Praktikum zo strojového učenia a umelej inteligencie na vizuálnych dátach

Andrej Lúčny

Katedra aplikovanej informatiky FMFI UK

lucny@fmph.uniba.sk

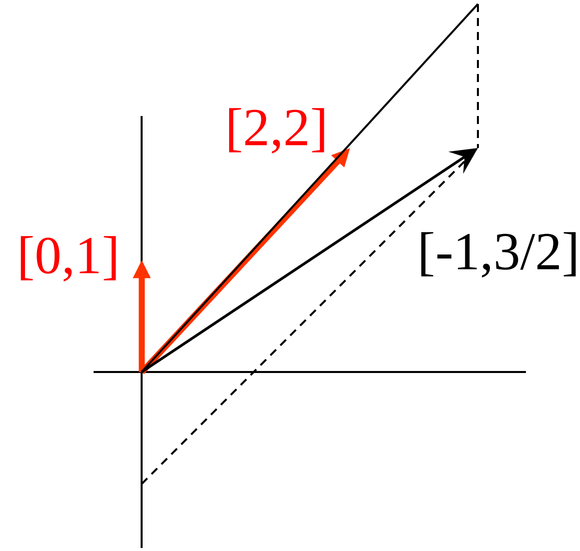
http://dai.fmph.uniba.sk/w/Andrej_Lucny

www.agentspace.org/praktikum

8

Zmena bázy

- matica prechodu:
  obrazy jednotkových vektorov
  (zobrazenie na vhodné lineárne kombinácie)

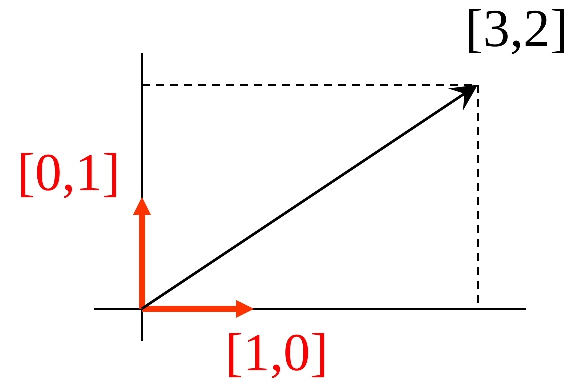


$$[x_1, x_2]\begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} = [x'_1, x'_2]$$

$$[-1, 3/2]\begin{pmatrix} 0 & 1 \\ 2 & 2 \end{pmatrix} = [3, 2]$$

znamená, že

$$x'_1 = a_{1,1}x_1 + a_{2,1}x_2$$
$$x'_2 = a_{1,2}x_1 + a_{2,2}x_2$$



$[x_1, x_2]$ v báze $[a_{11},a_{12}]$ $[a_{21},a_{22}]$ je to isté ako

$[x'_1, x'_2]$ v báze [1,0] [0,1]

Dataset

- Množina n vzoriek dimenzie m

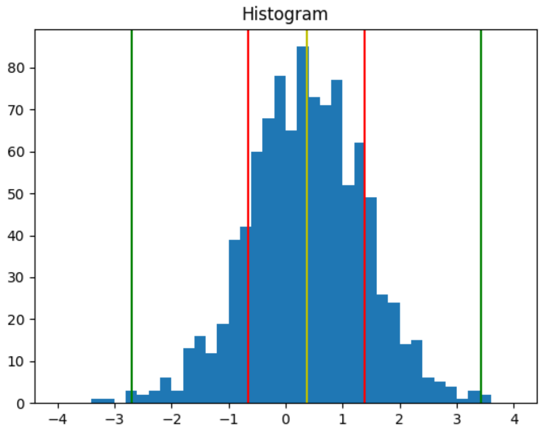
$$[a_1, \ldots, x_1, y_1 \ldots]$$
$$[a_2, \ldots, x_2, y_2 \ldots]$$
$$\ldots$$
$$[a_n, \ldots, x_n, y_n \ldots]$$

$$[x_1, x_2, \ldots, x_n]$$
$$[y_1, y_2, \ldots, y_n]$$

Ako zistiť či pozícia x súvisí s y ?

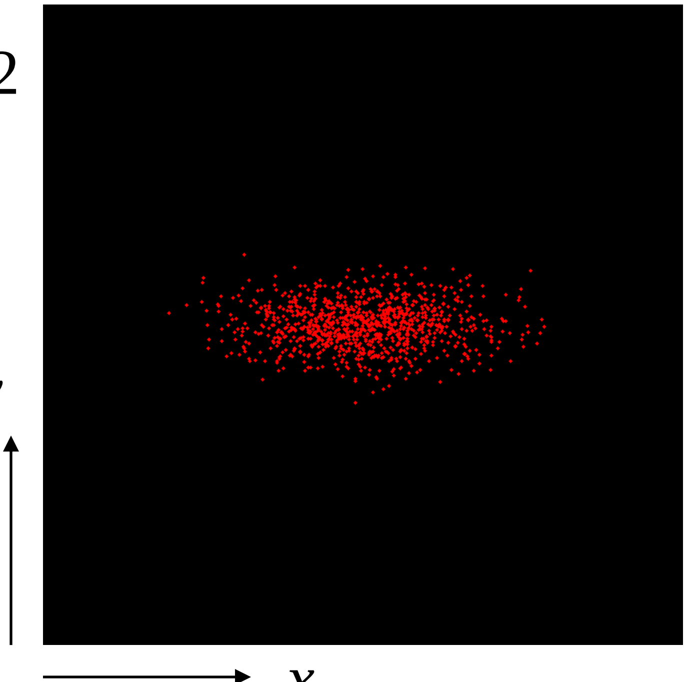
Priemer, smerodajná odchýľka

$$\overline{x} = \sum_{i=0}^{n} x_i \qquad \sigma_x = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})^2}$$

