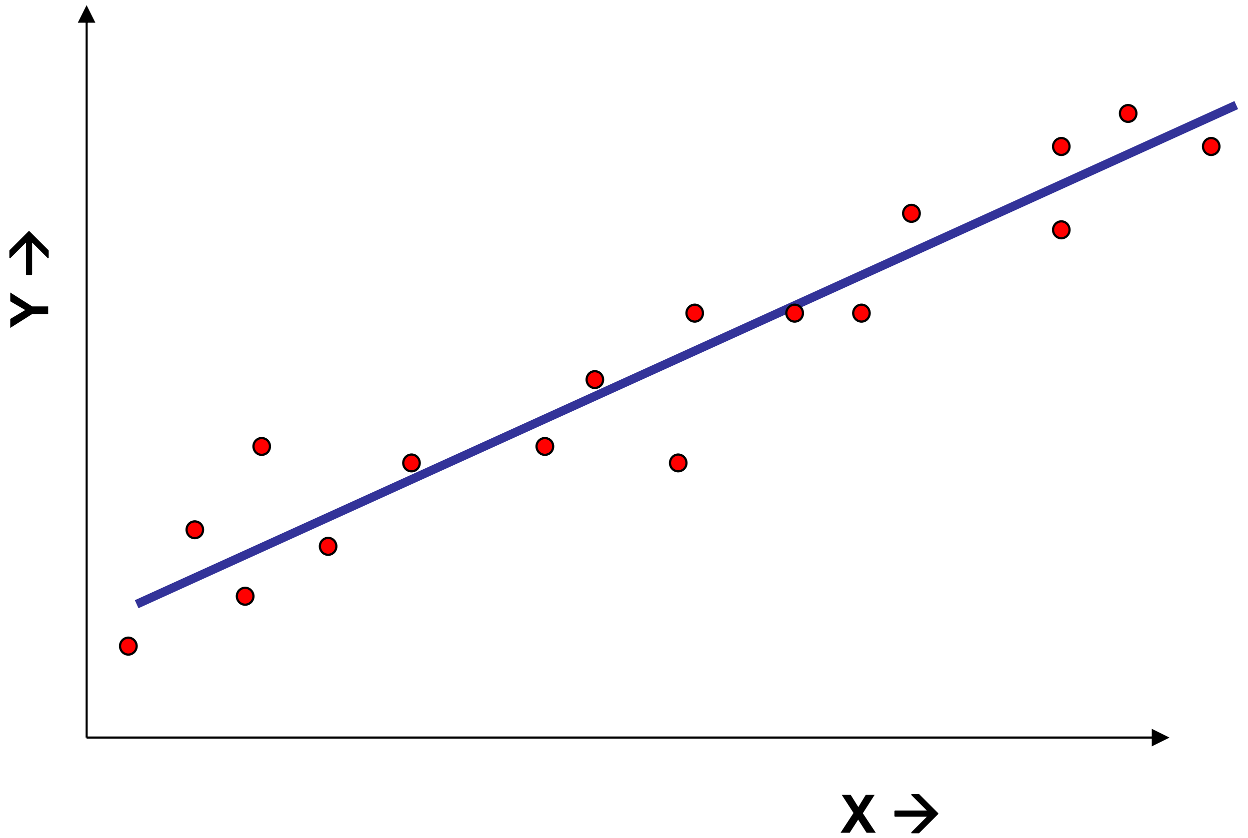
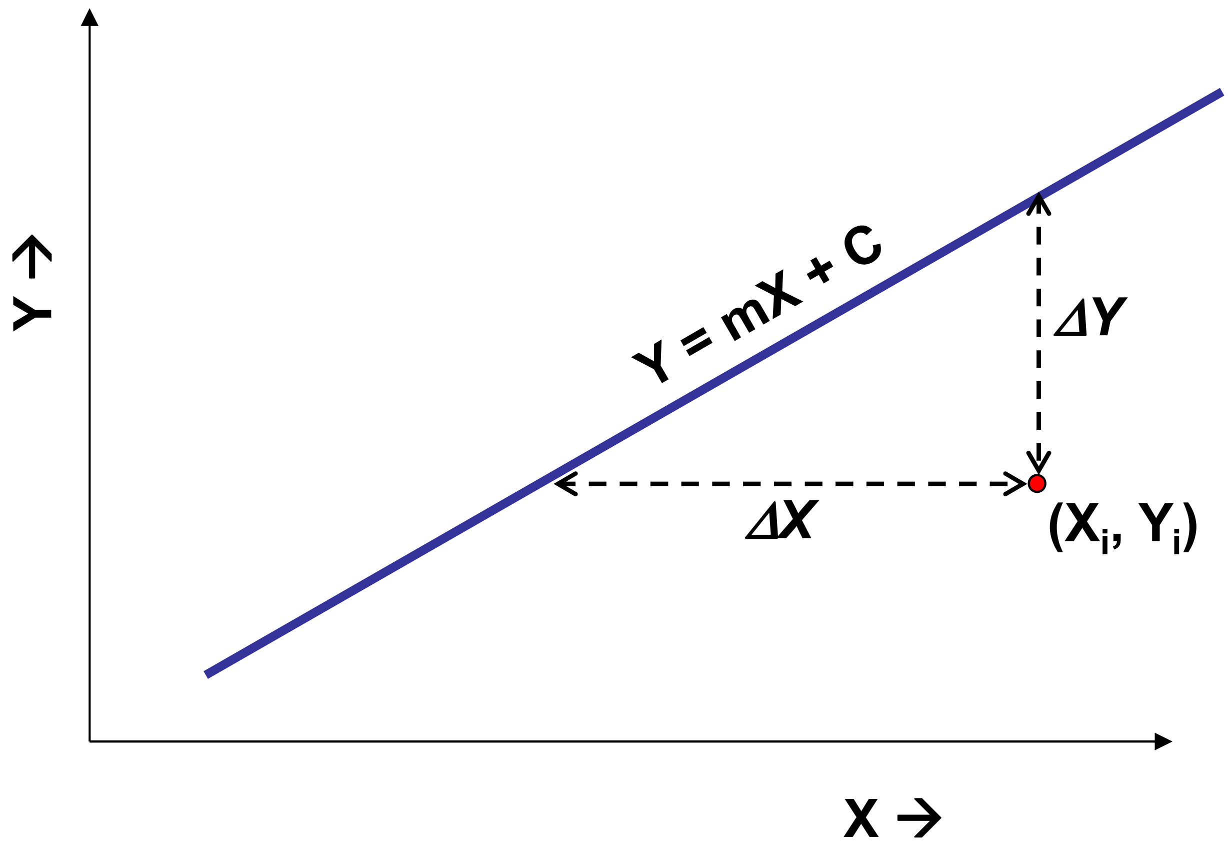
Linear Least Square Regression of a line



