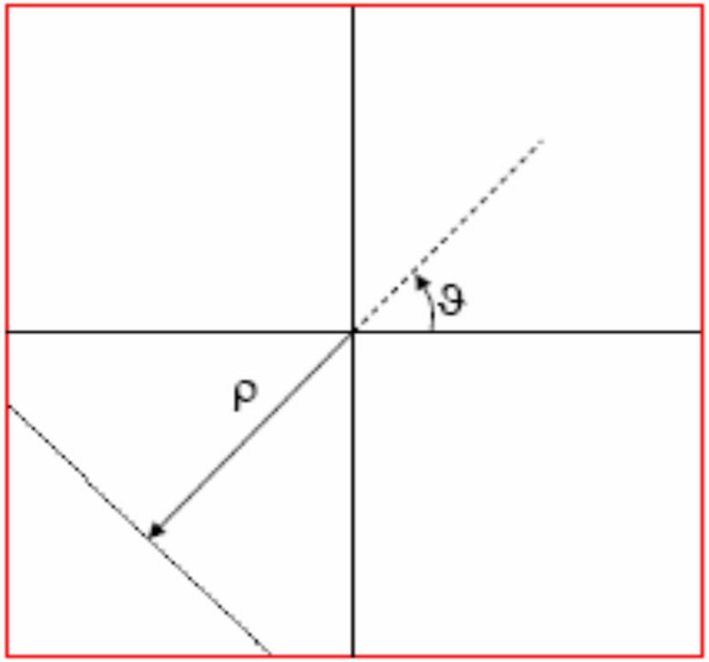
line $\rho>0, \theta<0$

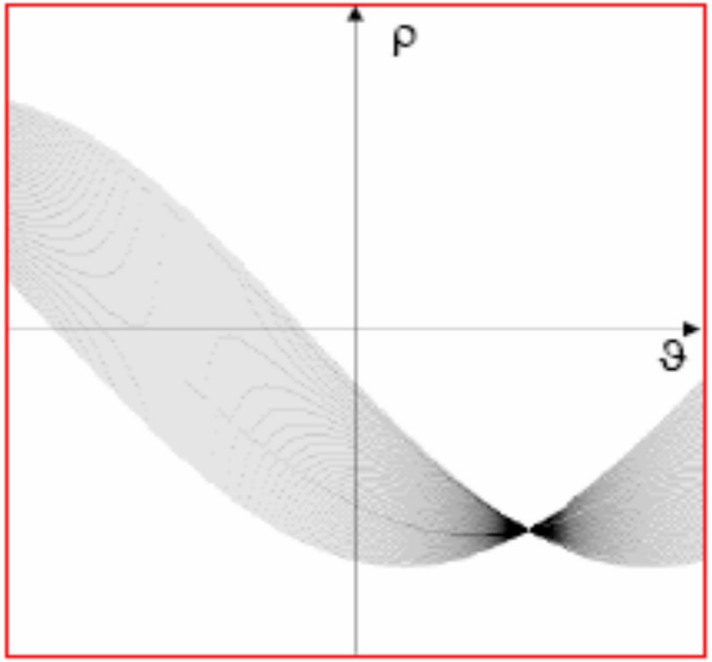




Hough Trasform

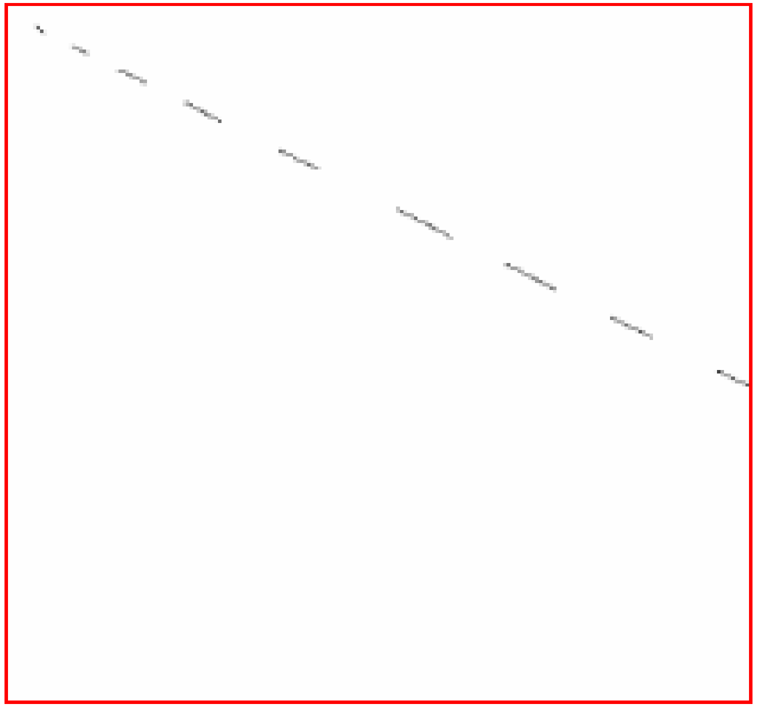
line $\rho<0,\theta>0$

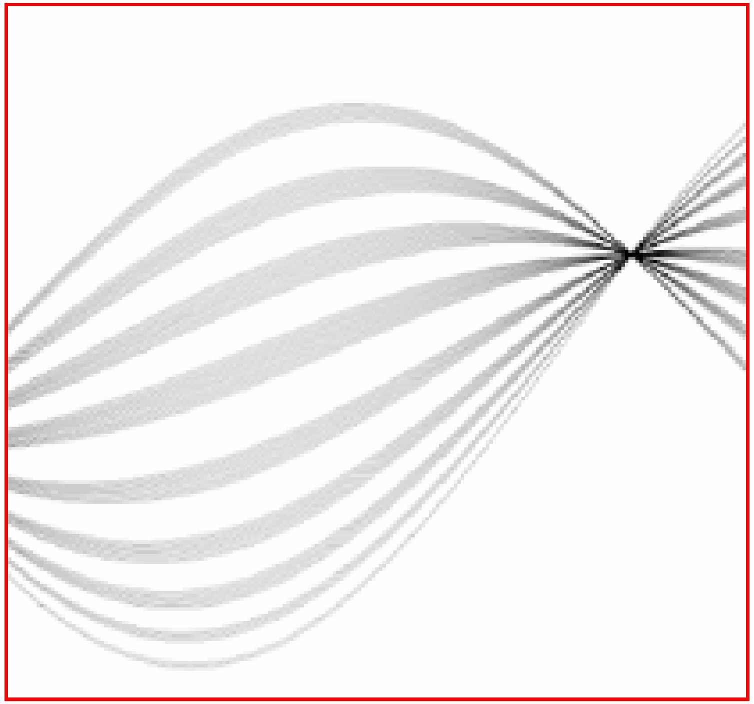




Hough Trasform

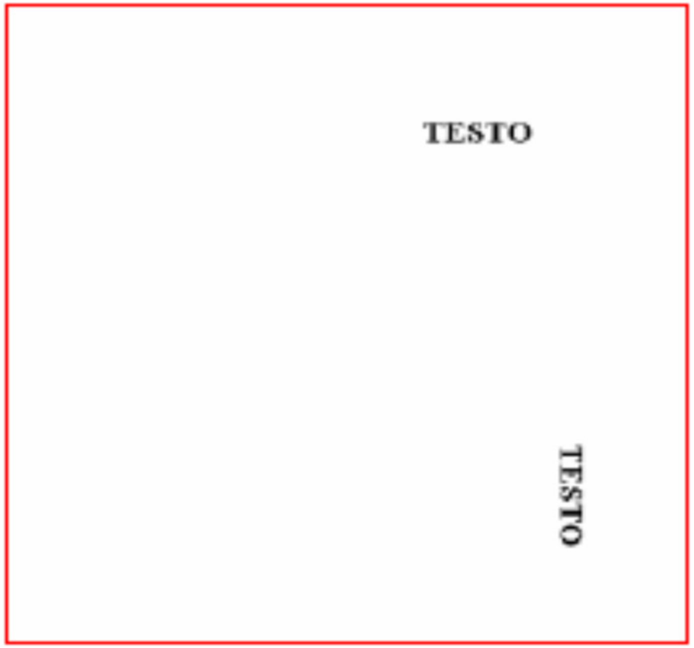
Dotted line

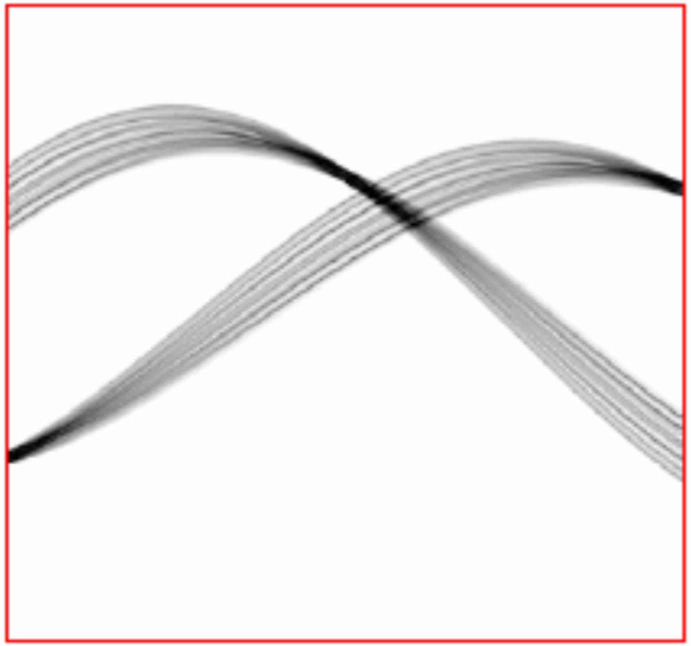




Hough Trasform

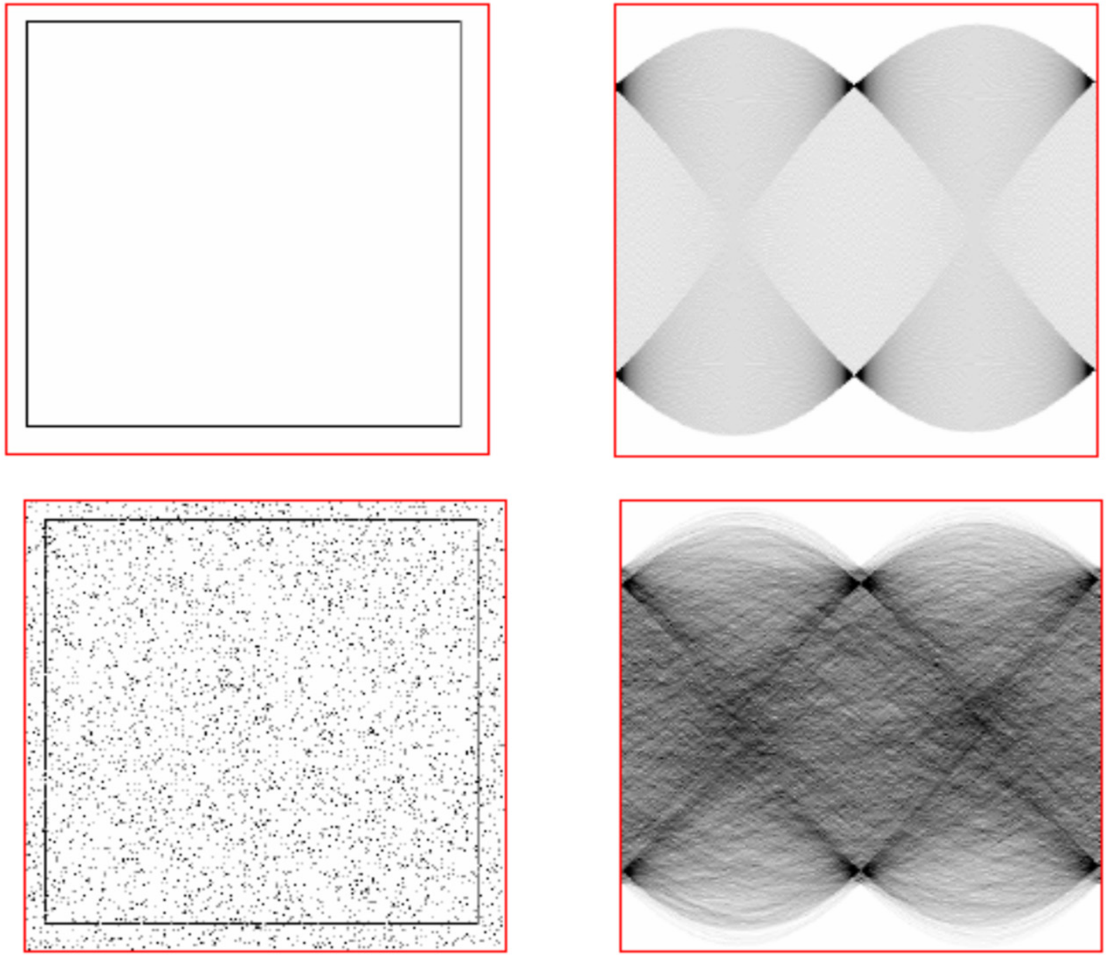
Same text with different orientations





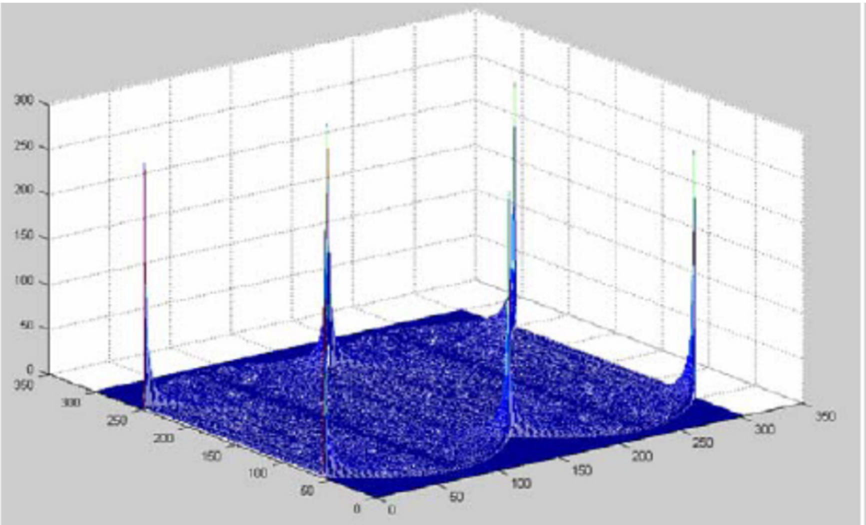
Hough Trasform

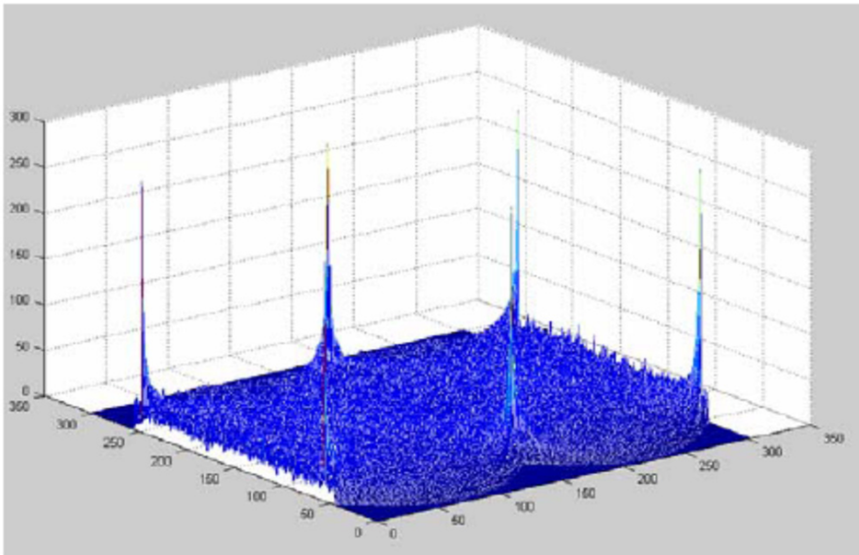
Noise and noiseless square



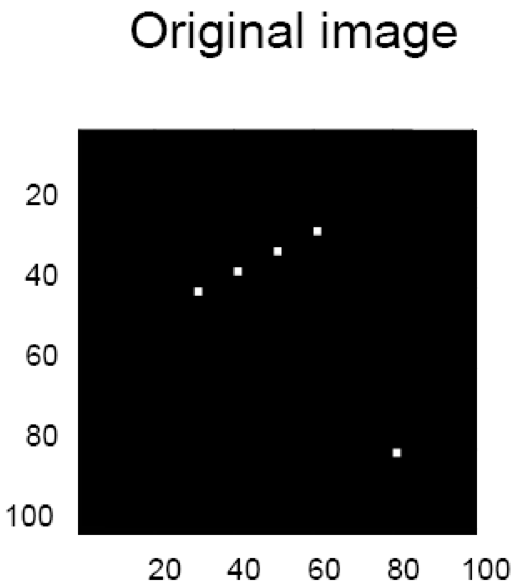
Hough Trasform

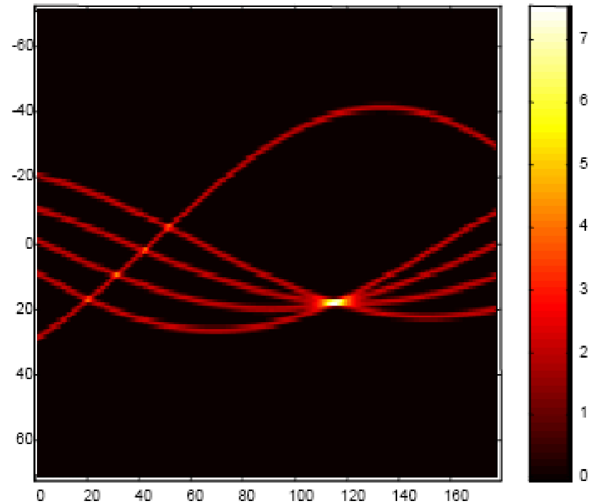
Accumulation matrices of the previous images