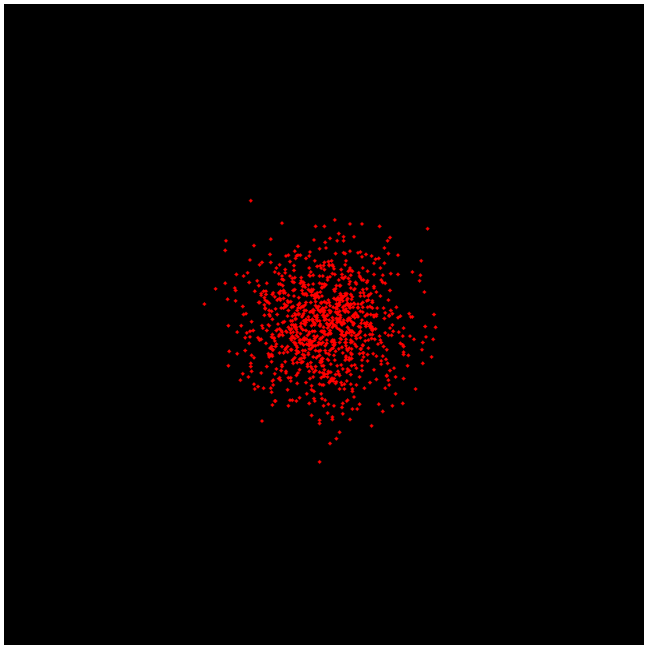


$$[x_i, y_i] \qquad [\frac{x_i - \overline{x}}{\sigma_x}, \frac{y_i - \overline{y}}{\sigma_y}]$$