A simple and trivial looking problem, but a good illustration



Assume X to be the independent variable with no errors.

Errors are only with Y.

Reduce deviations, d = ΔY.

METHOD - I

Cost Function: $$E = \sum_{i=1}^{N} d_i^2 = \sum_{i=1}^{N} (Y_i - \tilde{Y}_i)^2$$

$$= \sum_{i=1}^{N} (Y_i - mX_i - C)^2$$

Minimize using derivatives:

$$\frac{\partial E}{\partial m} = -2\sum_{i=1}^{N} (Y_i - mX_i - C)(-X_i) = 0;$$

$$\frac{\partial E}{\partial C} = -\sum_{i=1}^{N} 2(Y_i - mX_i - C) = 0;$$

Re-arranging, we get Normal Equations:

$$NC + (\sum_{i=1}^{N} X_i)m = \sum_{i=1}^{N} Y_i;$$

$$(\sum_{i=1}^{N} X_i)C + (\sum_{i=1}^{N} X_i^2)m = \sum_{i=1}^{N} (X_i Y_i);$$

Solving, we get:

$$m = \frac{N\sum_{i=1}^{N}(X_iY_i) - \sum_{i=1}^{N}X_i\sum_{i=1}^{N}Y_i}{DEN};$$

$$C = \frac{\sum_{i=1}^{N}Y_i\sum_{i=1}^{N}X_i^2 - \sum_{i=1}^{N}X_i\sum_{i=1}^{N}(X_iY_i)}{DEN};$$

where,

$$DEN = N\sum_{i=1}^{N}X_i^2 - (\sum_{i=1}^{N}X_i)^2$$

In parametric form:

$$m = \frac{N^2\sigma_{XY} - N^2\mu_X\mu_Y}{N^2\sigma_X^2 - N^2\mu_X^2} = \frac{\sigma_{XY} - \mu_X\mu_Y}{\sigma_X^2 - \mu_X^2}$$

In parametric form:

$$m = \frac{N^2\sigma_{XY} - N^2\mu_X\mu_Y}{N^2\sigma_X^2 - N^2\mu_X^2} = \frac{\sigma_{XY} - \mu_X\mu_Y}{\sigma_X^2 - \mu_X^2}$$

$$C = \frac{\sum_{i=1}^{N} Y_i \sum_{i=1}^{N} X_i^2 - \sum_{i=1}^{N} X_i \sum_{i=1}^{N} (X_i Y_i)}{N\sum_{i=1}^{N} X_i^2 - (\sum_{i=1}^{N} X_i)^2};$$

$$= \frac{N^2\mu_Y\sigma_X^2 - N^2\mu_X\sigma_{XY}}{N^2\sigma_X^2 - N^2\mu_X^2} = \frac{\mu_Y\sigma_X^2 - \mu_X\sigma_{XY}}{\sigma_X^2 - \mu_X^2}$$

Check from above that the LSQ-line passes through the point: (μ_X, μ_Y). Thus shift the origin to the point: (μ_X, μ_Y).