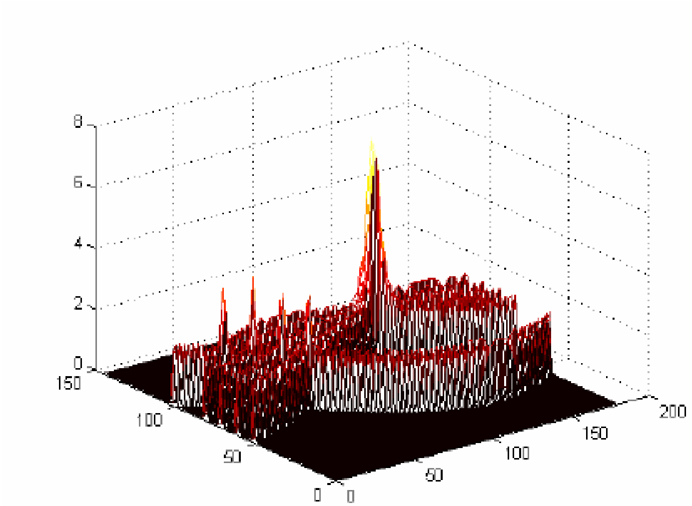




Examples



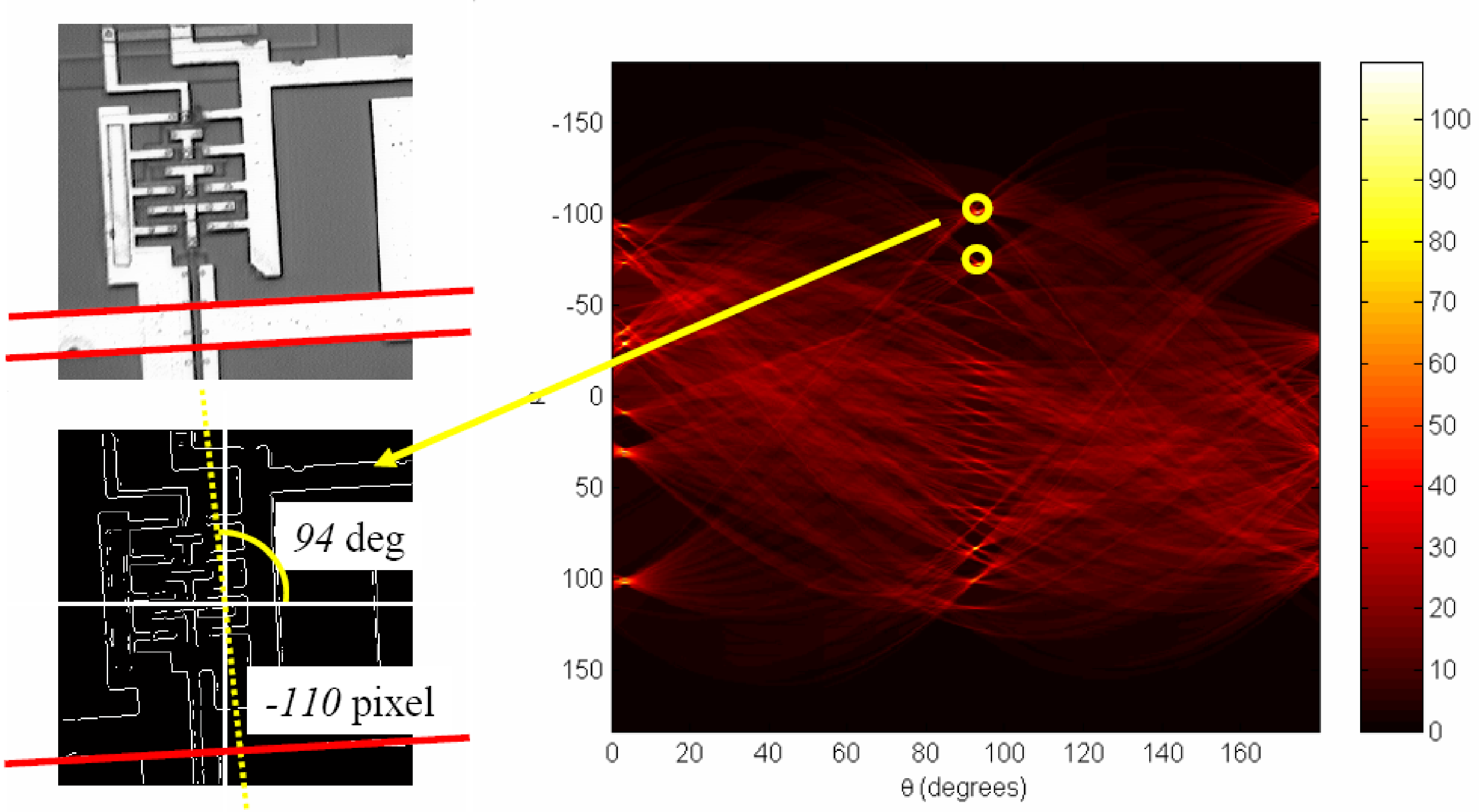




Courtesy: P. Salembier

Example

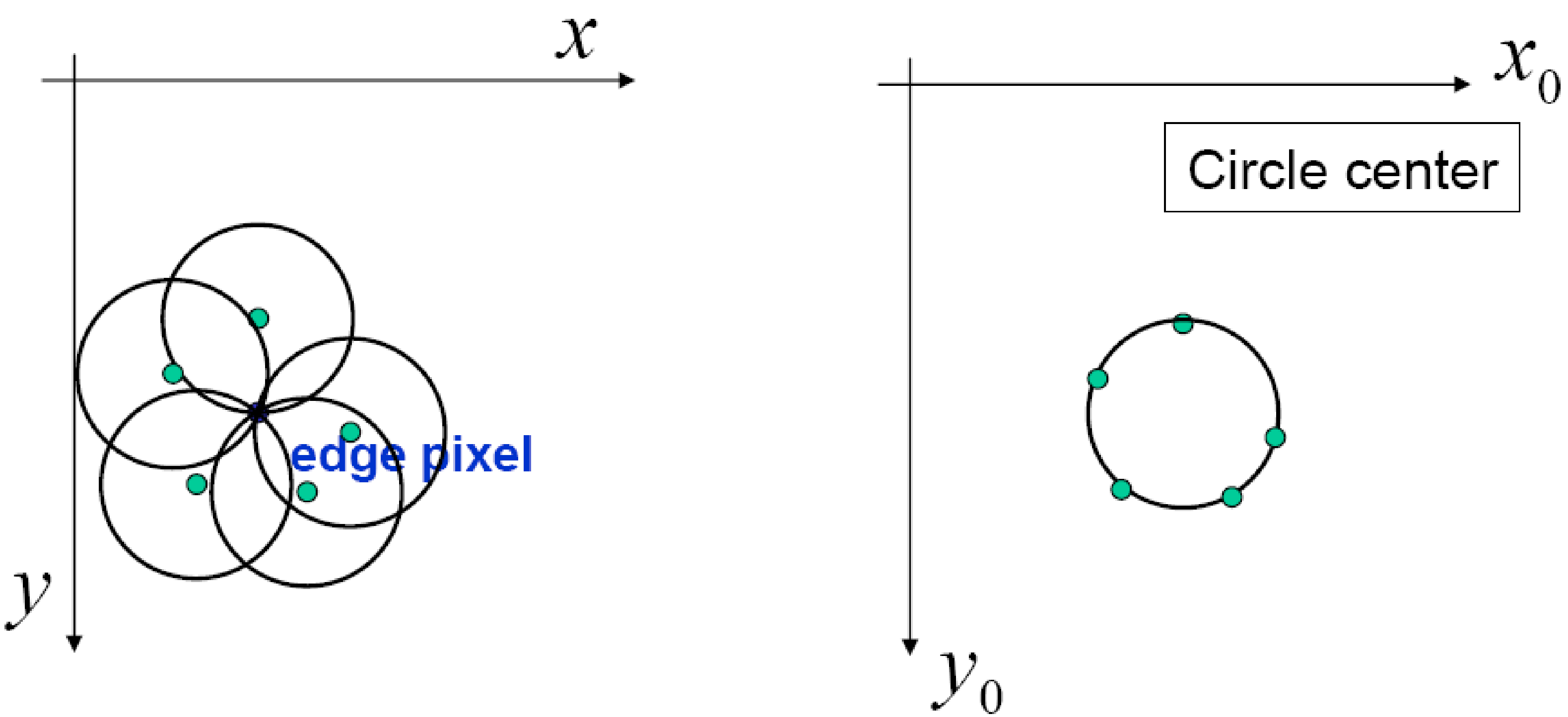
Original IC image (256x256)



Circle detection by Hough transform

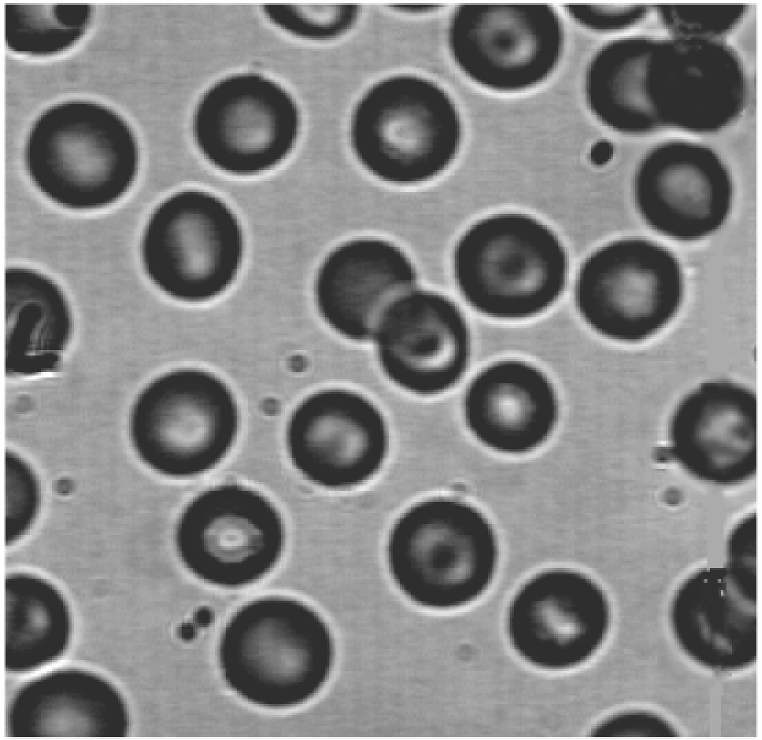
Find circles of fixed radius r

For circles of undetermined radius, use 3-d Hough transform for parameters (x_0, y_0, r)

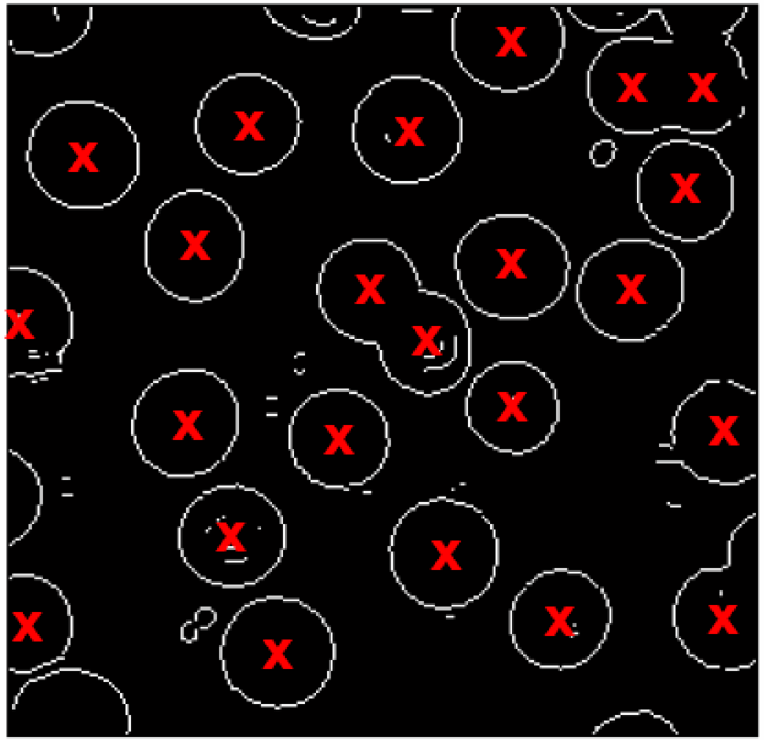


Example: circle detection by Hough transform

Original *blood* image



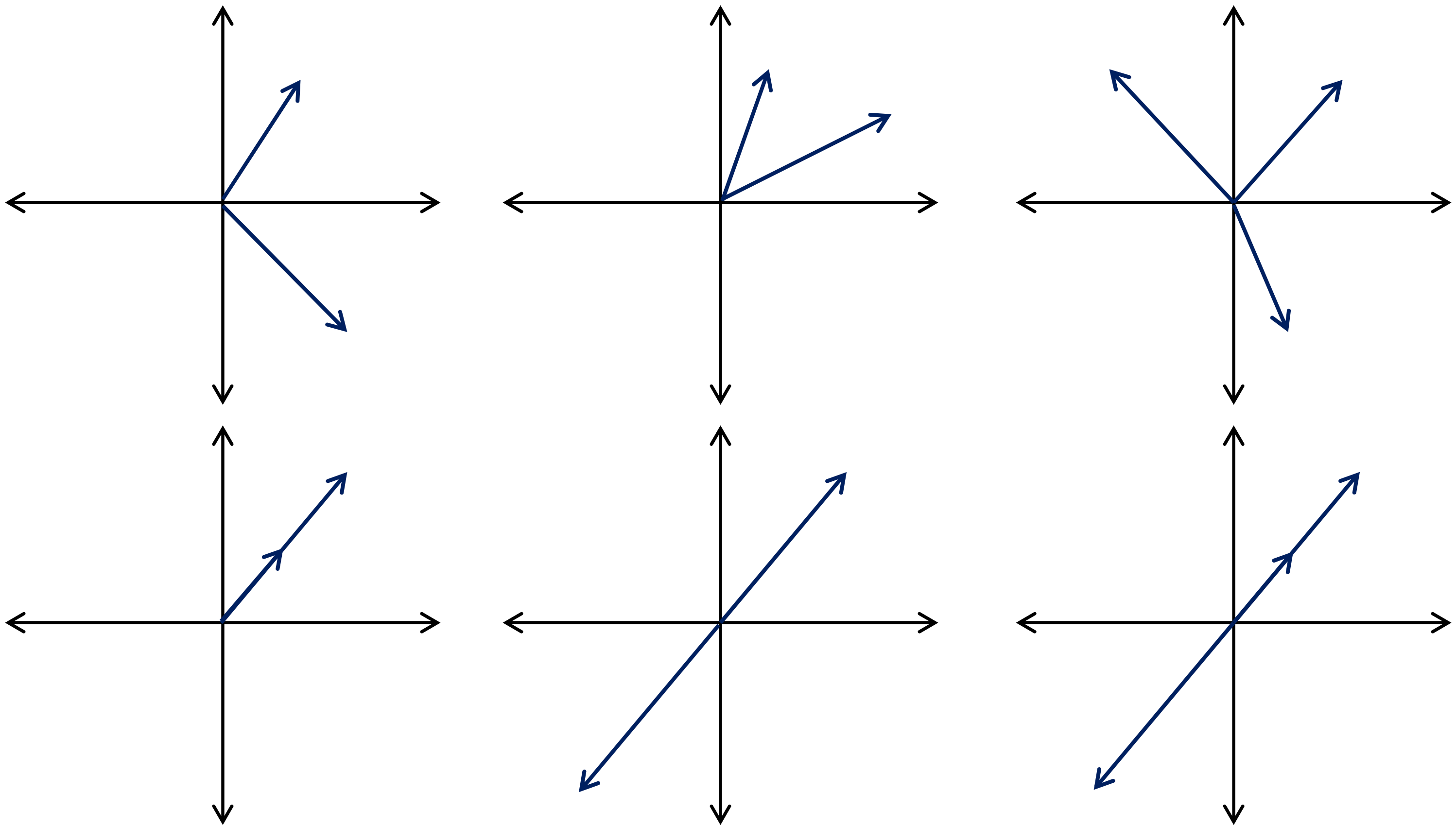
Prewitt edge detection



Linear Independence

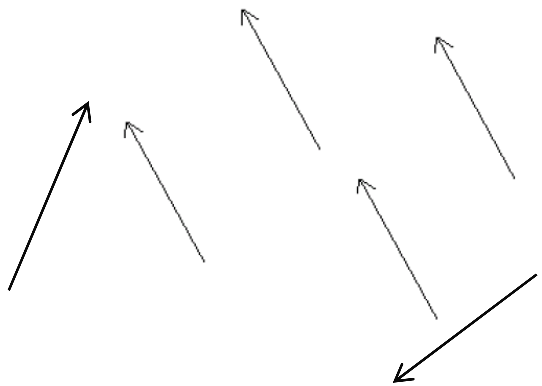
Different direction vectors are independent.

Parallel vectors are dependent. (+ve/-ve directions are parallel)

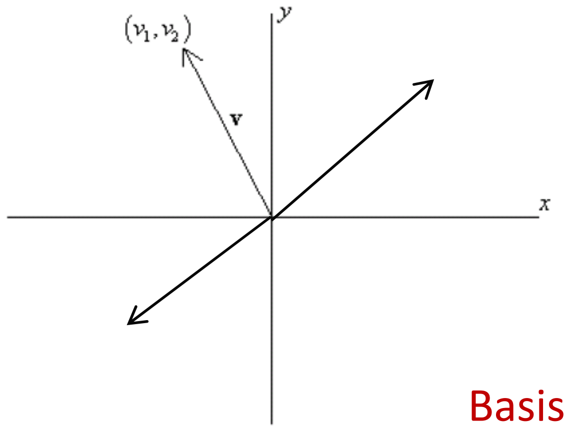


Basis & Orthonormal Bases

Basis (or axes): frame of reference



vs



Basis

$$\begin{bmatrix}2\\3\end{bmatrix}=2.\begin{bmatrix}1\\0\end{bmatrix}+3.\begin{bmatrix}0\\1\end{bmatrix};\ \begin{bmatrix}-9\\6\end{bmatrix}=-9.\begin{bmatrix}1\\0\end{bmatrix}+6.\begin{bmatrix}0\\1\end{bmatrix}\longrightarrow\left\{\begin{bmatrix}1\\0\end{bmatrix},\begin{bmatrix}0\\1\end{bmatrix}\right\}$$

is a set of representative for $\mathbb{R}^2$

$$\begin{bmatrix}2\\3\end{bmatrix}=\frac{9}{7}.\begin{bmatrix}2\\1\end{bmatrix}+\frac{4}{7}.\begin{bmatrix}-1\\3\end{bmatrix};\begin{bmatrix}-9\\6\end{bmatrix}=-3.\begin{bmatrix}2\\1\end{bmatrix}+3.\begin{bmatrix}-1\\3\end{bmatrix}\longrightarrow\left\{\begin{bmatrix}2\\1\end{bmatrix},\begin{bmatrix}-1\\3\end{bmatrix}\right\}$$

is another Set of representative for $\mathbb{R}^2$

Basis & Orthonormal Bases

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

$$\begin{bmatrix}-9\\6\end{bmatrix} = -9.\begin{bmatrix}1\\0\end{bmatrix} + 6.\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\begin{bmatrix}-9\\6\end{bmatrix};$$

Linear Transformation

$$\begin{bmatrix}2\\3\end{bmatrix} = \frac{9}{7}.\begin{bmatrix}2\\1\end{bmatrix} + \frac{4}{7}.\begin{bmatrix}-1\\3\end{bmatrix} = \begin{bmatrix}2 & -1\\1 & 3\end{bmatrix}\begin{bmatrix}9/7\\4/7\end{bmatrix};$$

$$\begin{bmatrix}-9\\6\end{bmatrix} = -3.\begin{bmatrix}2\\1\end{bmatrix} + 3.\begin{bmatrix}-1\\3\end{bmatrix} = \begin{bmatrix}2 & -1\\1 & 3\end{bmatrix}\begin{bmatrix}-3\\3\end{bmatrix};$$