m=2



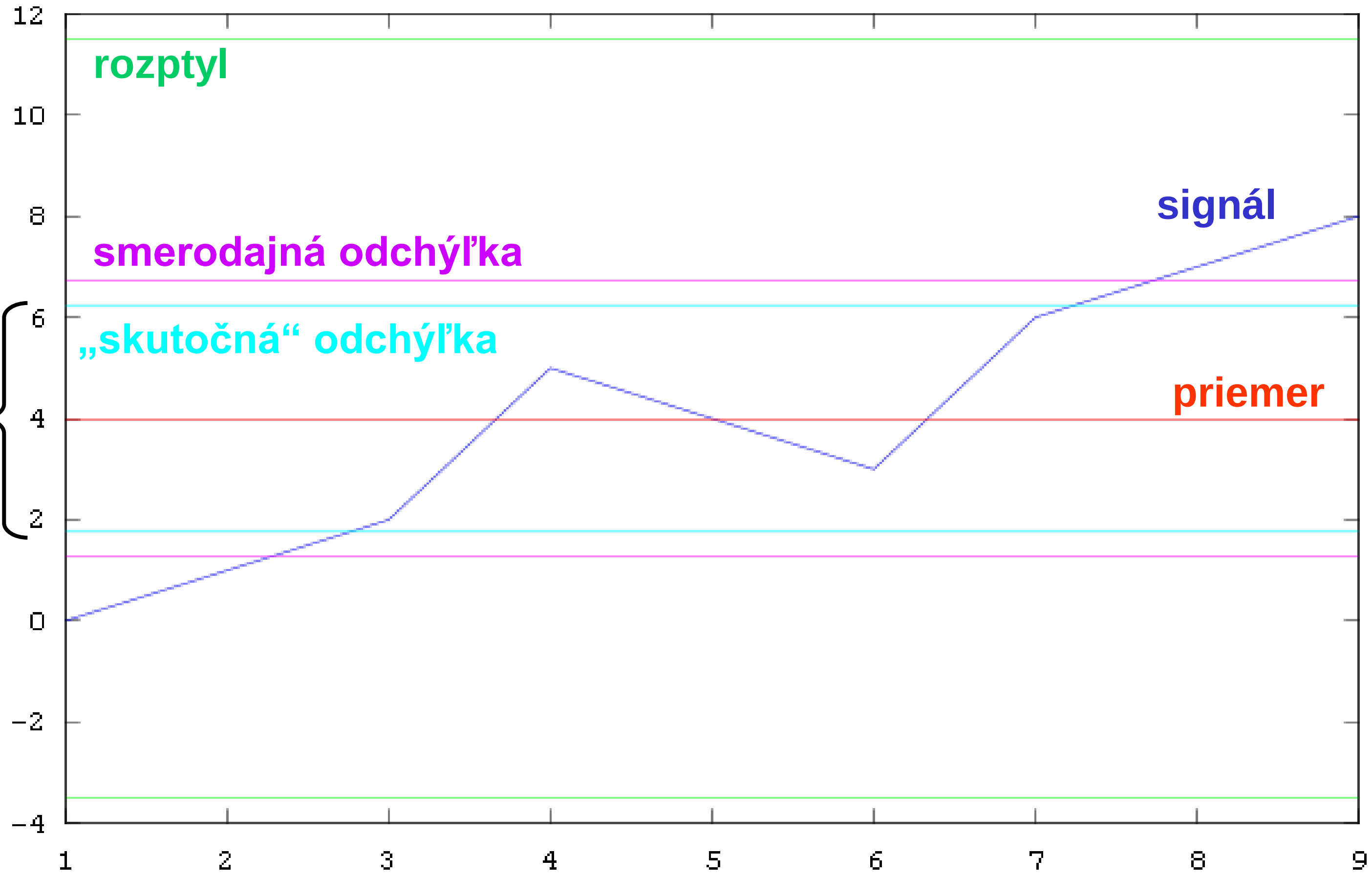
normalizácia



y

x

Odchýľka a smerodajná odchýľka



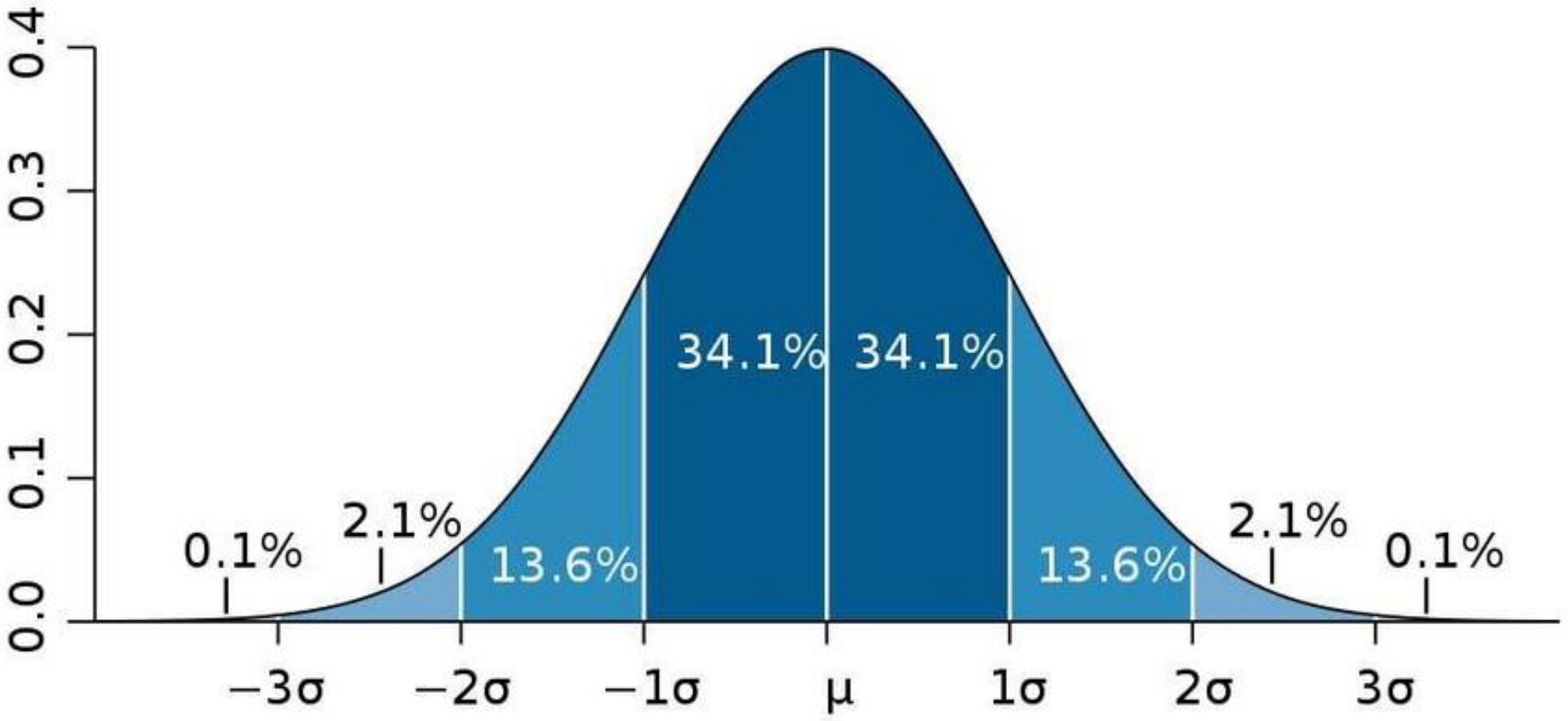
68% 50%

$$\langle \overline{x} - \sigma_x , \overline{x} + \sigma_x \rangle$$

99% $\langle \overline{x} - 3\sigma_x , \overline{x} + 3\sigma_x \rangle$ pravidlo 3σ

Gaussova krivka

$$\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



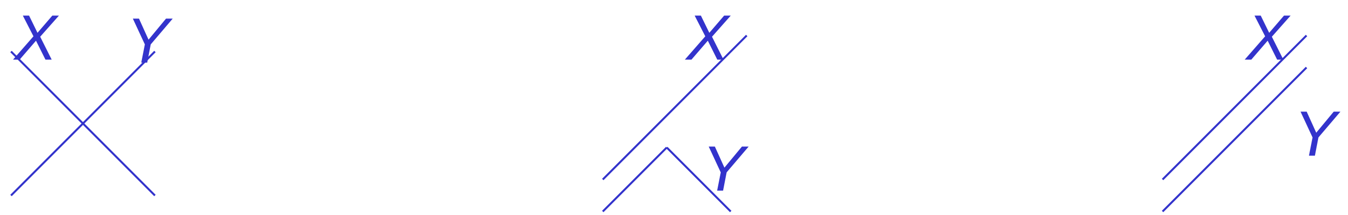
Kovariancia

$$\overline{x} = \sum_{i=0}^{n} x_i$$

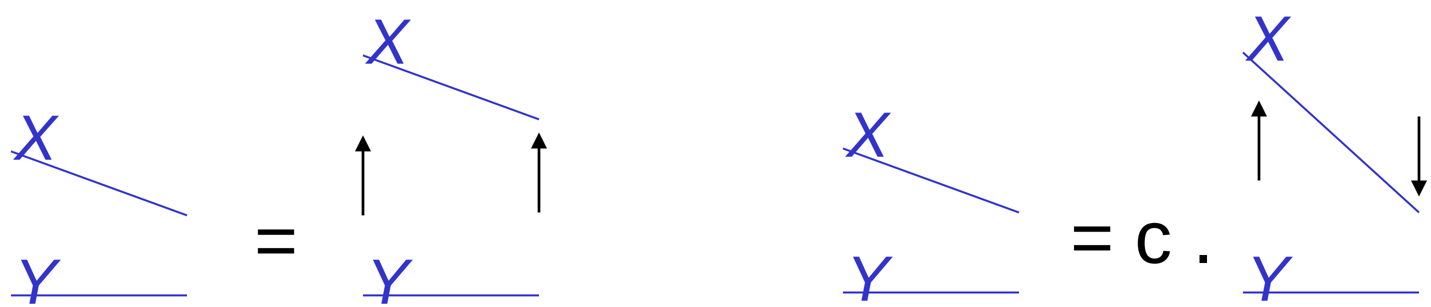
$$\overline{y} = \sum_{i=0}^{n} y_i$$

Kovariancia dvoch signálov X a Y:

$$cov(x,y) = \frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})(y_i - \overline{y})$$



záporná nulová kladná



$$cov(x,y) = \frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})(y_i - \overline{y}) =$$

$$= \frac{1}{n}\frac{\sqrt{\frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})^2}\sqrt{\frac{1}{n}\sum_{i=0}^{n}(y_i - \overline{y})^2}}{\sqrt{\frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})^2}\sqrt{\frac{1}{n}\sum_{i=0}^{n}(y_i - \overline{y})^2}}\sum_{i=0}^{n}(x_i - \overline{x})(y_i - \overline{y}) =$$

$$= \sigma_x\sigma_y\frac{\sum_{i=0}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=0}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=0}^{n}(y_i - \overline{y})^2}} =$$