The equation of the line in the transformed space:

$$m' = \sigma_{XY} / \sigma_X^2 ; C' = 0.$$

METHOD - II

Solving the same, using matrix concepts:

$$Y = mX + C \quad \Rightarrow \mathrm{mX} + \mathrm{C} = \mathrm{Y};$$

$$\begin{bmatrix} \mathrm{X} & 1 \end{bmatrix} \begin{bmatrix} m \\ C \end{bmatrix} = Y;$$

Any two points on the line, can give us the parameters:

$$\begin{bmatrix} \mathrm{X}_1 & 1 \end{bmatrix} \begin{bmatrix} m \\ C \end{bmatrix} = Y_1; \quad \begin{bmatrix} \mathrm{X}_2 & 1 \end{bmatrix} \begin{bmatrix} m \\ C \end{bmatrix} = Y_2;$$

$$\begin{bmatrix} \mathrm{X}_1 & 1 \\ \mathrm{X}_2 & 1 \end{bmatrix} \begin{bmatrix} m \\ C \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix};$$

Thus:

$$\begin{bmatrix} m \\ C \end{bmatrix} = \begin{bmatrix} X_1 & 1 \\ X_2 & 1 \end{bmatrix}^{-1} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \left(\frac{1}{X_1 - X_2}\right) \begin{bmatrix} 1 & -1 \\ -X_2 & X_1 \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$$

If you use this:

$$\begin{bmatrix} X_1 & 1 \end{bmatrix}\begin{bmatrix} m \\ C \end{bmatrix} = Y_1; \quad \begin{bmatrix} X_2 & 1 \end{bmatrix}\begin{bmatrix} m \\ C \end{bmatrix} = Y_2;$$

$$\begin{bmatrix} X_1 & 1 \\ X_2 & 1 \end{bmatrix}\begin{bmatrix} m \\ C \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix};$$

$^{N}C_2$ lines may be obtained for each pair.

In case of best fit:

$$\begin{bmatrix} X_1 & 1 \\ X_2 & 1 \\ & \\ & \\ X_N & 1 \end{bmatrix}\begin{bmatrix} m \\ C \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ \\ \\ Y_N \end{bmatrix};$$

We are basically trying to solve an ill-posed problem, where:

$$\begin{bmatrix} X_1 & 1 \\ X_2 & 1 \\ & \\ X_N & 1 \end{bmatrix} \begin{bmatrix} m \\ C \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ \\ Y_N \end{bmatrix};$$

Take this as: AX = B,
where A is a non-square
(or even singular square) matrix.

Use Pseudo-inverse in this case:

$$AX = B \Rightarrow A^T AX = A^T B;$$

$$(A^T A)X = A^T B \quad \Rightarrow \quad X = (A^T A)^{-1} A^T B;$$

$$X = A^+ B;$$

$$where, \quad A^+ = (A^T A)^{-1} A^T; \text{ is the Pseudo-inverse.}$$

(A^TA) is square and assumed to be non-singular (generally).
If not, look for alternative formula (hang on, for now)

A^+A or AA^+ is not equal to I (except non-singular square A), but I_p.

$$(A^T A)X = A^T B$$

$$\begin{bmatrix} X_1 & 1 \\ X_2 & 1 \\ & \\ X_N & 1 \end{bmatrix} \begin{bmatrix} m \\ C \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ \\ Y_N \end{bmatrix};$$

$$\Rightarrow \quad \beta = (A^T A)^{-1} A^T B;$$

$$\beta = A^+ B;$$

$$where, \quad A^+ = (A^T A)^{-1} A^T; \text{ is the Pseudo-inverse.}$$

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

where $\hat{y}_i = \hat{f}(x_i)$. The matrix $\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ appearing in equation (3.7) is sometimes called the "hat" matrix because it puts the hat on $\mathbf{y}$.

Altn. Form of Pseudo-inverse:

$$AX = B$$

$$\Rightarrow AX = (AA^T).(AA^T)^{-1} B;$$

$$thus, \quad A^+ = A^T (AA^T)^{-1}$$

$$A = \begin{bmatrix} X_1 & 1 \\ X_2 & 1 \\ \vdots & \vdots \\ X_N & 1 \end{bmatrix};$$

$$AX = B \Rightarrow A^T AX = A^T B;$$

$$X = (A^T A)^{-1} A^T B;$$

$$B = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix};$$

$$A^T A = \begin{bmatrix} X_1 & X_2 & \cdots & X_N \\ 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} X_1 & 1 \\ X_2 & 1 \\ \vdots & \vdots \\ X_N & 1 \end{bmatrix};$$

$$A^T A = \begin{bmatrix} \sum_{i=1}^{N} X_i^2 & \sum_{i=1}^{N} X_i \\ \sum_{i=1}^{N} X_i & N \end{bmatrix} = N \begin{bmatrix} \sigma_x^2 & \mu_x \\ \mu_x & 1 \end{bmatrix}$$