Basis & Orthonormal Bases

Basis: a space is totally defined by a set of vectors – any vector is a *linear combination* of the basis

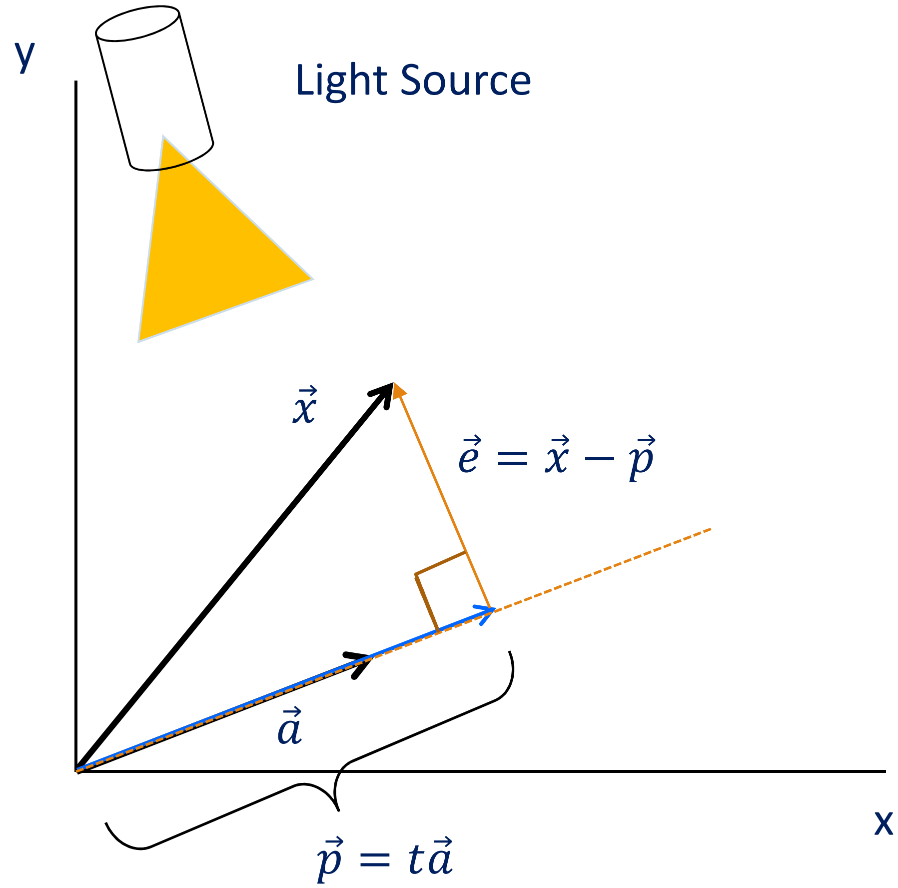
Ortho-Normal: orthogonal + normal

Orthogonal: dot product is zero
Normal: magnitude is one

$$\vec{x} = [1 \quad 0 \quad 0]^T \qquad \vec{x}.\vec{y} = 0$$
$$\vec{y} = [0 \quad 1 \quad 0]^T \qquad \vec{x}.\vec{z} = 0$$
$$\vec{z} = [0 \quad 0 \quad 1]^T \qquad \vec{y}.\vec{z} = 0$$

Projection: Using Inner Products

Projection of $\vec{x}$ along the direction $\vec{a}$



$$\cos 90^0 = \frac{\vec{a}.\vec{e}}{||\vec{a}||\,||\vec{e}||}$$

$$\Rightarrow \vec{a}.\vec{e} = 0$$

$$\Rightarrow \vec{a}.(\vec{x} - \vec{p}) = 0$$

$$\Rightarrow \vec{a}.(\vec{x} - t\vec{a}) = 0$$

$$\Rightarrow \vec{a}.\vec{x} - t\vec{a}.\vec{a} = 0$$

$$\Rightarrow t = \frac{\vec{a}.\vec{x}}{\vec{a}.\vec{a}}$$

Projection vector $\vec{p} = t\vec{a}$

$$\vec{p} = \vec{a}\,\frac{\vec{a}^T\vec{x}}{\vec{a}^T\vec{a}} = \frac{\vec{a}\vec{a}^T}{\vec{a}^T\vec{a}}\vec{x}$$

Projection Matrix

Eigenvalues & Eigenvectors

Eigenvectors (for a square $m \times m$ matrix **S**)

$$S\vec{v} = \lambda\vec{v}$$

(right) eigenvector

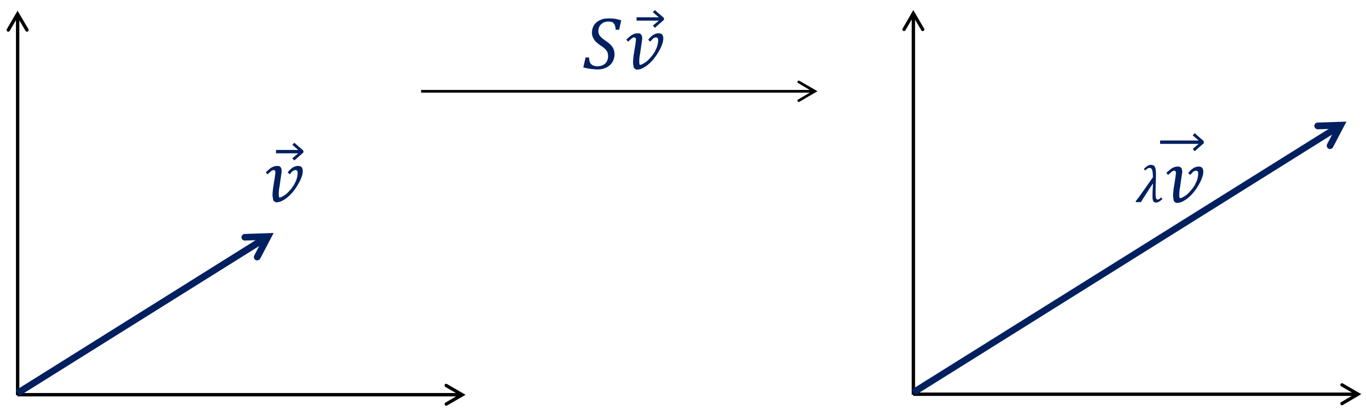
$\vec{v} \epsilon \mathbb{R}^m \neq \vec{0}$

eigenvalue

$\lambda \epsilon \mathbb{R}$

Example

$$\begin{pmatrix} 6 & -2 \\ 4 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \end{pmatrix} = 2 \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$



Eigenvalues & Eigenvectors (Properties)

The eigenvalues of real symmetric matrices are real.

$$A\vec{v} = \lambda\vec{v}$$

- Then we can conjugate to get $\bar{A}\bar{\vec{v}}=\bar{\lambda}\bar{\vec{v}}$
- If the entries of A are real, this becomes A$\bar{\vec{v}}=\bar{\lambda}\bar{\vec{v}}$
- This proves that complex eigenvalues of real valued matrices come in conjugate pairs
- Now transpose to get $\bar{\vec{v}}^t A^T = \bar{\vec{v}}^t\bar{\lambda}$. Because A is symmetric matrix we now have $\bar{\vec{v}}^t A = \bar{\vec{v}}^t\bar{\lambda}$
- Multiply both sides of this equation on the right with $\vec{v}$, i.e. $\bar{\vec{v}}^t A\vec{v} = \bar{\vec{v}}^t\bar{\lambda}\vec{v}$
- On the other hand multiply$A\vec{v} = \lambda\vec{v}$ on the left by $\bar{\vec{v}}^t$ to get $\bar{\vec{v}}^t A\vec{v} = \bar{\vec{v}}^t\lambda\vec{v}$

$$\Rightarrow \bar{\vec{v}}^t\bar{\lambda}\vec{v} = \bar{\vec{v}}^t\lambda\vec{v} \Rightarrow \bar{\lambda} = \lambda \Rightarrow \lambda\ is\ real$$

Eigenvalues & Eigenvectors (Properties)

If A is an n x n symmetric matrix, then any two eigenvectors that come from distinct eigenvalues are orthogonal.

$$A\vec{v}_i = \lambda_i \vec{v}_i$$

- Left multiply with $\vec{v}_j^T$ to $A\vec{v}_i = \lambda_i \vec{v}_i \;\Rightarrow \vec{v}_j^T A\vec{v}_i = \vec{v}_j^T \lambda_i \vec{v}_i$
- Similarly $\vec{v}_i^T A\vec{v}_j = \vec{v}_i^T \lambda_j \vec{v}_j$
- From the above two equations $(\lambda_j - \lambda_i)\vec{v}_i^T \vec{v}_j = 0$

$$\Longrightarrow \vec{v}_i^T \vec{v}_j = 0$$

$\therefore$ $\vec{v}_j$ and $\vec{v}_i$ are perpendicular

Diagonalization

Stack up all the eigen vectors to get

$$\mathrm{AV} = \mathrm{V}\Lambda$$

Where $V = [\vec{v}_1 \quad \vec{v}_2 \quad \dots \quad \vec{v}_n]$; $\Lambda = diag(\lambda_1 \quad \lambda_2 \quad \dots \quad \lambda_n)$

If all eigenvectors are linearly independent , V is invertible.

$$A = V\Lambda V^{-1}$$

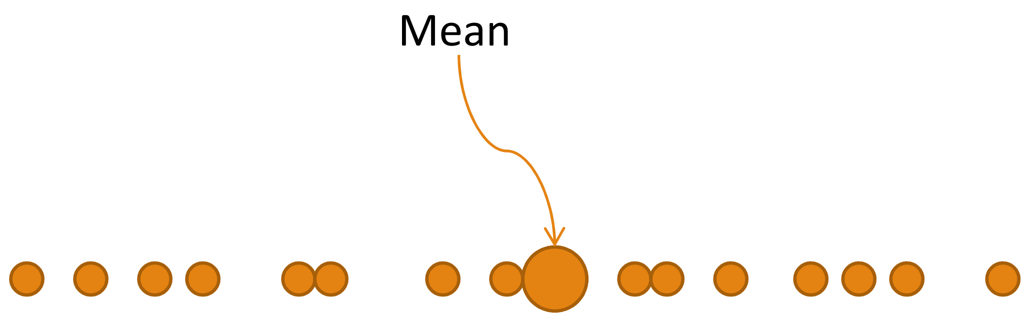
Suppose A is symmetric matrix, then $V^T = V^{-1}$

Therefore $A = V\Lambda V^T$

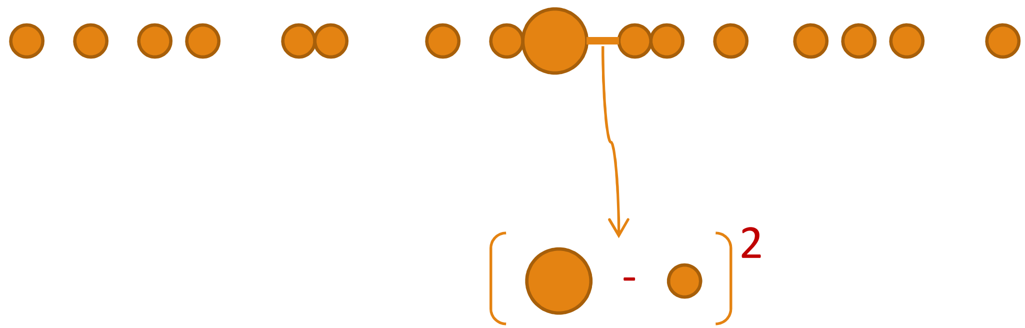
Variance

Variance is the average squared deviation from the mean of a set of data. It is used to find the **standard deviation**.

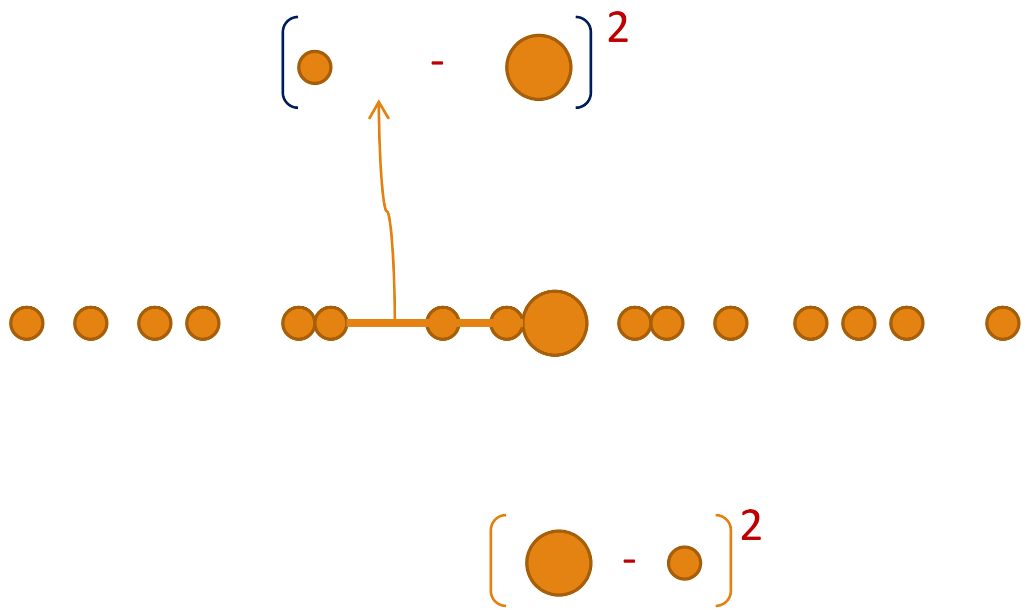
Variance



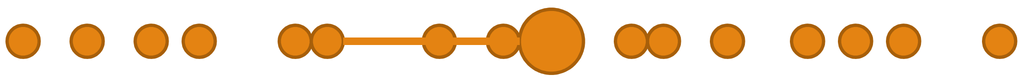
Variance

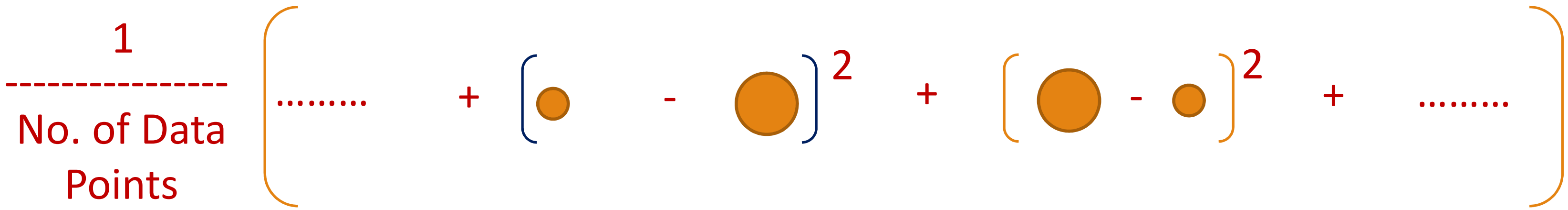


Variance



Variance





Variance

Find the mean

Find the deviation of each value from the mean

Square the deviations

Sum the squared deviations

Divide the sum by *n*

(gives typical squared deviation from mean)

Variance Formula

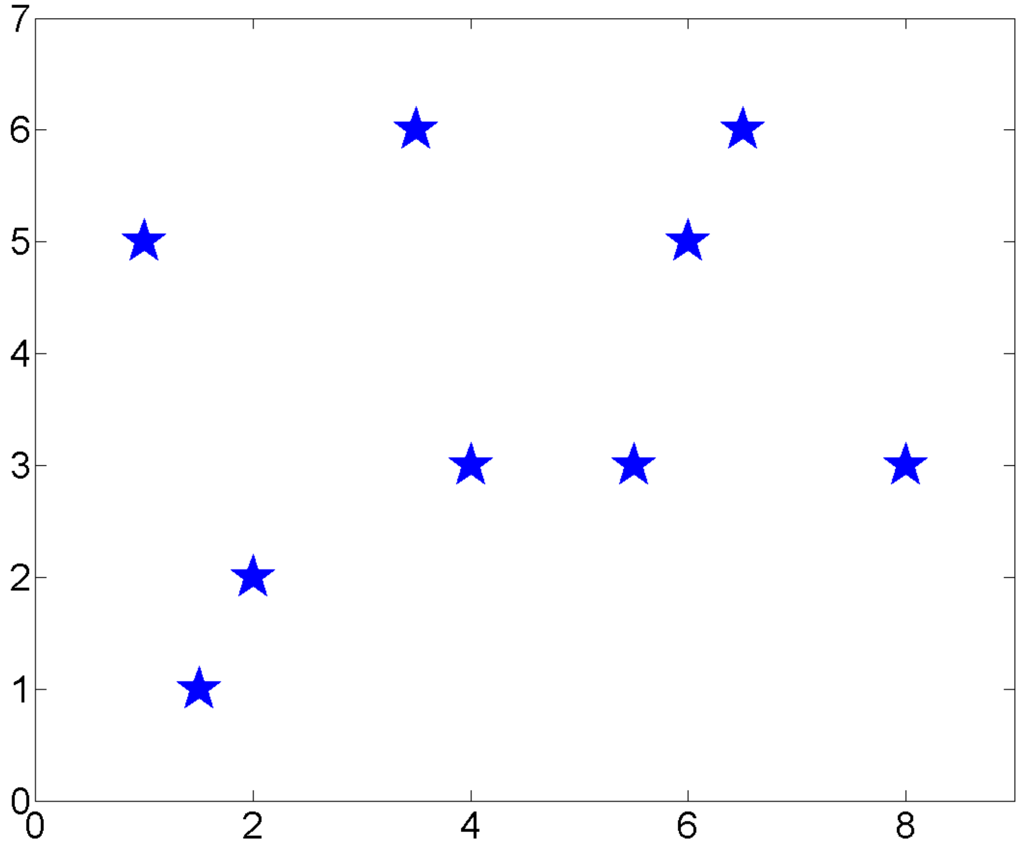
$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Standard Deviation