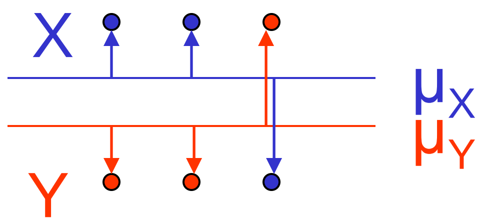
$$= \sigma_x\sigma_y\frac{(x - \overline{x})\cdot(y - \overline{y})}{|x - \overline{x}||y - \overline{y}|} =$$

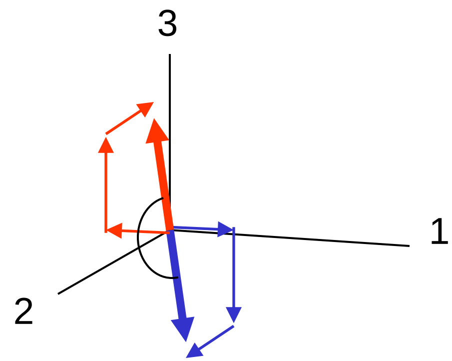
$$= \sigma_x\sigma_y\cos(\phi) = \sigma_x\sigma_y cor(x,y)$$

$$cov(a,a) = \sigma_a\sigma_a\cos(0) = \sigma_a^2 = \rho_a \qquad cov(a,b) = cov(b,a)$$