X =[m C]ᵀ

$$A^T A = N\begin{bmatrix} \sigma_x^2 & \mu_x \\ \mu_x & 1 \end{bmatrix}; \ (A^T A)^{-1} = (\frac{1}{N.D})\begin{bmatrix} 1 & -\mu_x \\ -\mu_x & \sigma_x^2 \end{bmatrix}$$

where, D= $\sigma_x^2 - \mu_x^2$

$$A^T B = \begin{bmatrix} X_1 & X_2 & & X_N \\ 1 & 1 & & 1 \end{bmatrix}\begin{bmatrix} Y_1 \\ Y_2 \\ \\ \\ Y_N \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N} X_i Y_i \\ \sum_{i=1}^{N} Y_i \end{bmatrix} = N\begin{bmatrix} \sigma_{XY} \\ \mu_Y \end{bmatrix};$$

Thus:

$$X = A^+ B = (A^T A)^{-1} A^T B = \left(1/D\right)\begin{bmatrix} 1 & -\mu_X \\ -\mu_X & \sigma_x^2 \end{bmatrix}\begin{bmatrix} \sigma_{XY} \\ \mu_Y \end{bmatrix}$$

$$= \left(1/D\right)\begin{bmatrix} \sigma_{XY} - \mu_X \mu_Y \\ \sigma_x^2 \mu_Y - \sigma_{XY} \mu_X \end{bmatrix}$$

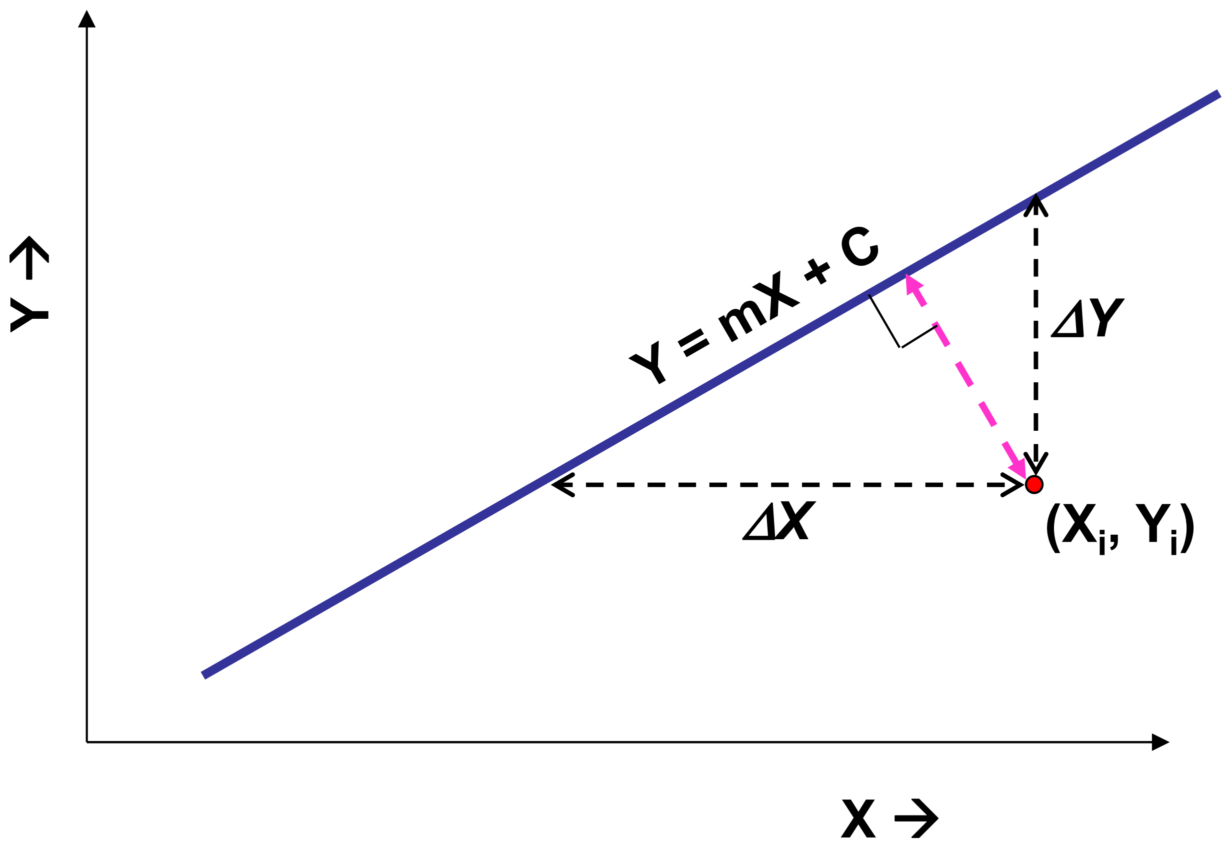
The solution is same as in LSQ-FIT.

Pseudo-inverse satisfies the Least-square criteria.

So we have seen the relation between LSQ and Pseudo-inverse.