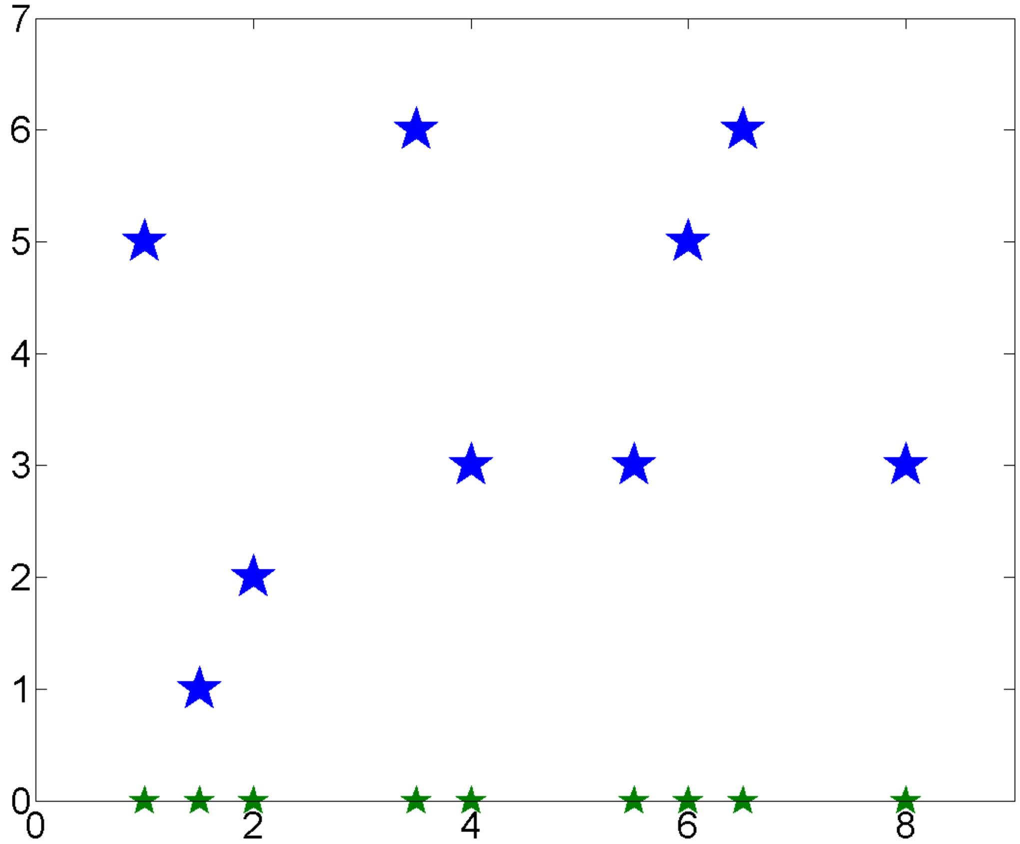
$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

[standard deviation = square root of the variance]

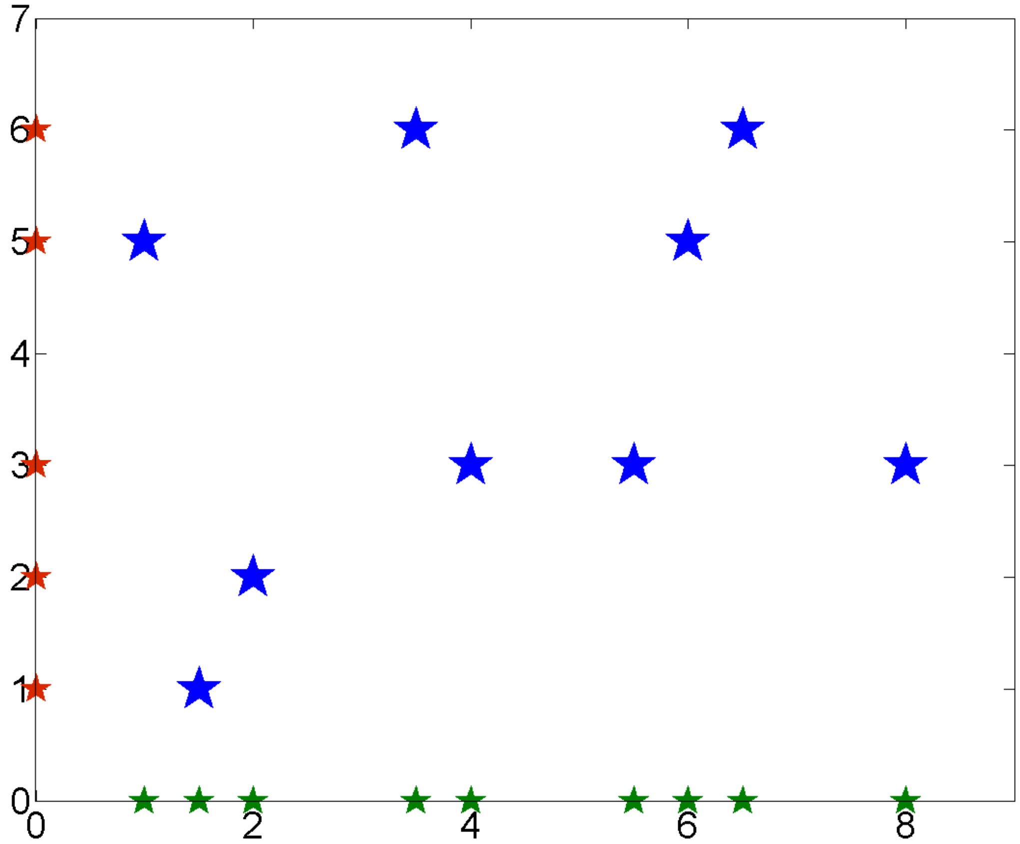
Variance (2D)



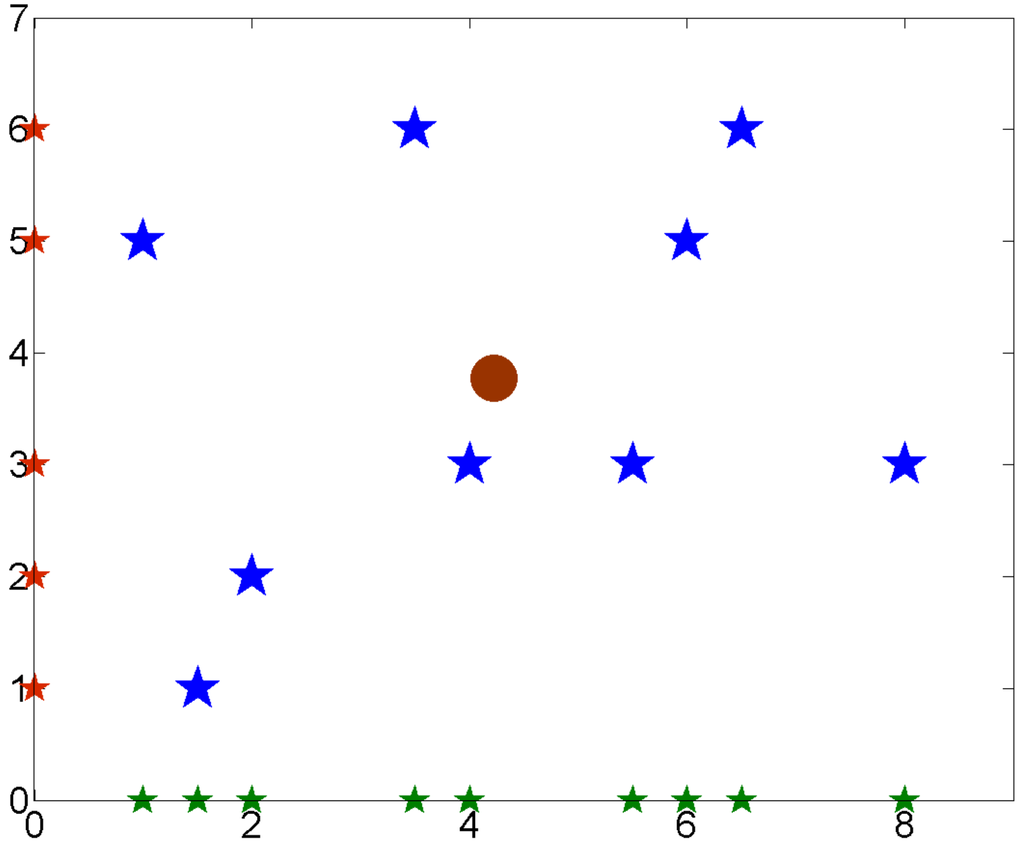
Variance (2D)



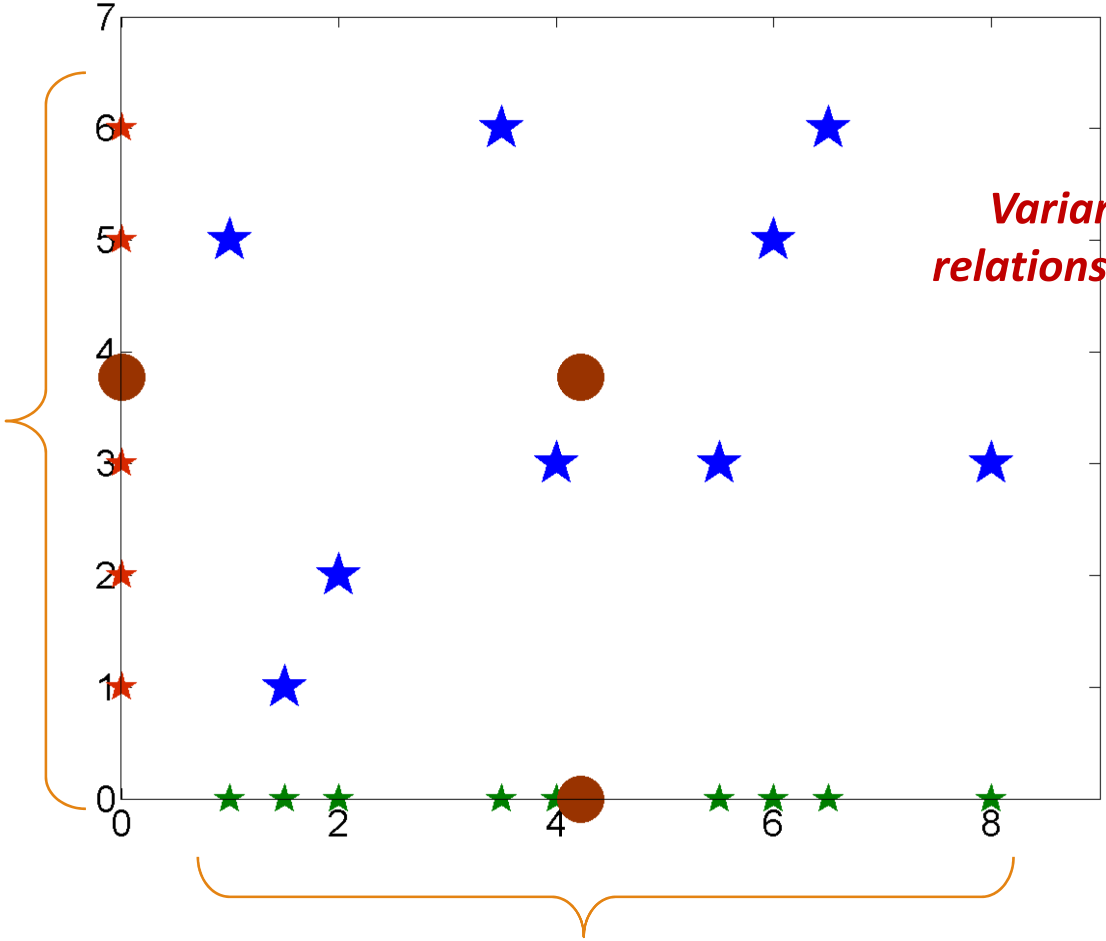
Variance (2D)



Variance (2D)



Variance (2D)



Variance doesn't explore relationship between variables

Covariance

$$\text{Variance(x)} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

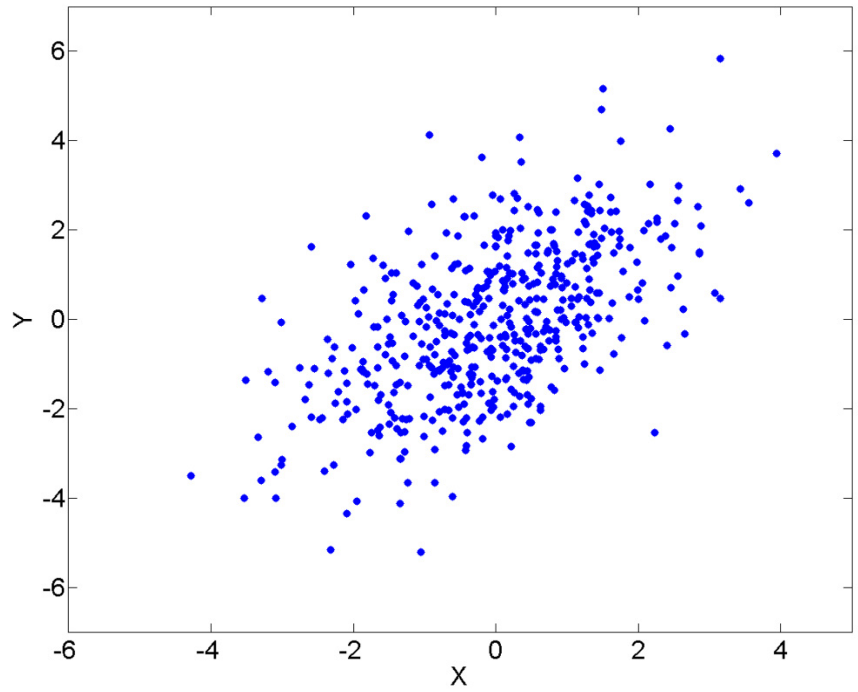
$$= \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})$$

$$\text{Covariance}(\text{x}, \text{y}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

- $\text{Covariance}(\text{x}, \text{x}) = \text{var}(\text{x})$
- $\text{Covariance}(\text{x}, y) = \text{Covariance}(\text{y}, \text{x})$