Korelácia

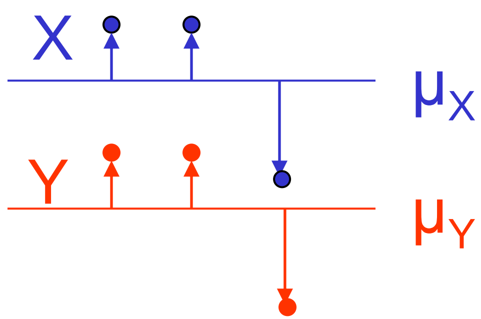
n=3

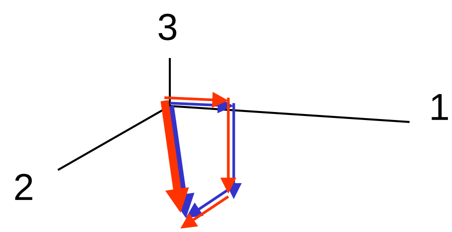




$\varphi = \pi$

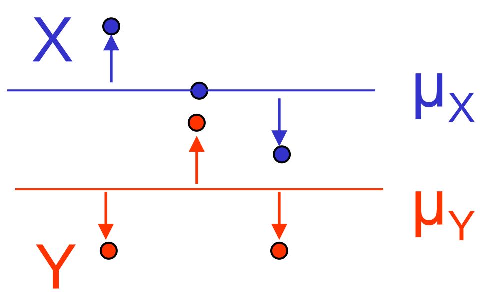
cor(X,Y) = -1

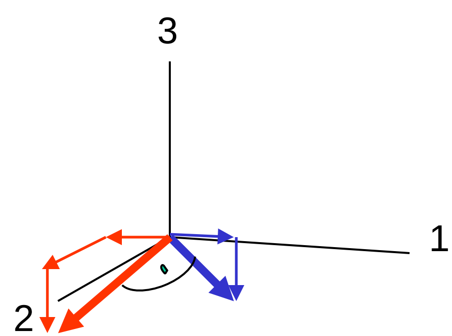




$\varphi = 0$

cor(X,Y) = 1





$\varphi = \pi/2$

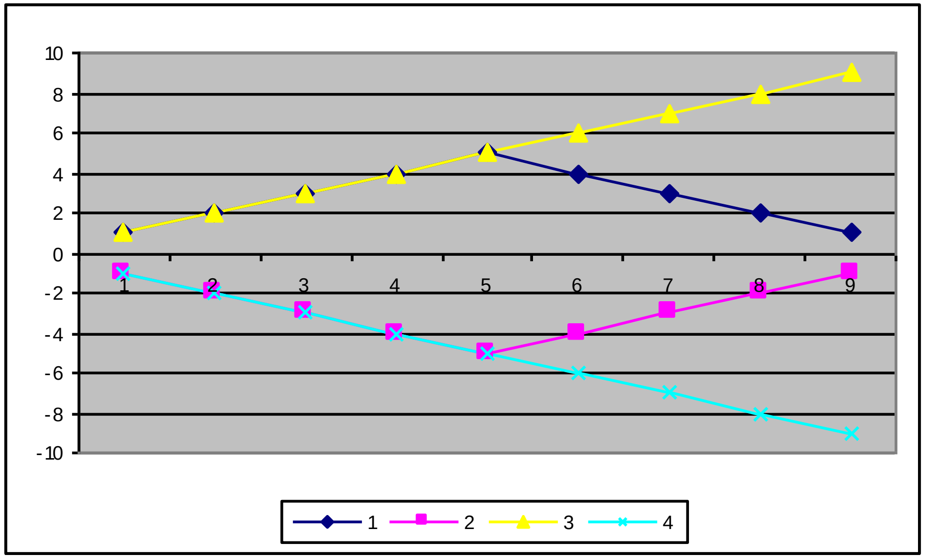
cor(X,Y) = 0

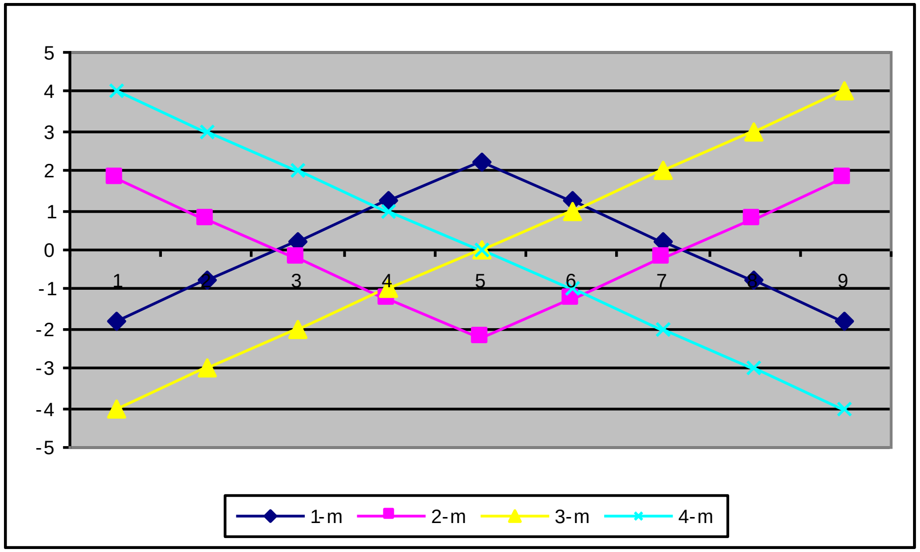
Kovariančná matica

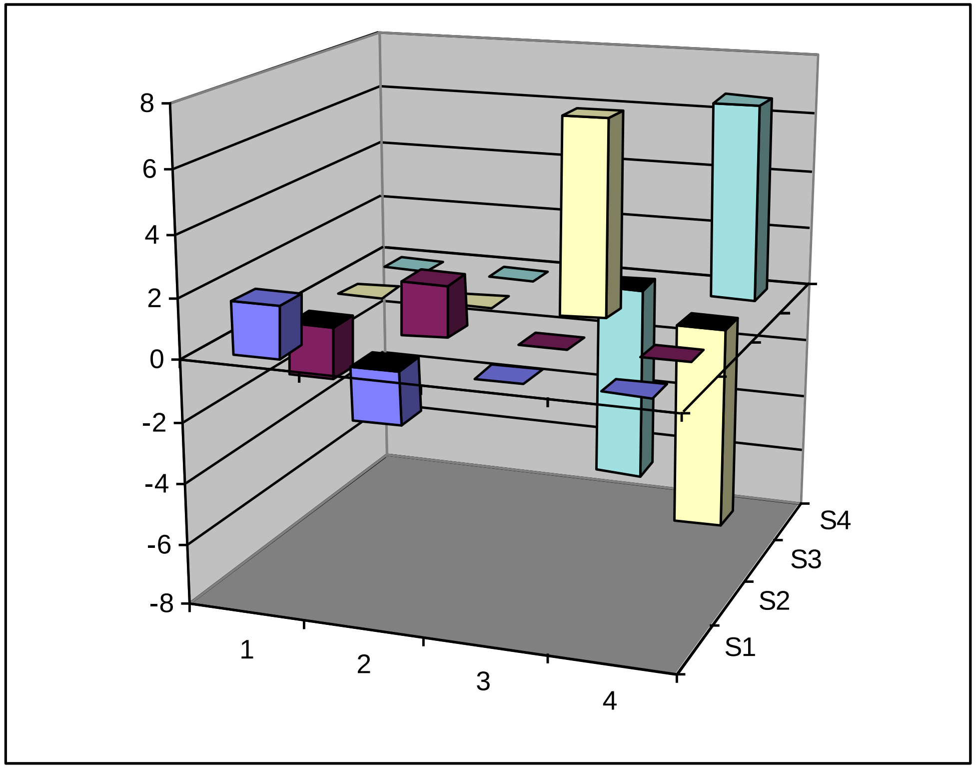
Matica rozmerov m x m udávajúca vzájomné kovariancie dimenzií datasetu

$$\begin{pmatrix} cov(a,a) & \ldots & cov(a,x) & cov(a,y) & \ldots \\ \ldots & & & & \\ cov(x,a) & \ldots & cov(x,x) & cov(x,y) & \ldots \\ cov(y,a) & \ldots & cov(y,x) & cov(y,y) & \ldots \\ \ldots & & & & \end{pmatrix}$$