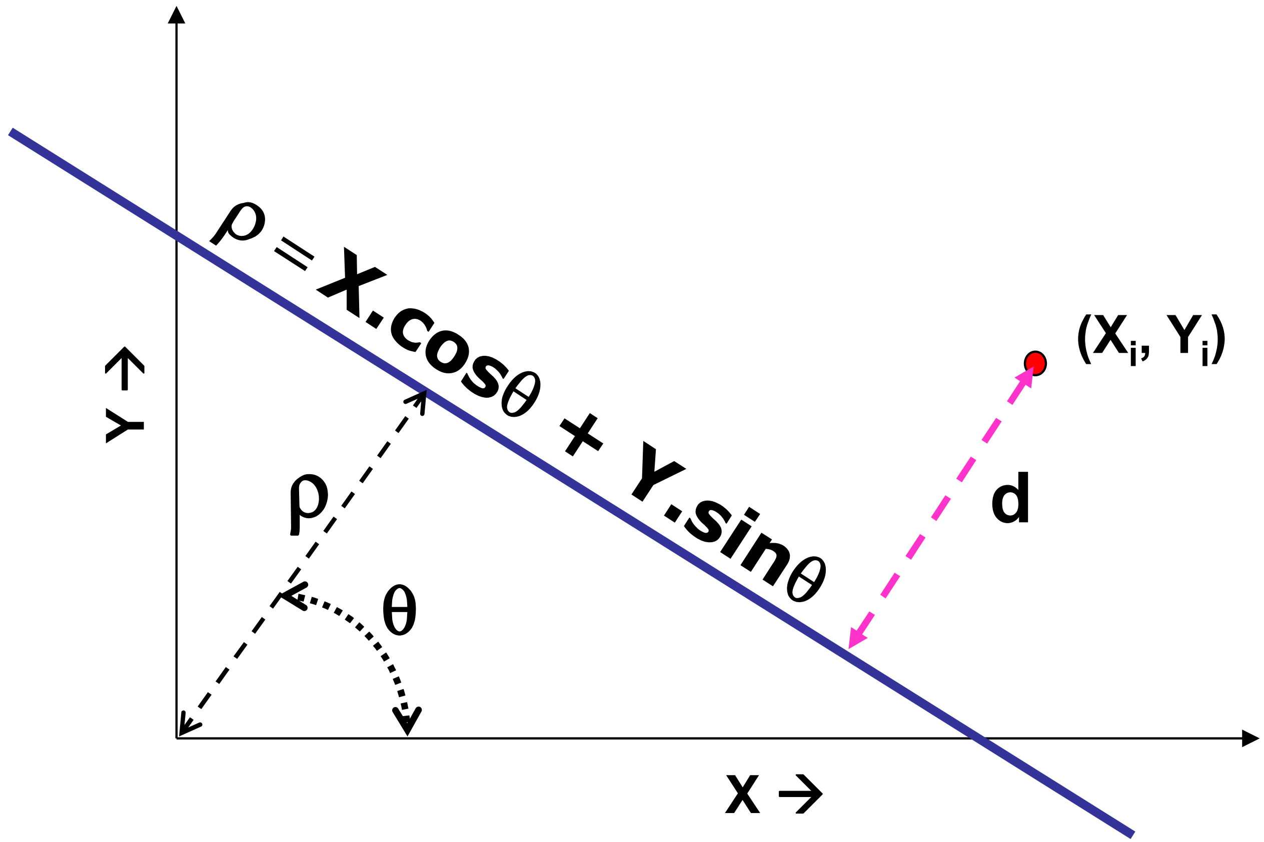
Where and how does the eigen-space (PCA) help us?

What if, we want to minimize sum of perpendicular distances – errors are in both X and Y coordinates



METHOD - III

Change to Polar coordinate representation



$$d = \rho - X_i.\cos(\theta) - Y_i.\sin(\theta)$$

New cost Function:

$$E = \sum_{i=1}^{N} d_i^2 = \sum_{i=1}^{N} (\rho - X_i \cos\theta - Y_i \sin\theta)^2$$

Minimize using derivatives:

$$\partial E / \partial\theta = \sum_{i=1}^{N} (X_i \cos\theta + Y_i \sin\theta - \rho)(Y_i \cos\theta - X_i \sin\theta) = 0;$$

$$\partial E / \partial\rho = \sum_{i=1}^{N} (X_i \cos\theta + Y_i \sin\theta - \rho) = 0.$$