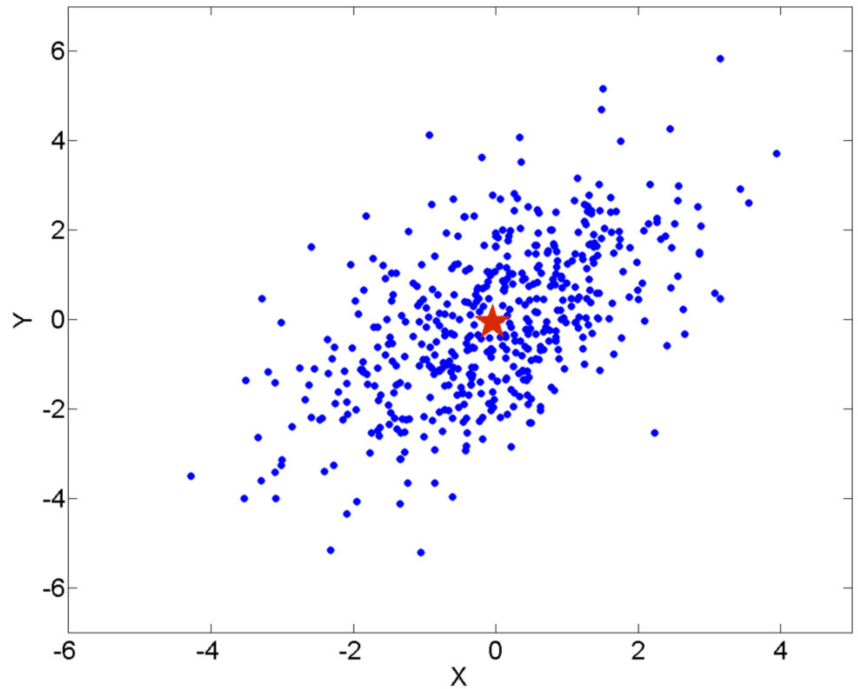
Covariance

$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



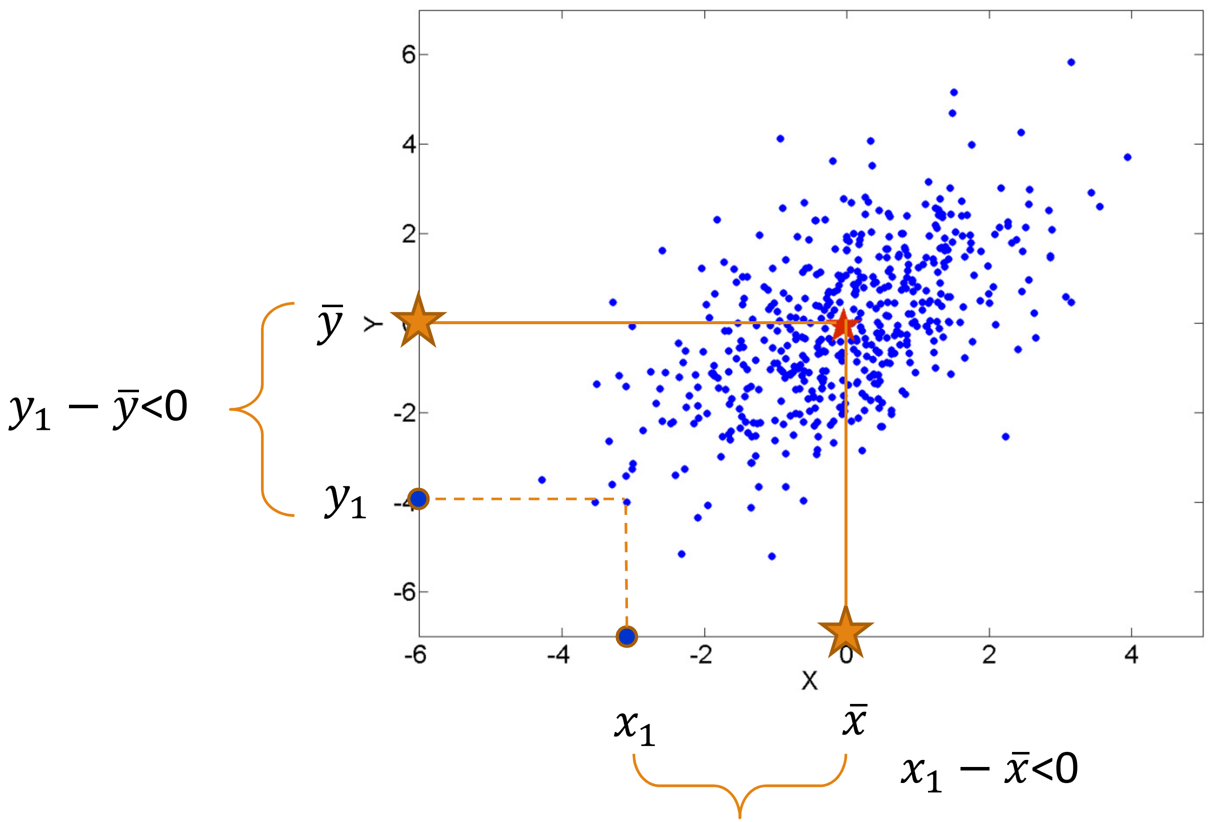
Covariance

$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



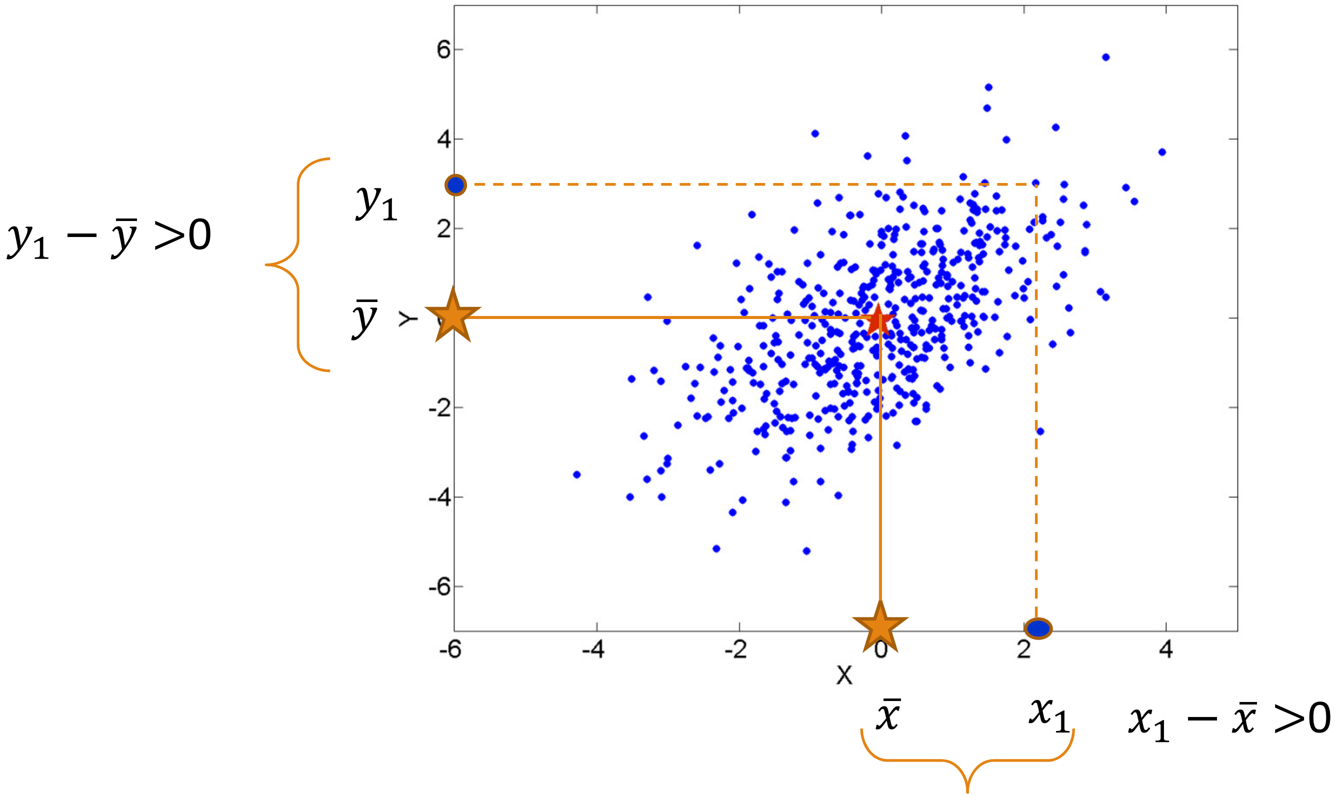
Covariance

$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



Covariance

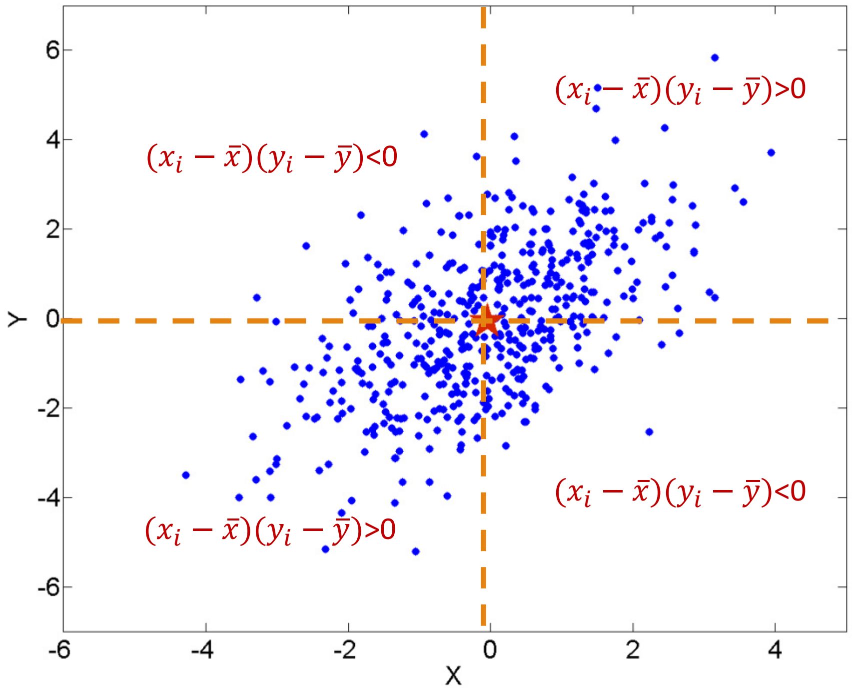
$$\text{Covariance}(x,y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



Covariance

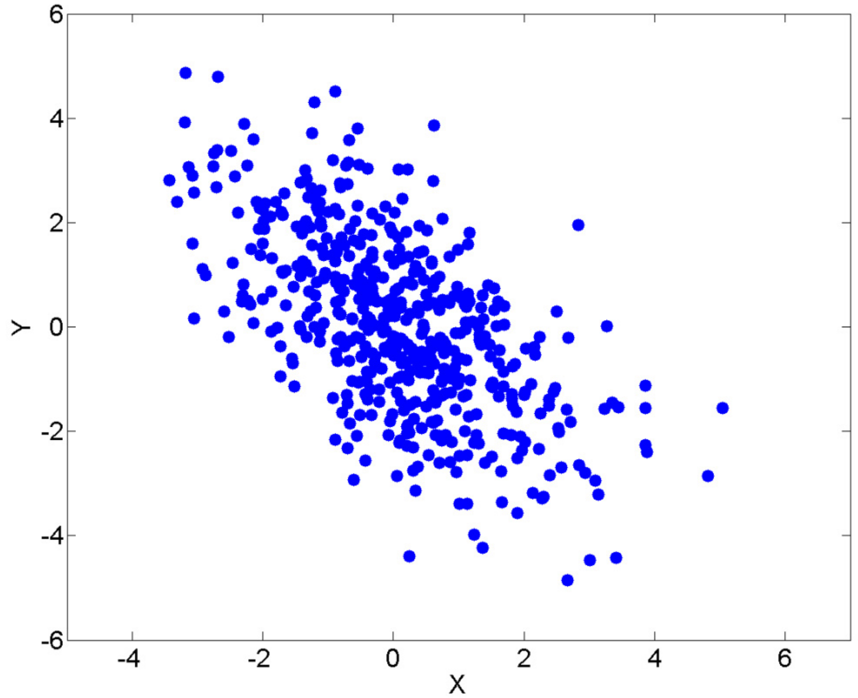
$$\text{Covariance(x, y)} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

Positive Relation



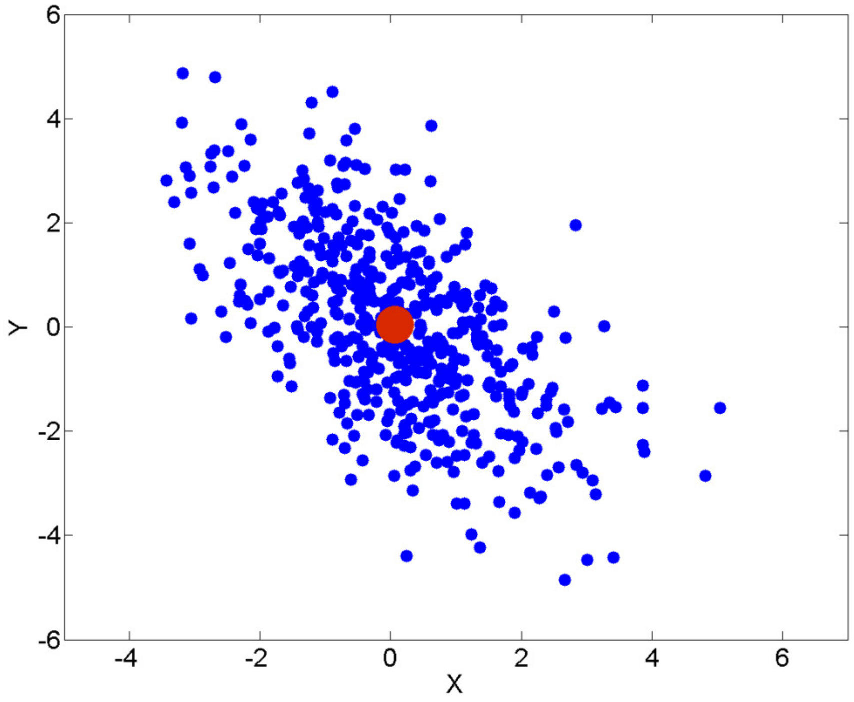
Covariance

$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



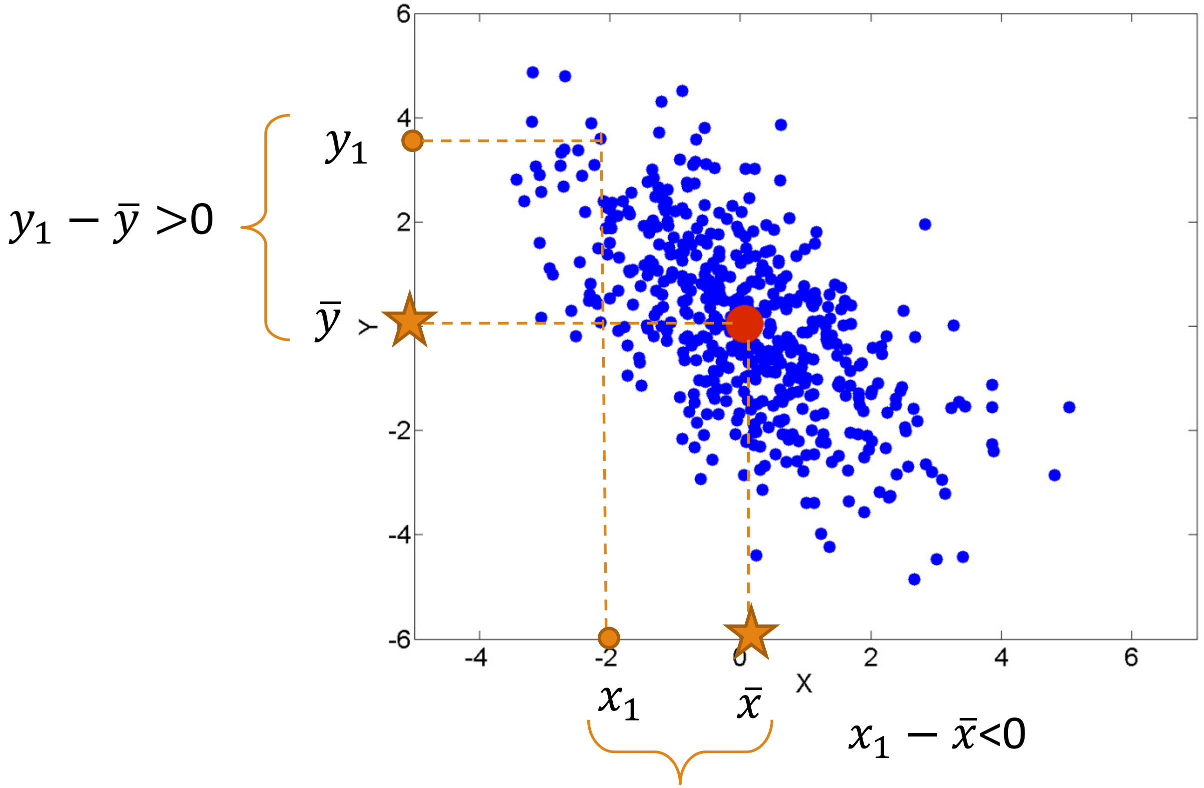
Covariance

$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



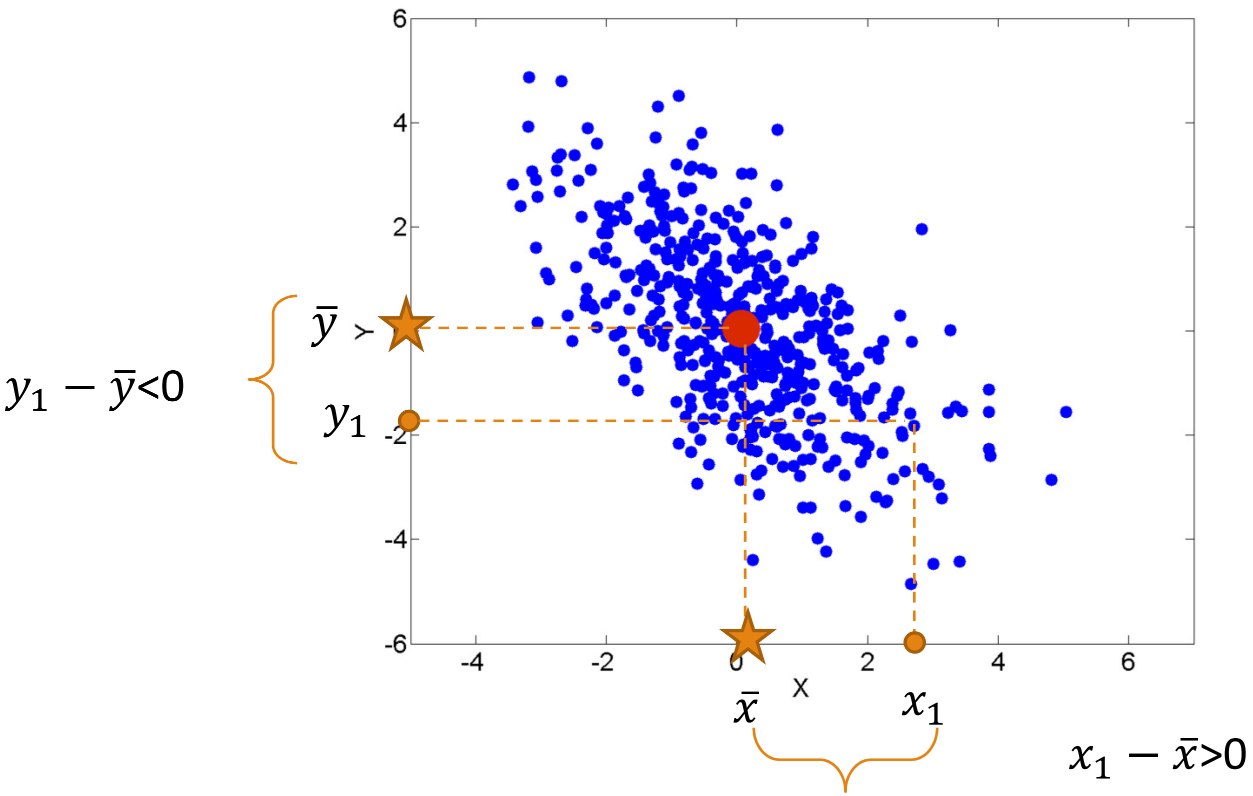
Covariance

$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



Covariance

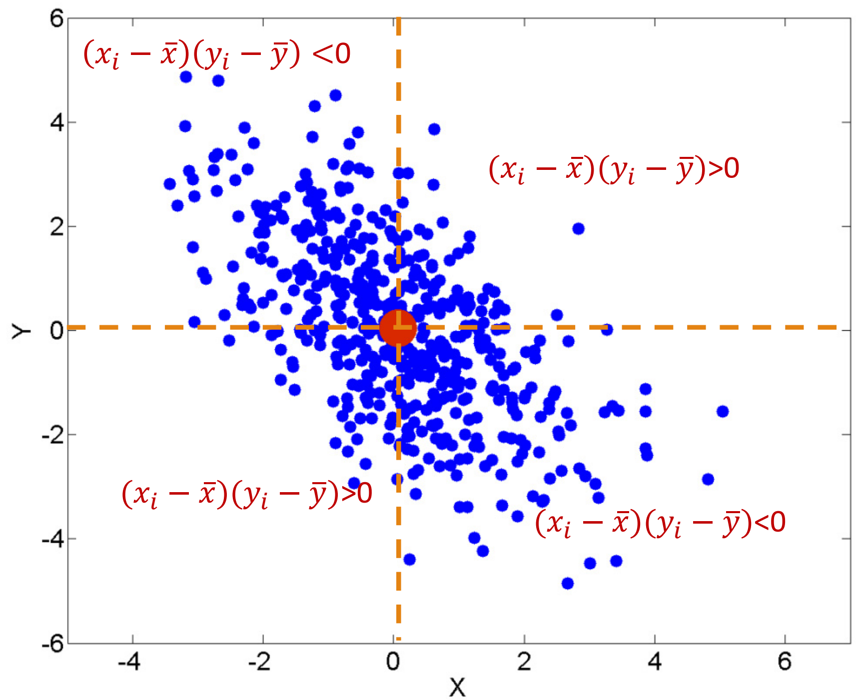
$$\text{Covariance}(\mathrm{x},\mathrm{y}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



Covariance

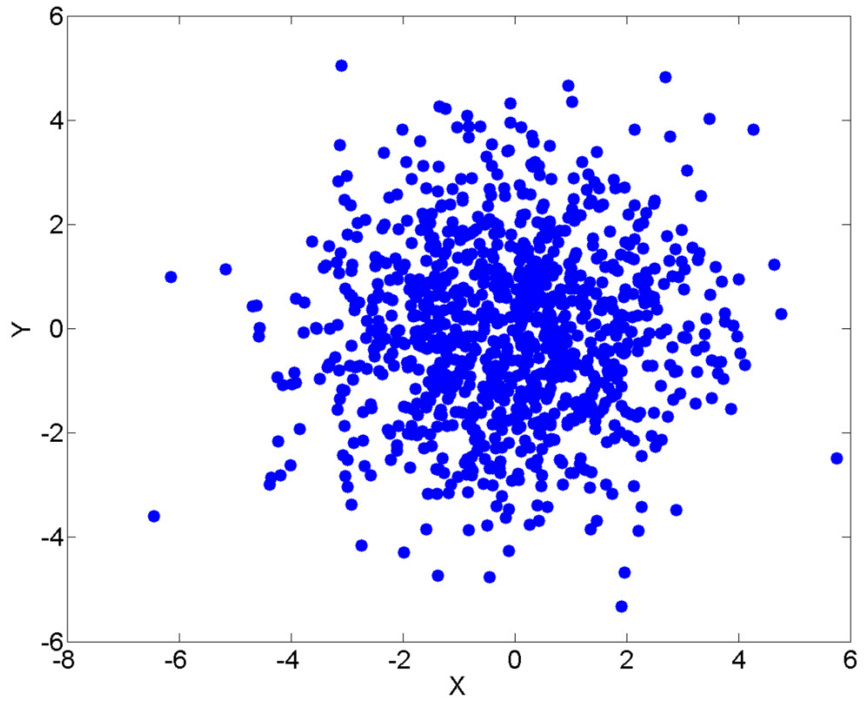
$$\text{Covariance}(\mathrm{x},\mathrm{y}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