Kovariančná matica





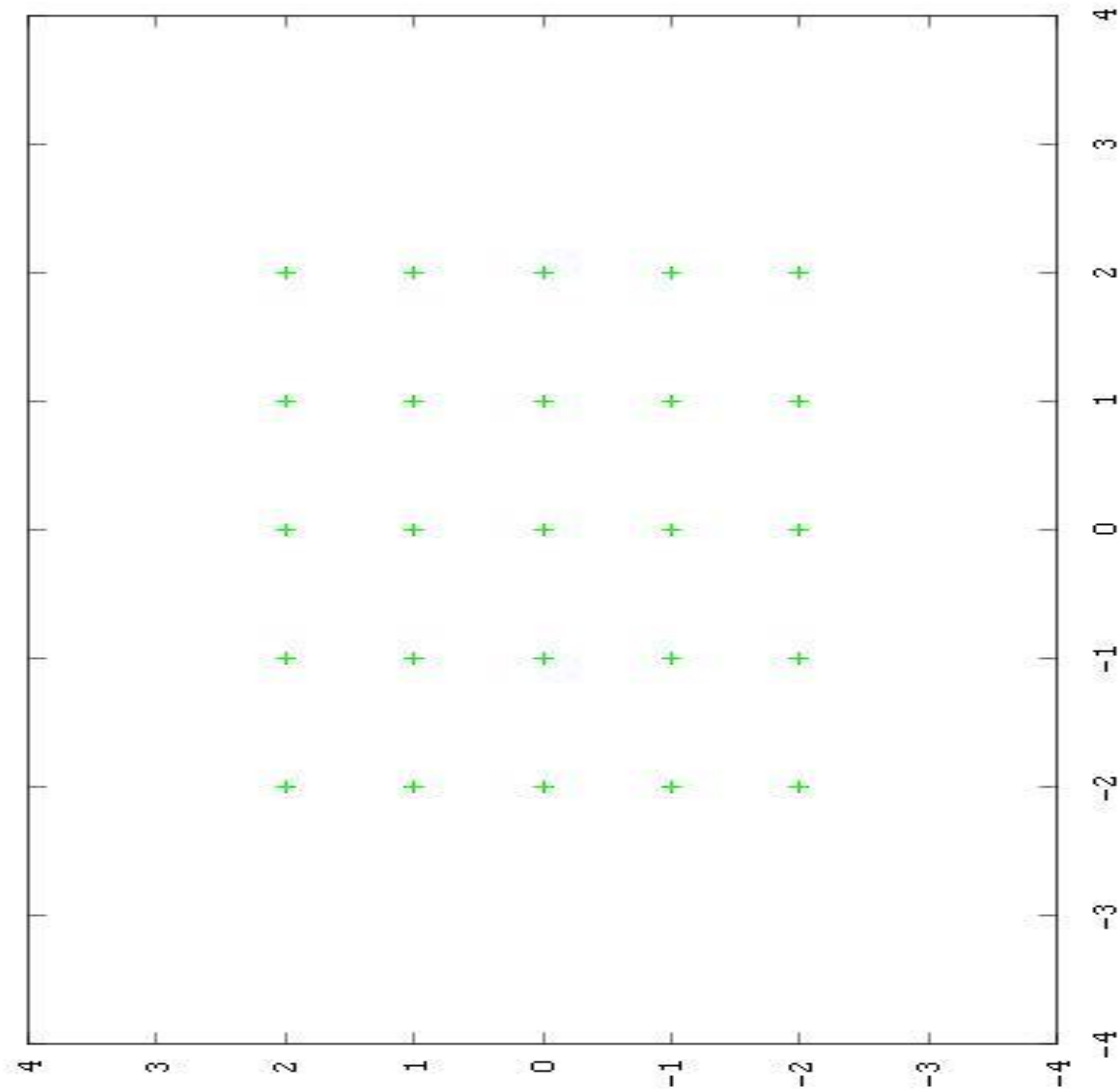


Idea PCA

- Vedeli by sme nájsť takú zmenu bázy, že kovariančná matica vyjde diagonálna?
- To je dobré na to, že jednotlivé dimenzie sú od seba nezávislé

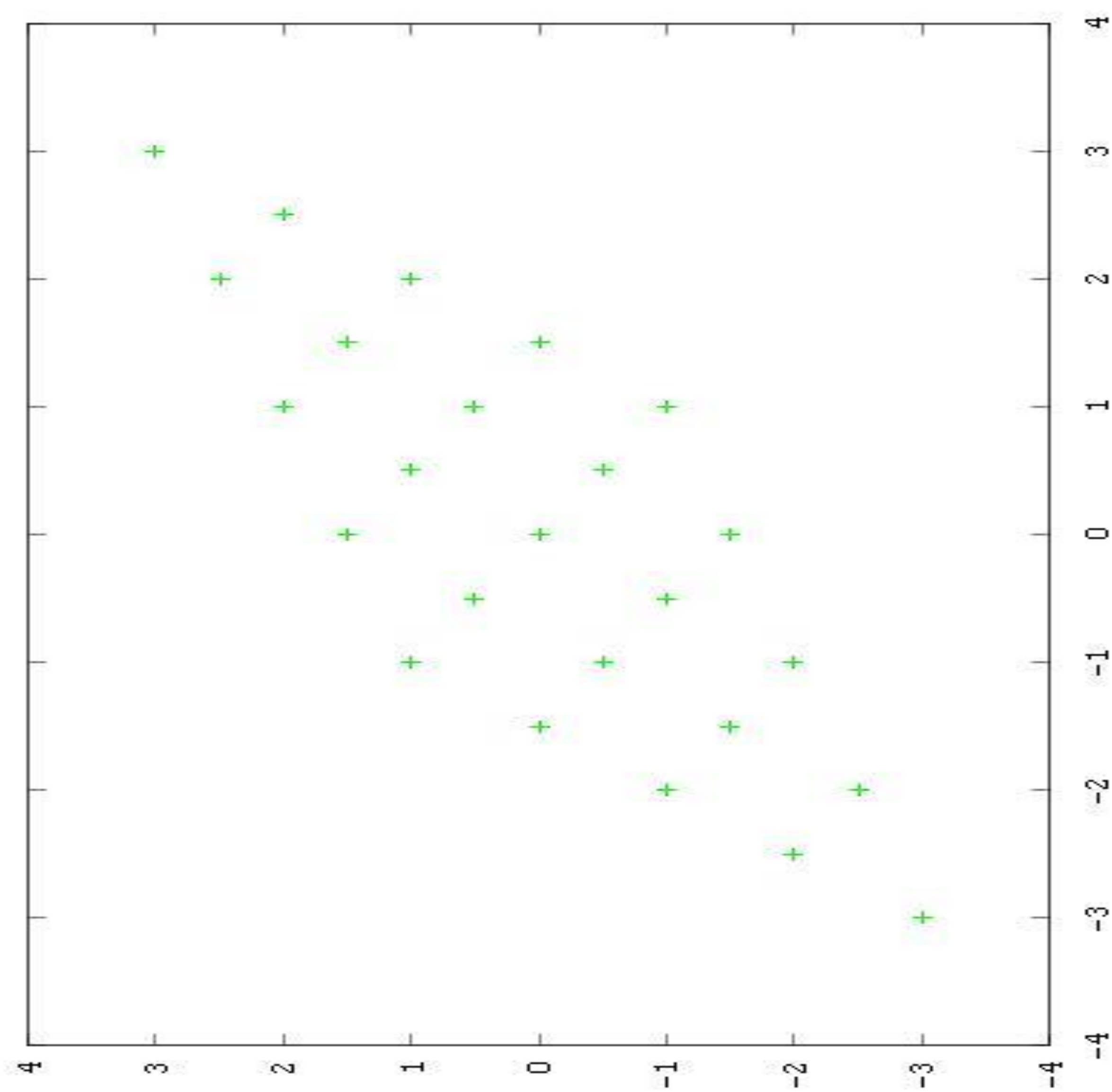
- Odpoveď: vedeli, takú zmenu bázy zabezpečí matica prechodu po stĺpcoch zložená z normalizovaných vlastných vektorov. Hodnoty na diagonále potom budú vlastnými hodnotami kovariančnej matice a zodpovedajú rozptylu