Solve, to get:

$$\rho = \frac{(\sum_{i=1}^{N} X_i)\cos\theta + (\sum_{i=1}^{N} Y_i)\sin\theta}{N}$$

$$\tan(2\theta) = \frac{(\sum_{i=1}^{N} X_i)(\sum_{i=1}^{N} Y_i) - N(\sum_{i=1}^{N} X_i Y_i)}{N(\sum_{i=1}^{N} Y_i^2 - \sum_{i=1}^{N} X_i^2) + (\sum_{i=1}^{N} X_i)^2 - (\sum_{i=1}^{N} Y_i)^2}$$

$$\rho = \mu_x \cos\theta + \mu_y \sin\theta$$

$$tan\, 2\theta = \frac{2\, cov(X, Y)}{[var(X) - var(Y)]}$$

$$\theta = \frac{1}{2} tan^{-1}\left(\frac{2\, cov(X, Y)}{[var(X) - var(Y)]}\right)$$

$$\tan(2\theta) = \frac{(\sum_{i=1}^{N} X_i)(\sum_{i=1}^{N} Y_i) - N(\sum_{i=1}^{N} X_i Y_i)}{N(\sum_{i=1}^{N} Y_i^2 - \sum_{i=1}^{N} X_i^2) + (\sum_{i=1}^{N} X_i)^2 - (\sum_{i=1}^{N} Y_i)^2} = Num/Den$$

Any Problem in the above expression above ?

- $\theta = \pi/4$
- **Exact value (quadrant and sign) of θ**

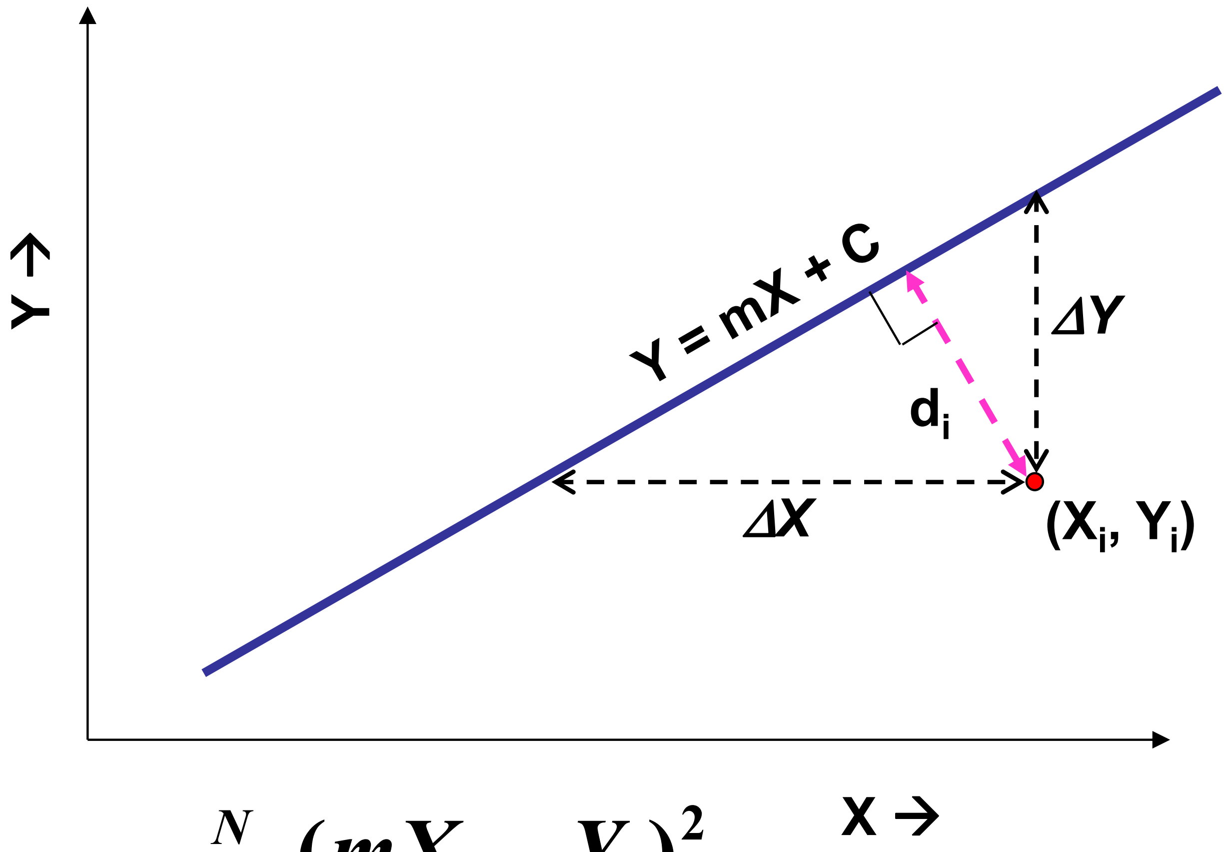
Use:

$$\theta = (1/2)\sin^{-1}\left[\frac{Num}{sqrt(Num^2 + Den^2)}\right]$$

Also use the sign of cos(2θ) ansd sin(2θ) to obtain the quadrant of 2θ.