Negative Relation



Covariance

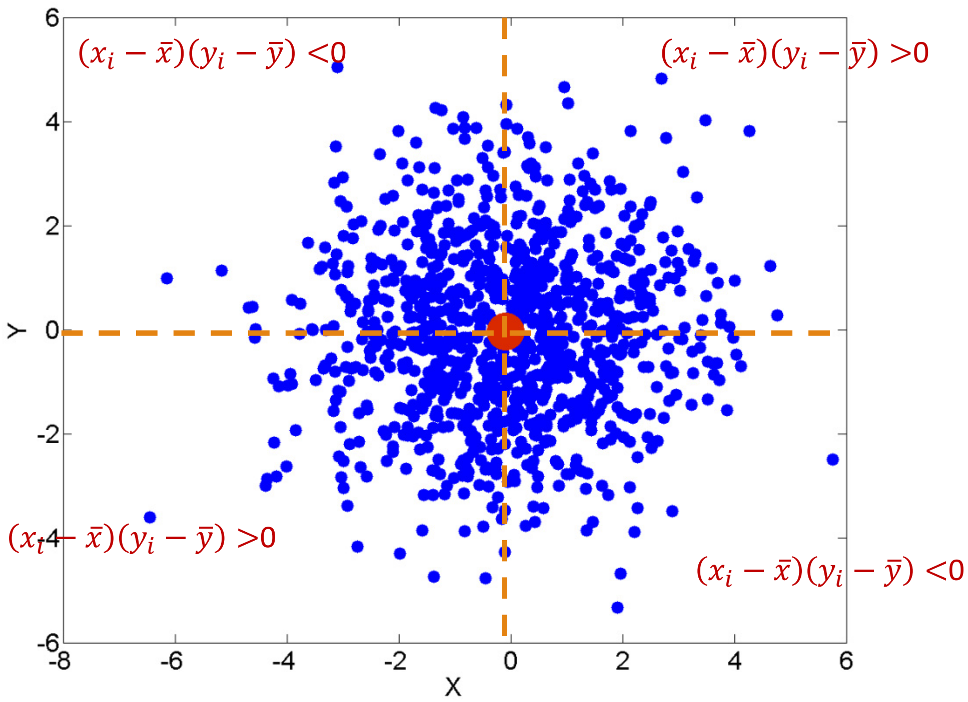
$$\text{Covariance}(x, y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$



Covariance

$$\text{Covariance}(\mathrm{x},\mathrm{y}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

No Relation



Covariance Matrix

$$Cov\ (\Sigma) = \begin{bmatrix} cov(x_1, x_1) & cov(x_1, x_2) & \cdots & cov(x_1, x_m) \\ cov(x_2, x_1) & cov(x_2, x_2) & \cdots & cov(x_2, x_m) \\ \vdots & \vdots & \vdots & \vdots \\ cov(x_m, x_1) & cov(x_m, x_2) & \cdots & cov(x_m, x_m) \end{bmatrix}$$

$$Cov\ (\Sigma) = \frac{1}{n}(X - \bar{X})(X - \bar{X})^T; where\ X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

Covariance Matrix

$$Cov\ (\Sigma) = \begin{bmatrix} cov(x_1, x_1) & cov(x_1, x_2) & \cdots & cov(x_1, x_m) \\ cov(x_2, x_1) & cov(x_2, x_2) & \cdots & cov(x_2, x_m) \\ \vdots & \vdots & \vdots & \vdots \\ cov(x_m, x_1) & cov(x_m, x_2) & \cdots & cov(x_m, x_m) \end{bmatrix}$$

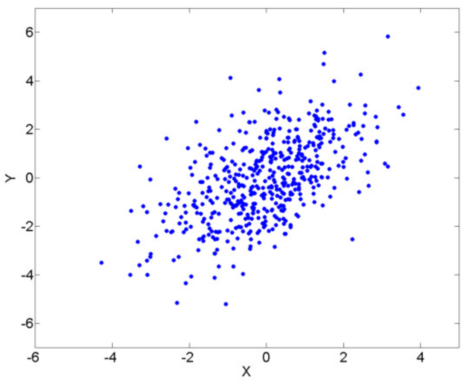
- Diagonal elements are variances, i.e. Cov(x, x)=$var(x)$.
- Covariance Matrix is symmetric.
- It is a positive semi-definite matrix.

Covariance Matrix

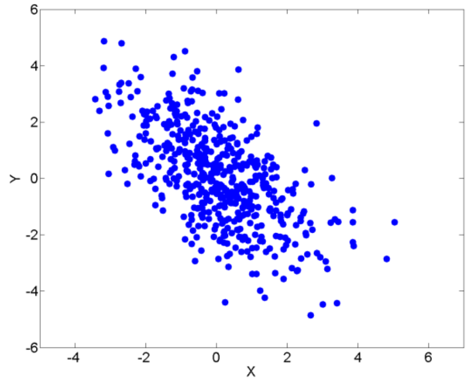
- Covariance is a real symmetric positive semi-definite matrix.
  - All eigenvalues must be real
  - Eigenvectors corresponding to different eigenvalues are orthogonal
  - All eigenvalues are greater than or equal to zero
  - Covariance matrix can be diagonalized,

    i.e. $\mathrm{Cov} = \mathrm{PDP}^{\mathrm{T}}$

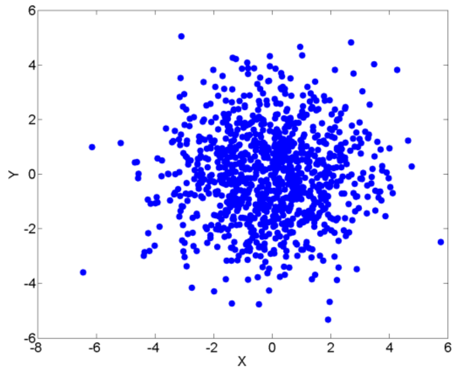
Correlation



Positive relation



Negative relation



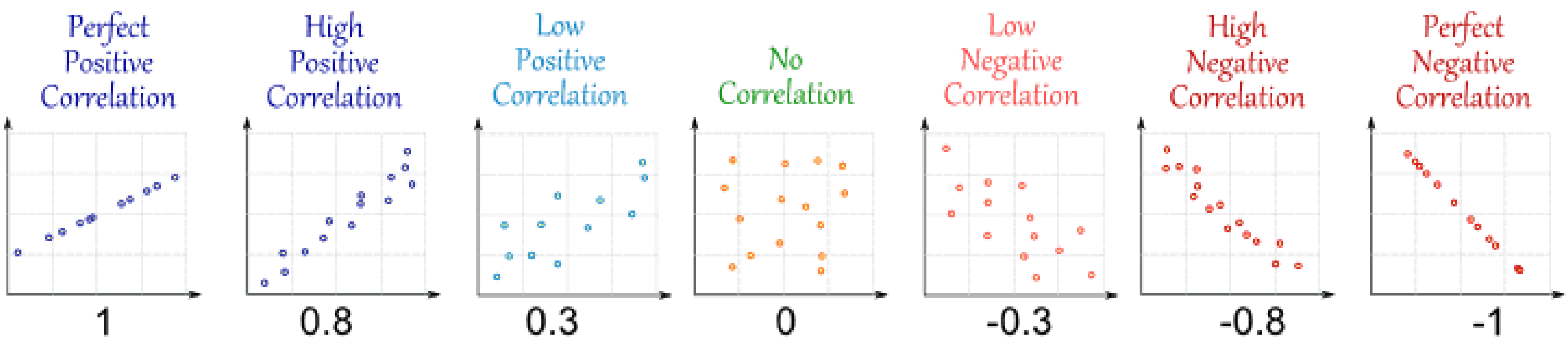
No relation

- Covariance determines whether relation is positive or negative, but it was impossible to measure the degree to which the variables are related.
- Correlation is another way to determine how two variables are related.
- In addition to whether variables are positively or negatively related, correlation also tells the degree to which the variables are related each other.

Correlation

$$\rho_{xy} = Correlation\,(x,y) = \frac{cov(x,y)}{\sqrt{var(x)}\sqrt{var(y)}}.$$

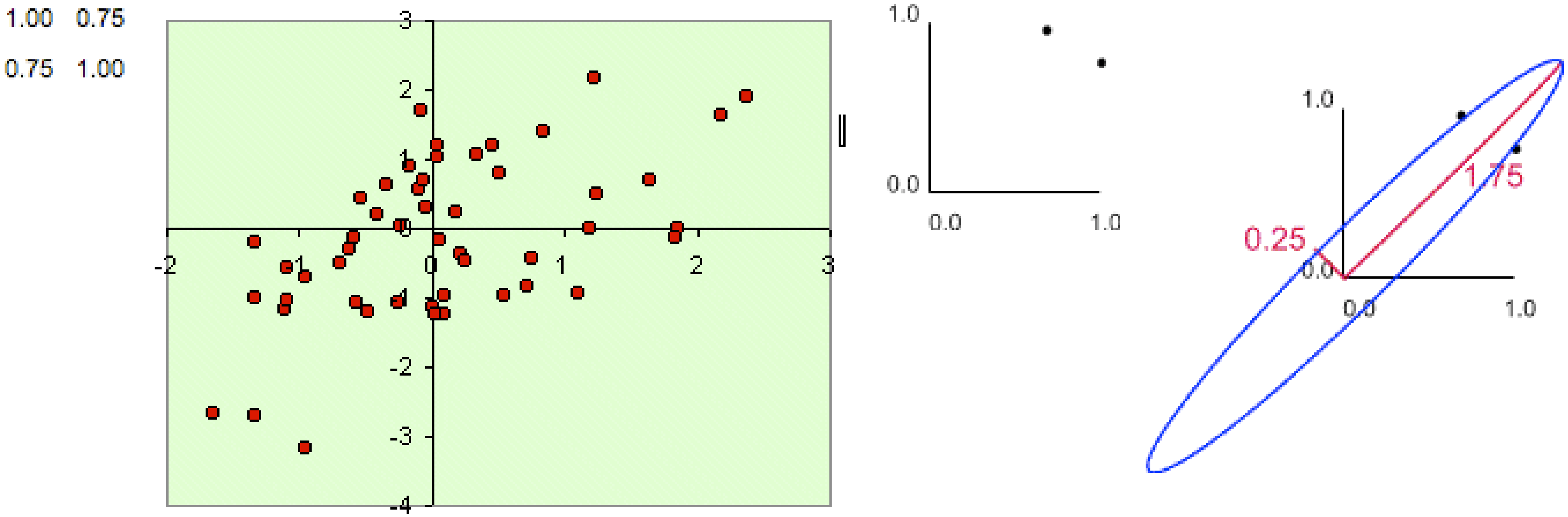
$$-1 \leq Correlation(x,y) \leq +1$$



Eivectors and Eigenvalues

We can interpret this correlation as an ellipse whose major axis is one eigenvalue and the minor axis length is the other:

No correlation yields a circle, and perfect correlation yields a line.

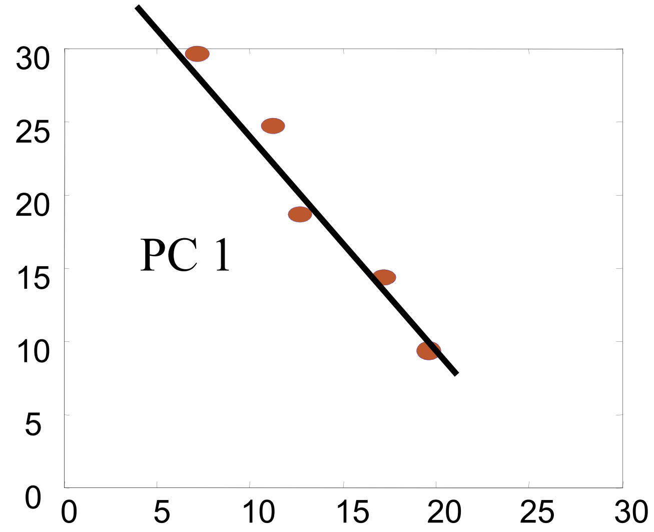


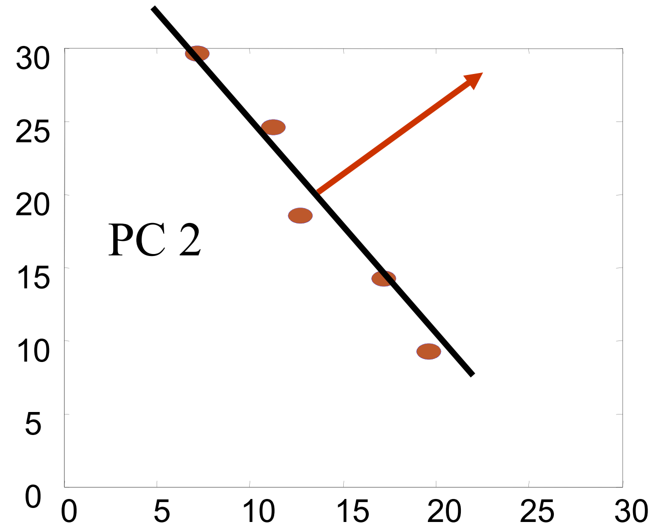
The Principal Components

All principal components (PCs) start at the origin of the ordinate axes.

First PC is direction of maximum variance from origin

Subsequent PCs are orthogonal to 1st PC and describe maximum residual variance



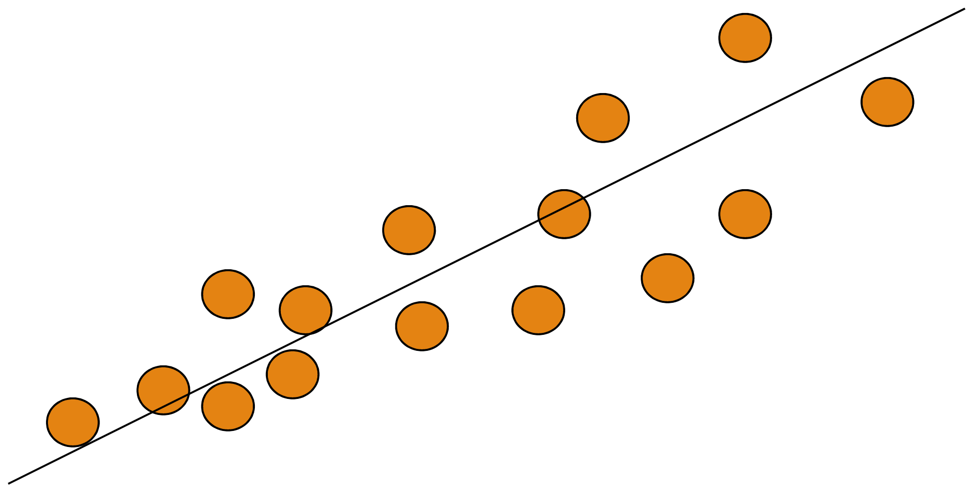


Algebraic Interpretation

Given m points in a n dimensional space, for large n, how does one project on to a low dimensional space while preserving broad trends in the data and allowing it to be visualized?

Algebraic Interpretation – 1D

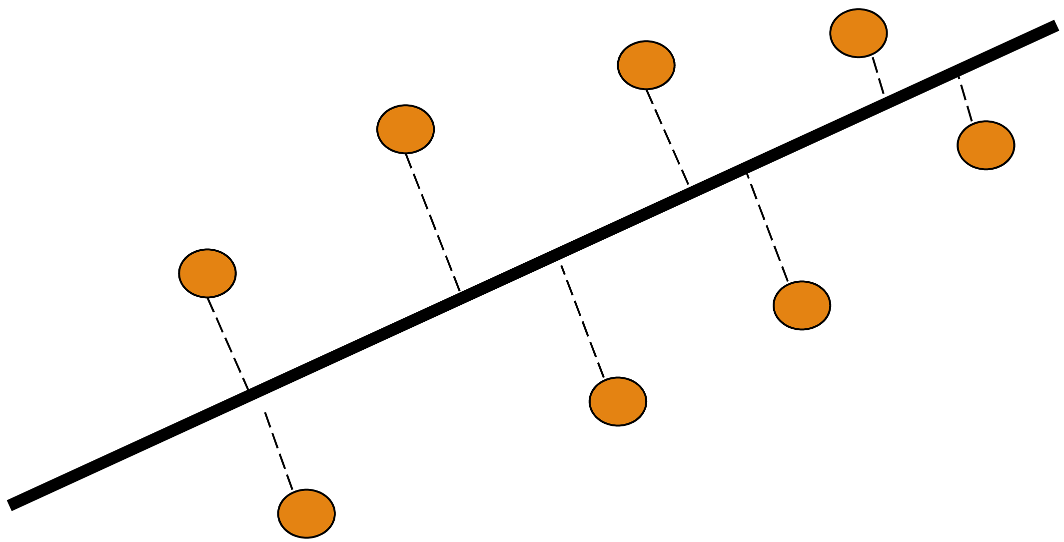
Given m points in a n dimensional space, for large n, how does one project on to a 1 dimensional space?