Vlastné vektory a hodnoty



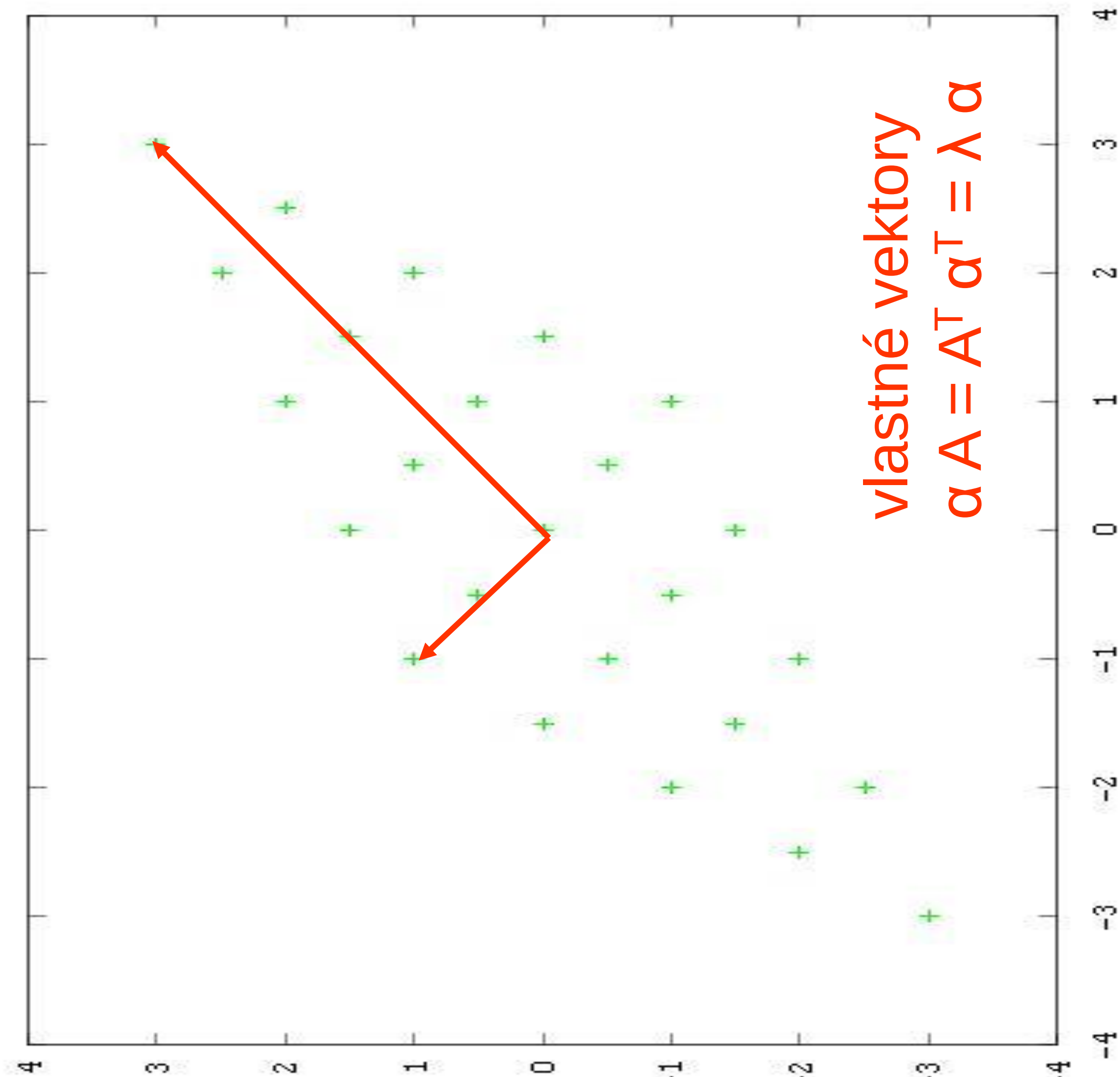
- Transformujme tieto body pomocou lineárnej transformácie