METHOD - IV

Minimize perpendicular distances in Cartesian Coordinate form:



Derive:

$$d_i\Big|_{C=0} = \frac{m\left(X_i - Y_i/m\right)}{\sqrt{1+m^2}}$$

Cost Function:

$$E = \sum_{i=1}^{N} d_i = \sum_{i=1}^{N} \frac{(mX_i - Y_i)^2}{1+m^2}$$

Solution for m, using:

$$\partial E/\partial m = 0; \quad \sum_{i=1}^{N} [mX_i^2 + (m^2-1)X_iY_i - mY_i^2] = 0;$$

Solution for slope m: $$\sum_{i=1}^{N}[mX_i^2+(m^2-1)X_iY_i-mY_i^2]=0;$$

Considering $\mu_X = \mu_Y = 0;$

$$m\sigma_X^2+(m^2-1)\sigma_{XY}-m\sigma_Y^2=0$$

$$\equiv m^2\sigma_{XY}+m(\sigma_X^2-\sigma_Y^2)-\sigma_{XY}=0$$

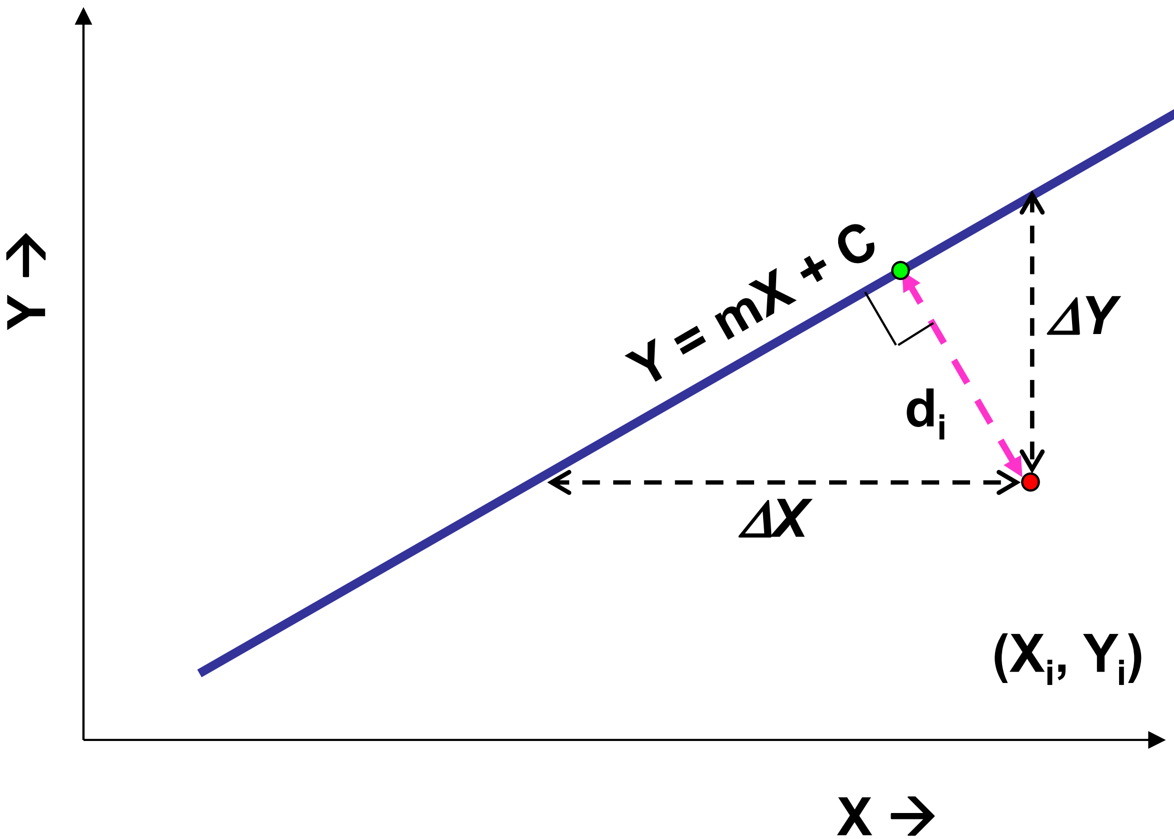
$$m=\frac{\sigma_Y^2-\sigma_X^2}{2\sigma_{XY}}\pm\sqrt{[\frac{\sigma_Y^2-\sigma_X^2}{2\sigma_{XY}}]^2+1}$$

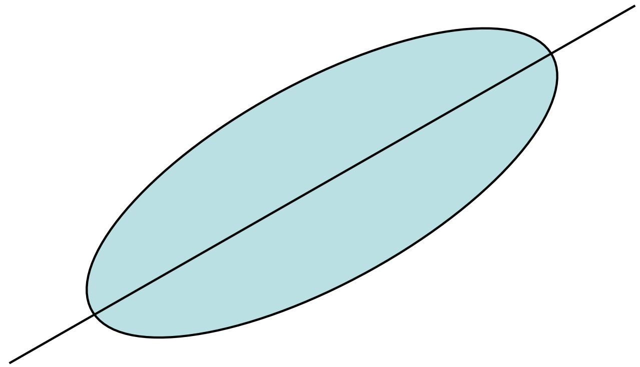
Note down this expression given above.