Choose a line that fits the data so the points are spread out well along the line

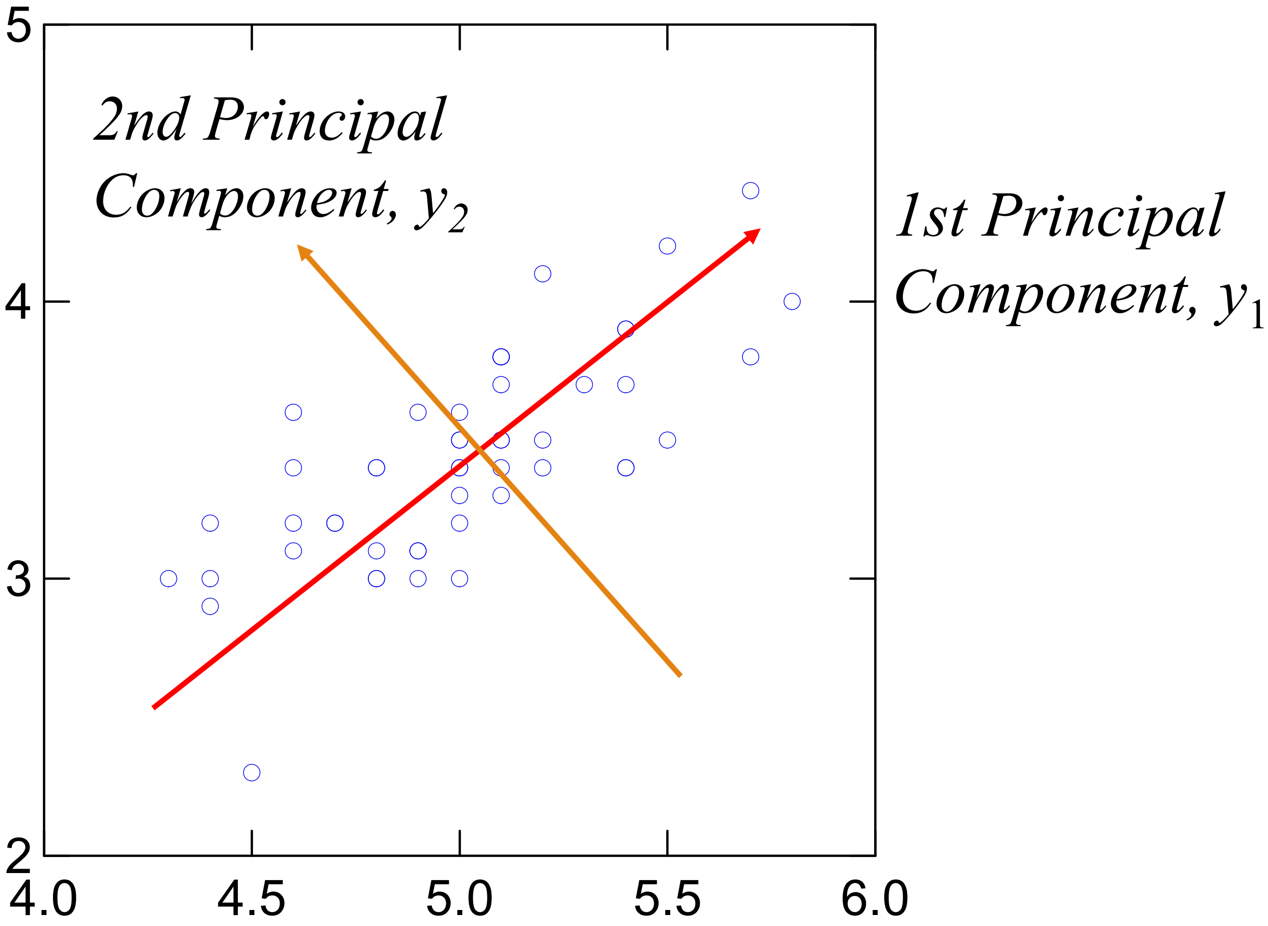
Algebraic Interpretation – 1D

Formally, minimize sum of squares of distances to the line.

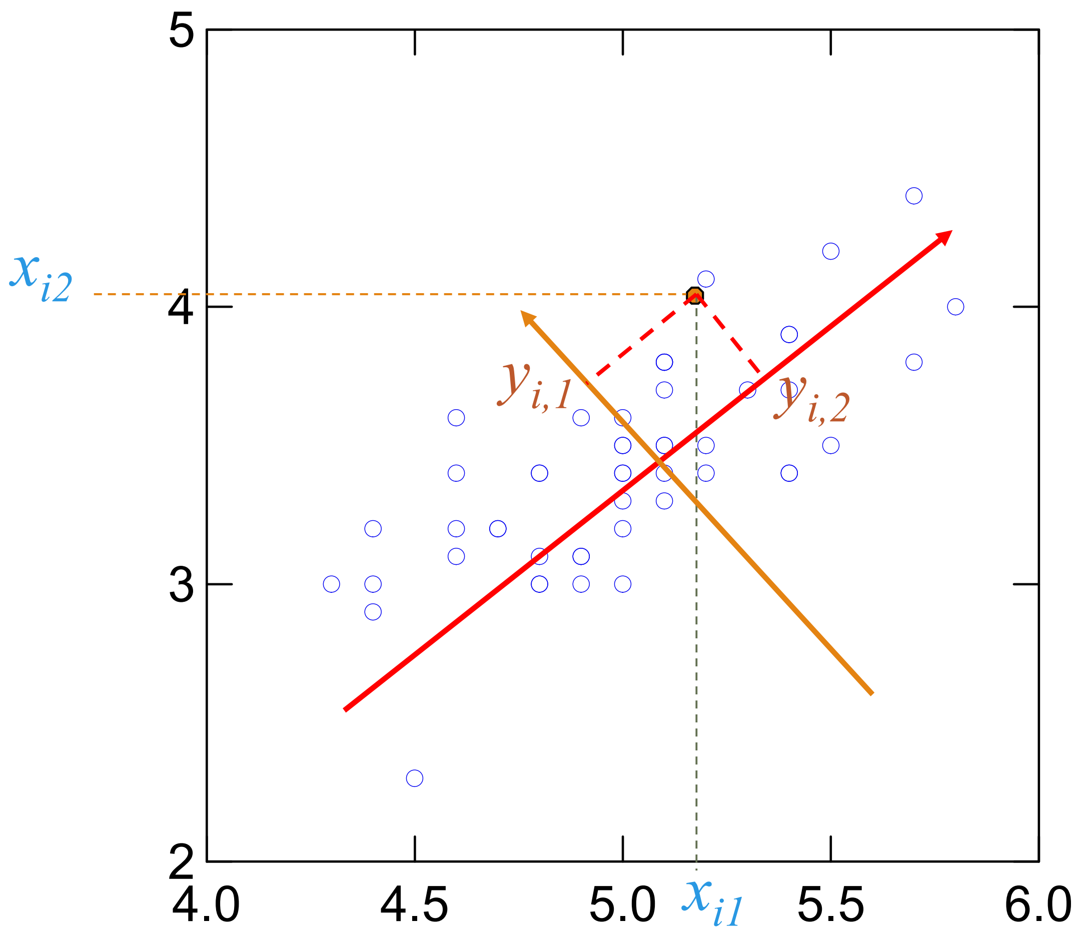


Why sum of squares? Because it allows fast minimization.

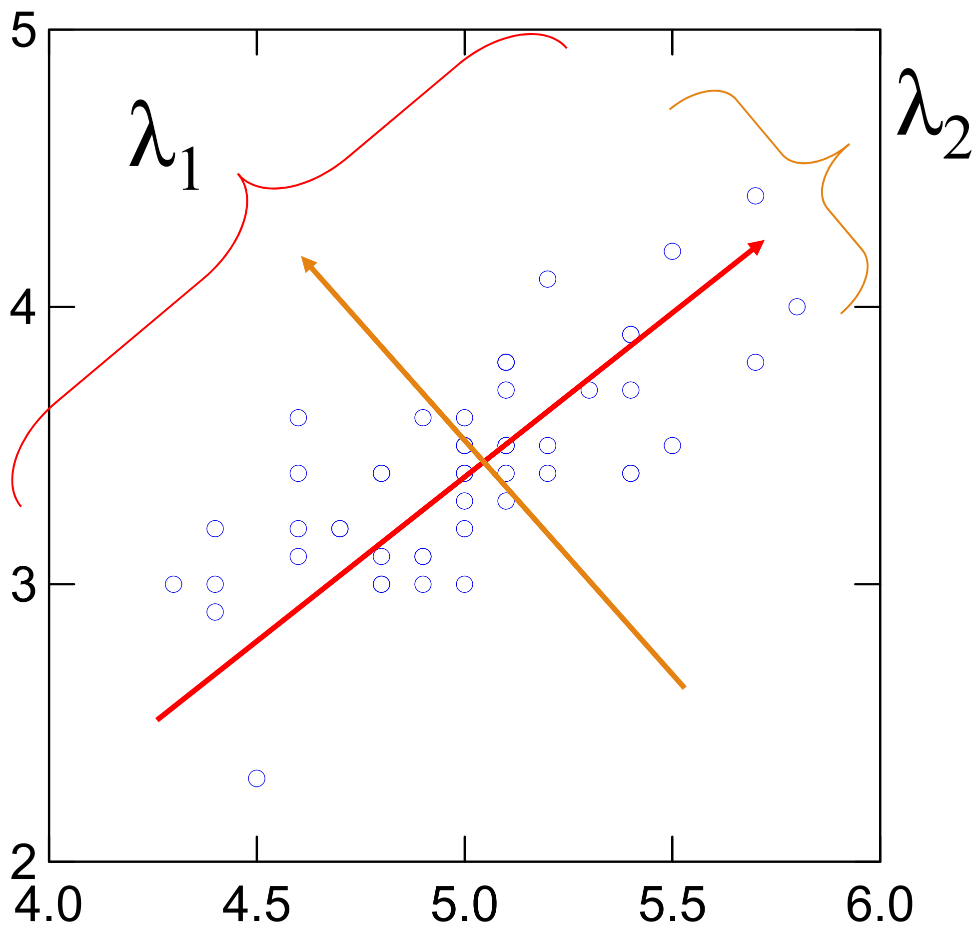
PCA: *2D representation*



PCA Scores

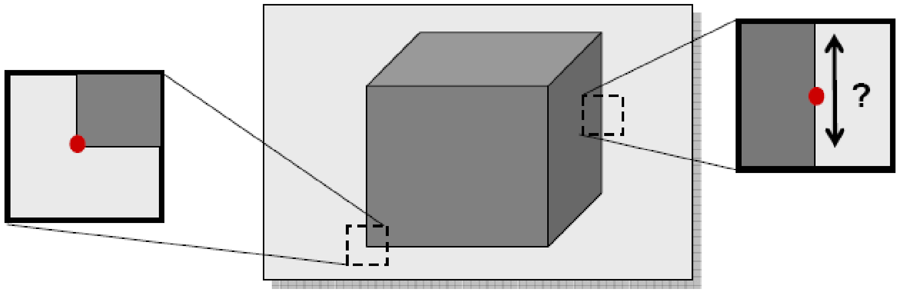


PCA Eigenvalues



Harris Corner Detector

Many applications benefit from features localized in *(x,y)*



Edges well localized only in one direction -> detect corners

Desirable properties of corner detector

- Accurate localization
- Invariance against shift, rotation, scale, brightness change
- Robust against noise, high repeatability

What patterns can be localized most accurately?

Local displacement sensitivity

$$S(\Delta x, \Delta y) = \sum_{(x,y)\in window} \left[f(x,y) - f(x+\Delta x, y+\Delta y) \right]^2$$

Linear approximation for small $\Delta x, \Delta y$

$$f(x+\Delta x, y+\Delta y) \approx f(x,y) + f_x(x,y)\Delta x + f_y(x,y)\Delta y$$

$$S(\Delta x, \Delta y) \approx \sum_{(x,y)\in window} \left[\begin{pmatrix} f_x(x,y) & f_y(x,y) \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \right]^2$$

Iso-sensitivity curves are ellipses

$$S(\Delta x, \Delta y) \approx \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \left(\sum_{(x,y)\in window} \begin{bmatrix} f_x^2(x,y) & f_x(x,y) f_y(x,y) \\ f_x(x,y) f_y(x,y) & f_y^2(x,y) \end{bmatrix} \right) \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} =$$

$$= \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \mathbf{M} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

Harris criterium

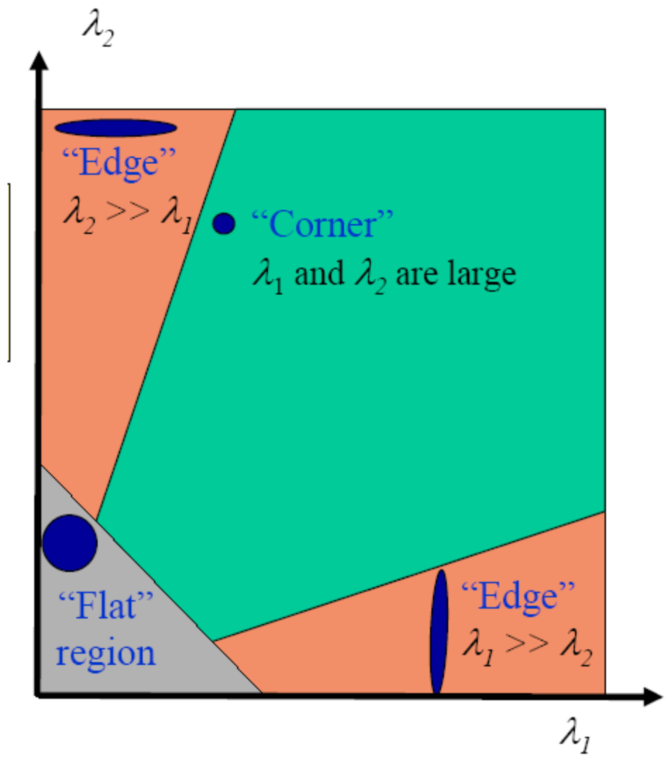
Often based on eigenvalues λ_1, λ_2 *of* "structure matrix" (or "normal matrix" or "second-moment matrix")

$$\mathbf{M} = \begin{bmatrix} \sum_{(x,y)\in window} f_x^2(x,y) & \sum_{(x,y)\in window} f_x(x,y) f_y(x,y) \\ \sum_{(x,y)\in window} f_x(x,y) f_y(x,y) & \sum_{(x,y)\in window} f_y^2(x,y) \end{bmatrix}$$

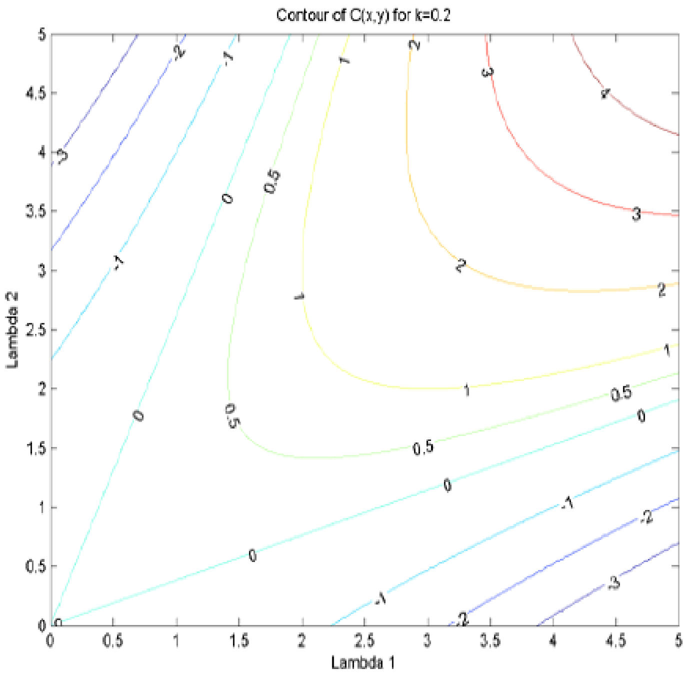
$f_x(x,y)$ – horizontal image gradient
$f_y(x,y)$ – vertical image gradient

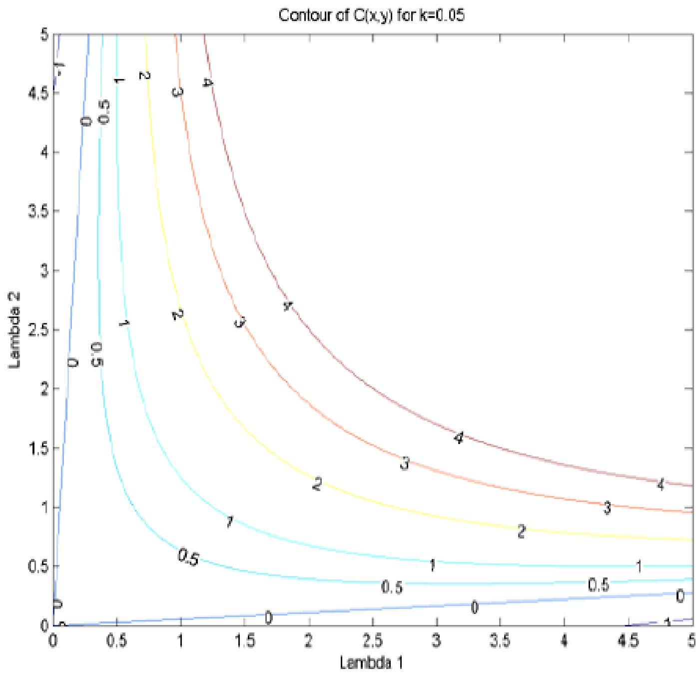
Measure of "cornerness"

$$C(x,y) = \det(\mathbf{M}) - k \cdot (trace(\mathbf{M}))^2 = \lambda_1 \lambda_2 - k \cdot (\lambda_1 + \lambda_2)$$



Harris corner values

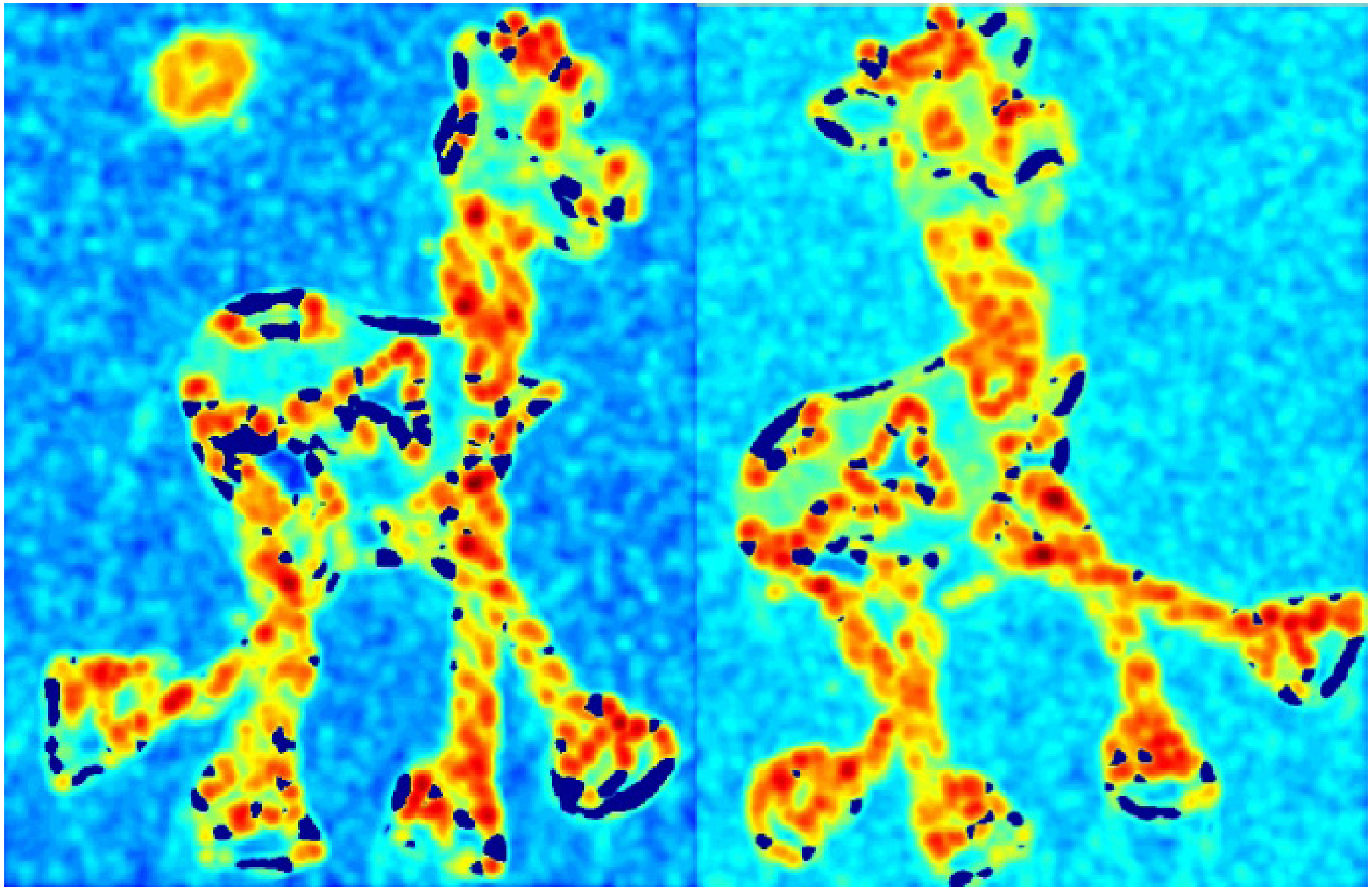




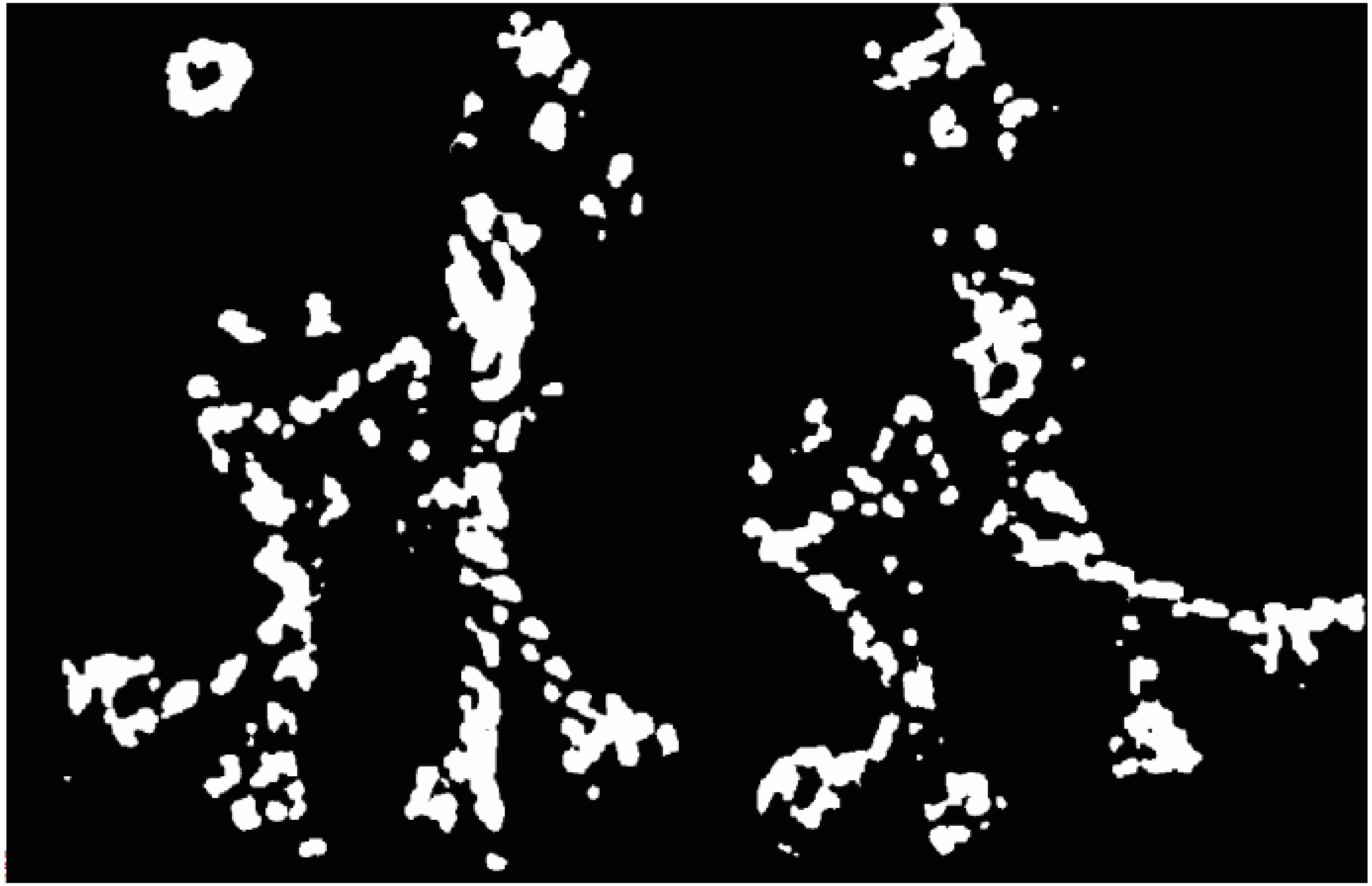
$$C(x,y) = \det(\mathbf{M}) - k \cdot \left(trace(\mathbf{M})\right)^2$$
$$= \lambda_1 \lambda_2 - k \cdot \left(\lambda_1 + \lambda_2\right)^2$$

Keypoint Detection: Input

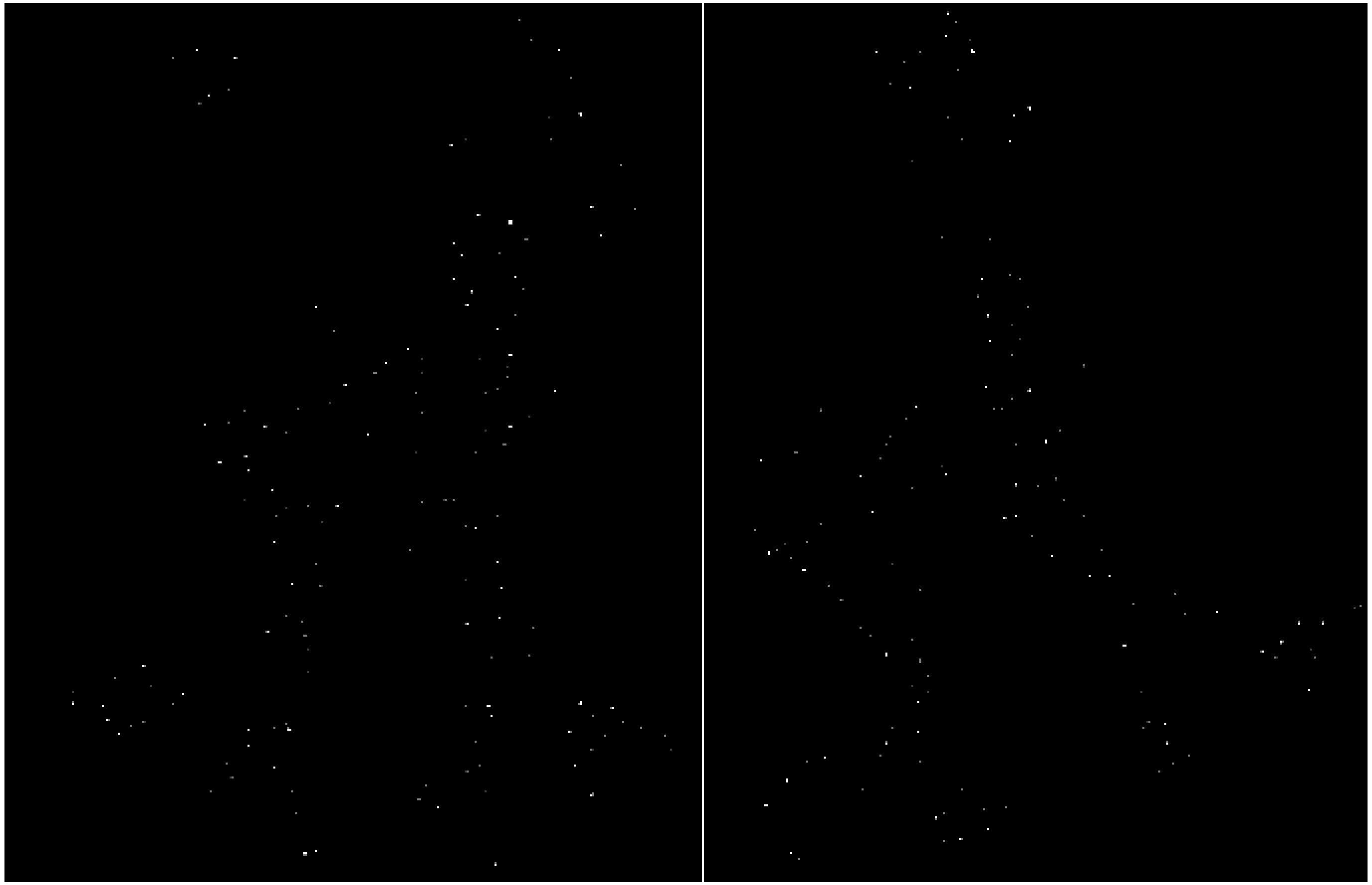
Harris cornerness



Thresholded cornerness



Local maxima of cornerness

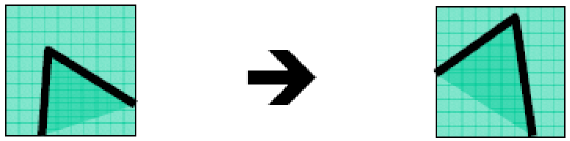


Superimposed keypoints

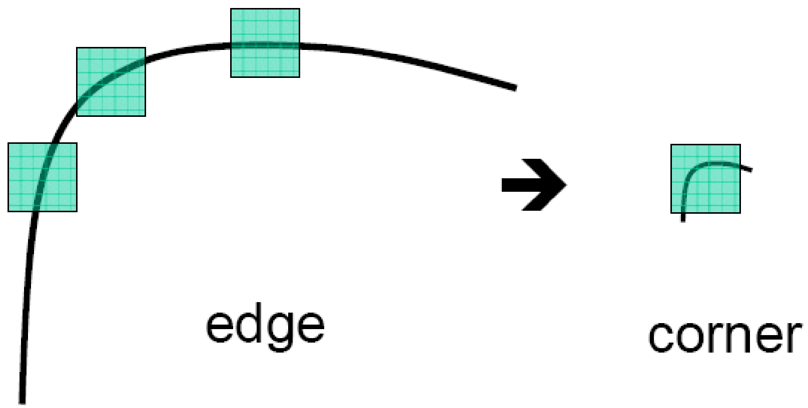
Robustness of Harris Corner Detector

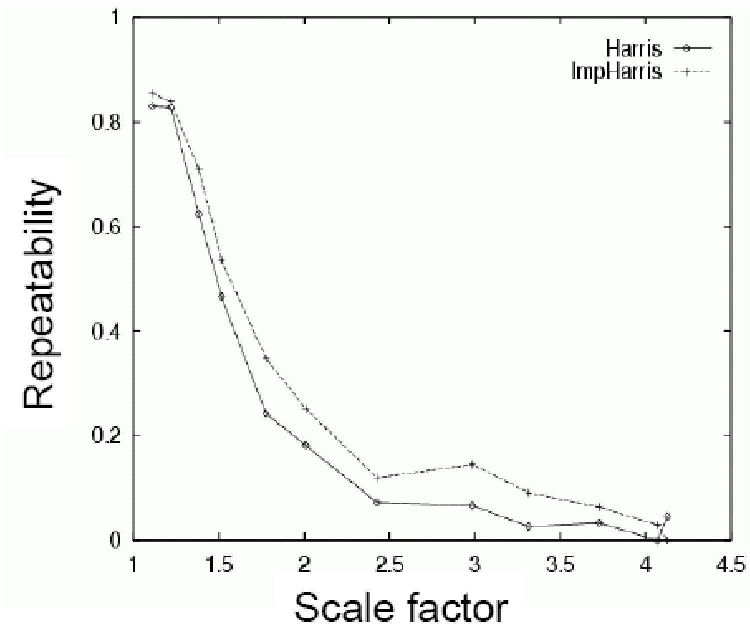
Invariant to brightness offset: $f(x,y) \rightarrow f(x,y) + c$

Invariant to shift and rotation

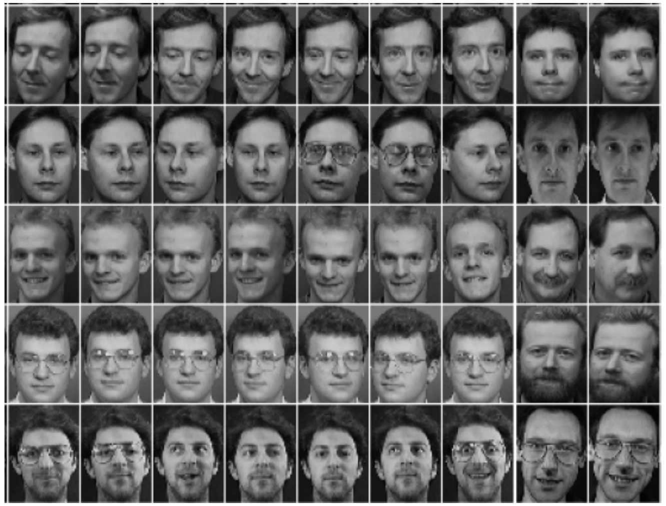


Not invariant to scaling





High-dimensional data in computer vision

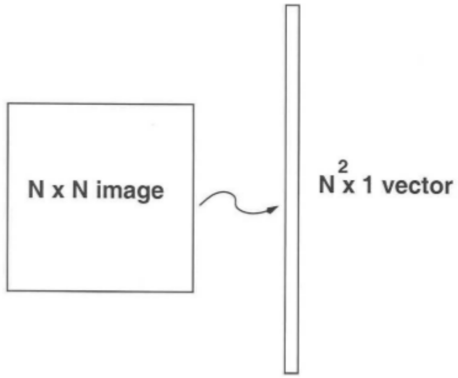


Face images

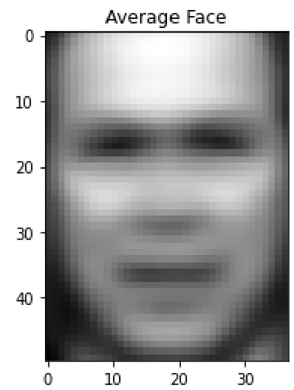
Handwritten digits

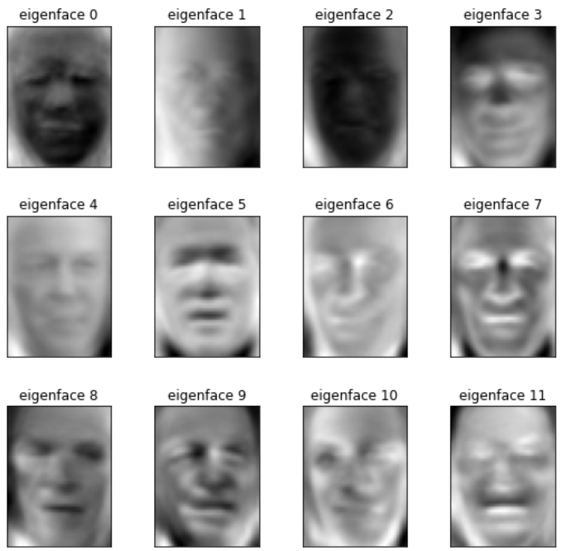
Eigenfaces

Images are converted into vectors:



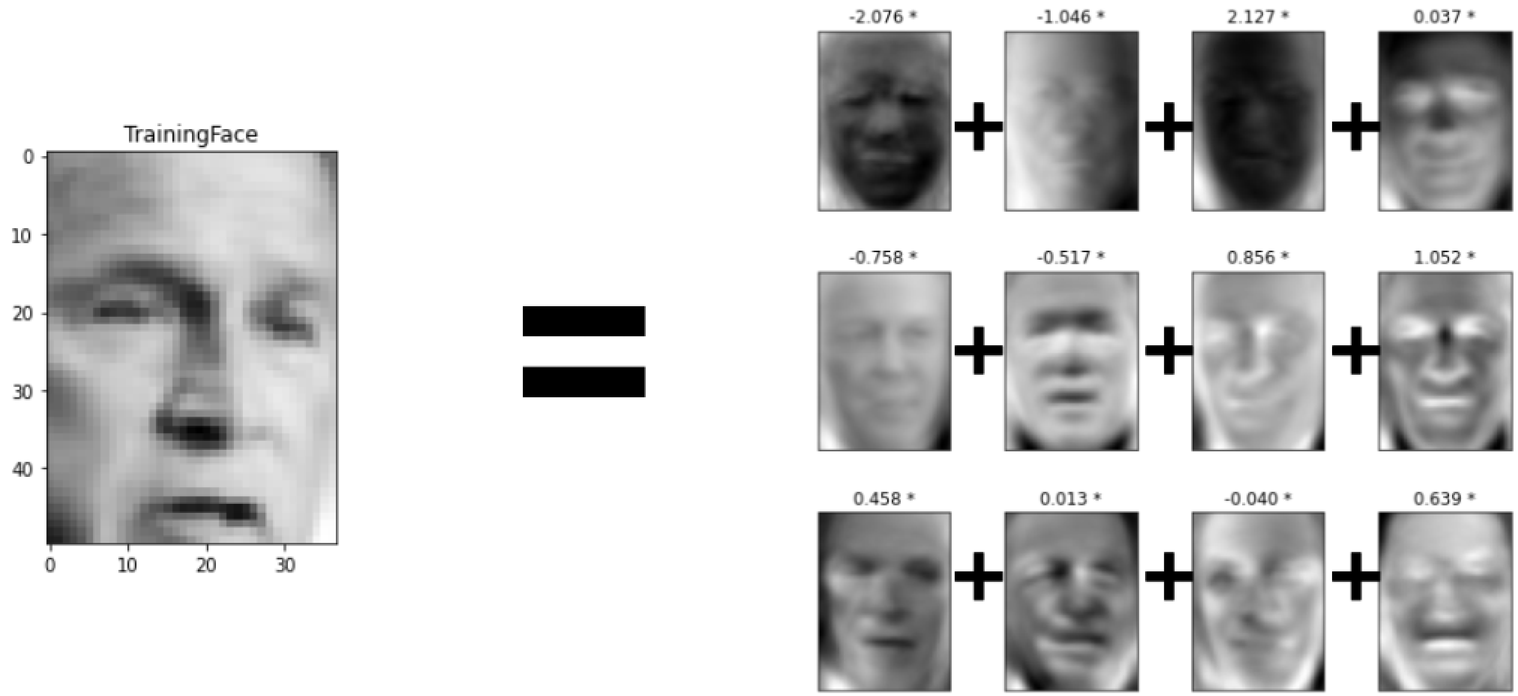
Then all training images are user to build the average face and the covariance matrix, whose eigenvectors are called eigenfaces.





Eigenfaces

Each new face can then be assumed as a weighted sum of the eigenfaces.



The weights of each eigenface represent a possible signature of a face for face-recognition tasks.

PCA for image compression

p=1

p=2

p=4

p=8

p=16

p=32

p=64

p=100

Original Image