METHOD – V

PCA-based

Derivation using PCA or eigen-analysis





We are looking for a direction along which we have the maximum variance of the projections.

The co-variance matrix of the data set (2-D) is:

$$S = \begin{bmatrix} \sigma_X^2 & \sigma_{XY} \\ \sigma_{XY} & \sigma_Y^2 \end{bmatrix}$$

Let us compute the eigenvalues of the co-variance matrix, S:

$$S - \lambda I = 0 \qquad |S - \lambda I| = \begin{bmatrix} \sigma_X^2 - \lambda & \sigma_{XY} \\ \sigma_{XY} & \sigma_Y^2 - \lambda \end{bmatrix}$$

$$(\lambda - \sigma_X^2)(\lambda - \sigma_Y^2) - \sigma_{XY}^2 = 0$$

$$\Rightarrow \lambda^2 - \lambda(\sigma_X^2 + \sigma_Y^2) + [(\sigma_X \sigma_Y)^2 - \sigma_{XY}{}^2] = 0$$

$$\lambda = \frac{\sigma_X^2 + \sigma_Y^2 \pm \sqrt{(\sigma_X^2 + \sigma_Y^2)^2 - 4(\sigma_X \sigma_Y)^2 + 4\sigma_{XY}^2}}{2}$$

$$= \left(\frac{\sigma_X^2 + \sigma_Y^2}{2}\right) \pm \sqrt{\left(\frac{\sigma_X^2 - \sigma_Y^2}{2}\right)^2 + \sigma_{XY}^2}$$

$$\lambda = \left(\frac{\sigma_X^2 + \sigma_Y^2}{2}\right) \pm \sqrt{\left(\frac{\sigma_X^2 - \sigma_Y^2}{2}\right)^2 + \sigma_{XY}^2}$$

Eigenvectors (*v*) are those, for which S*V* = λ*V*:

$$\begin{bmatrix} \sigma_X^2 & \sigma_{XY} \\ \sigma_{XY} & \sigma_Y^2 \end{bmatrix} \begin{bmatrix} V_X \\ V_Y \end{bmatrix} = \lambda \begin{bmatrix} V_X \\ V_Y \end{bmatrix}$$

Which gives us two equations:

$$(\lambda - \sigma_X^2) V_X = \sigma_{XY} V_Y; \qquad \sigma_{XY} V_X = (\lambda - \sigma_Y^2) V_Y$$

Direction of the vector:

$$V_Y / V_X = \frac{\lambda - \sigma_X^2}{\sigma_{XY}} = \frac{\sigma_{XY}}{\lambda - \sigma_Y^2}$$

$$V_Y / V_X = \left(\frac{1}{\sigma_{XY}}\right) \left[\left(\frac{\sigma_Y^2 - \sigma_X^2}{2}\right) \pm \sqrt{\left(\frac{\sigma_X^2 - \sigma_Y^2}{2}\right)^2 + \sigma_{XY}^2} \right]$$

Thus, we have got so far:

$$V_Y / V_X = \frac{\lambda - \sigma_X^2}{\sigma_{XY}}$$

where,

$$\lambda = \left(\frac{\sigma_X^2 + \sigma_Y^2}{2}\right) \pm \sqrt{\left(\frac{\sigma_X^2 - \sigma_Y^2}{2}\right)^2 + \sigma_{XY}^2}$$

$$V_Y / V_X =$$

$$=$$

Isn't that our m (slope) obtained earlier (method 4) using the derivative of the cost function with Perp. distances?

$$m = \frac{\sigma_{XY} - \mu_X \mu_Y}{\sigma_X^2 - \mu_X^2}$$

$$C = \frac{\mu_Y \sigma_X^2 - \mu_X \sigma_{XY}}{\sigma_X^2 - \mu_X^2}$$

$$\rho = \frac{(\sum_{i=1}^{N} X_i)\cos\theta + (\sum_{i=1}^{N} Y_i)\sin\theta}{N}$$

$$\theta = \frac{1}{2} tan^{-1}\left(\frac{2\, cov(X,Y)}{[var(X) - var(Y)]}\right)$$

$$V_Y / V_X = \left(\frac{1}{\sigma_{XY}}\right)\left[\left(\frac{\sigma_Y^2 - \sigma_X^2}{2}\right) \pm \sqrt{\left(\frac{\sigma_X^2 - \sigma_Y^2}{2}\right)^2 + \sigma_{XY}^2}\right]$$

$$= \left(\frac{\sigma_Y^2 - \sigma_X^2}{2\sigma_{XY}}\right) \pm \sqrt{\left(\frac{\sigma_Y^2 - \sigma_X^2}{2\sigma_{XY}}\right)^2 + 1}$$

$$m = \frac{\sigma_Y^2 - \sigma_X^2}{2\sigma_{XY}} \pm \sqrt{\left[\frac{\sigma_Y^2 - \sigma_X^2}{2\sigma_{XY}}\right]^2 + 1}$$

But we still can't explain why and how it (PCA) works for face recognition!