Vlastné vektory a hodnoty

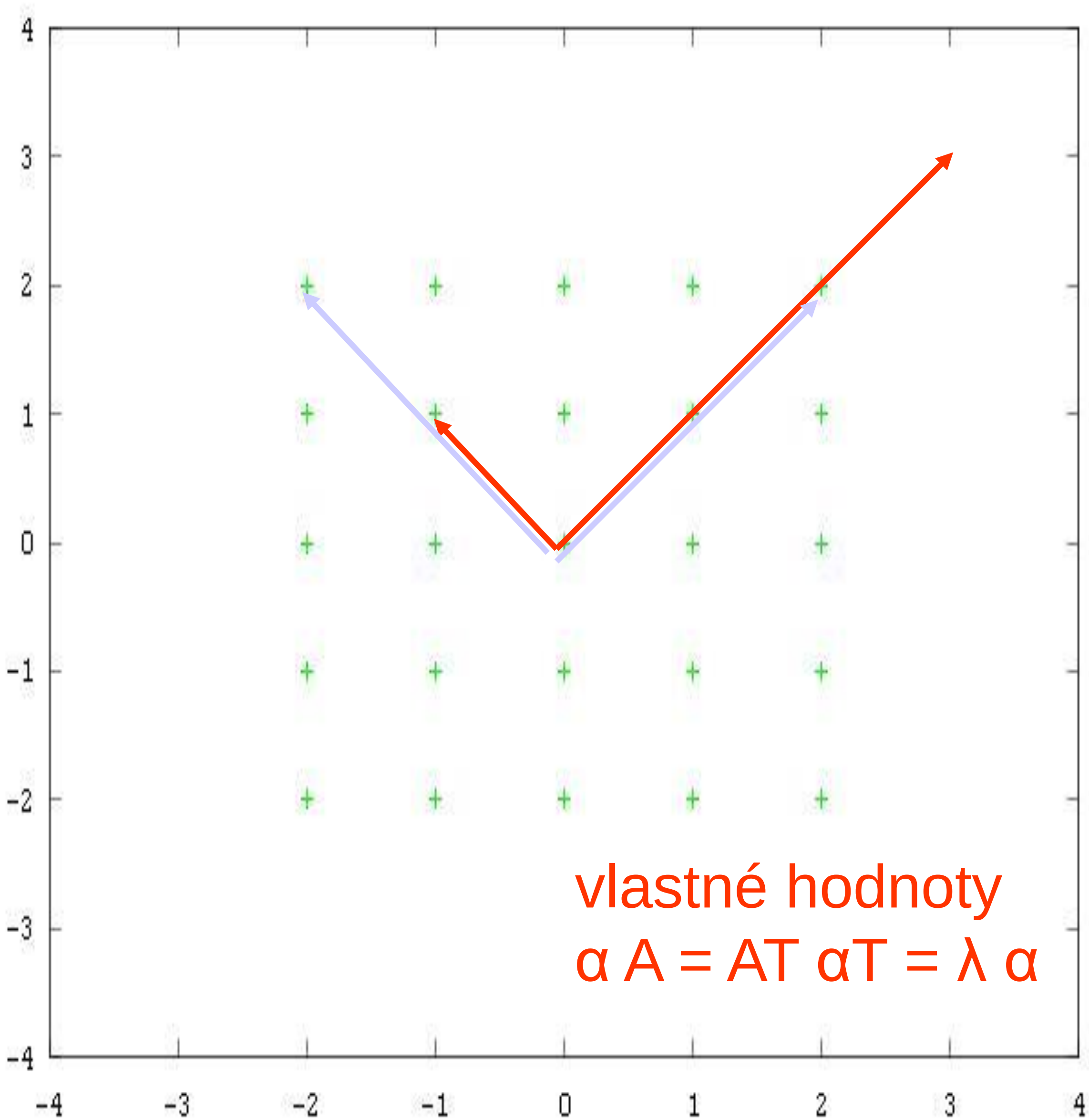


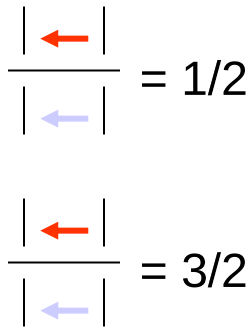
- Transformované body
pomocou transformácie A

Vlastné vektory a hodnoty



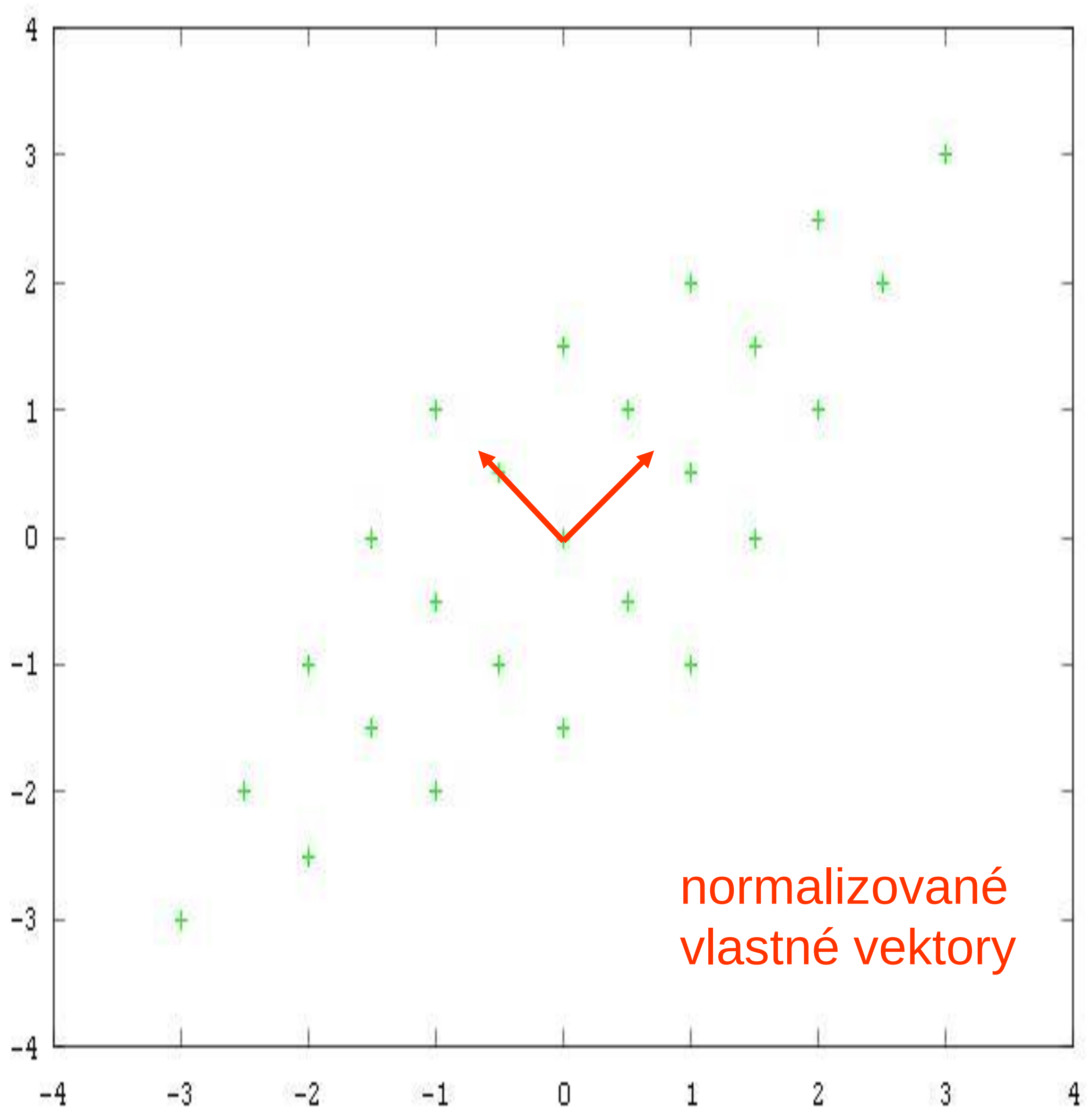
vlastné vektory udávajú v akých smeroch sa vzor deformuje na obraz vlastné hodnoty udávajú zväčšovanie či zmenšovanie v deformačných smeroch





vlastné hodnoty
α A = AT αT = λ α

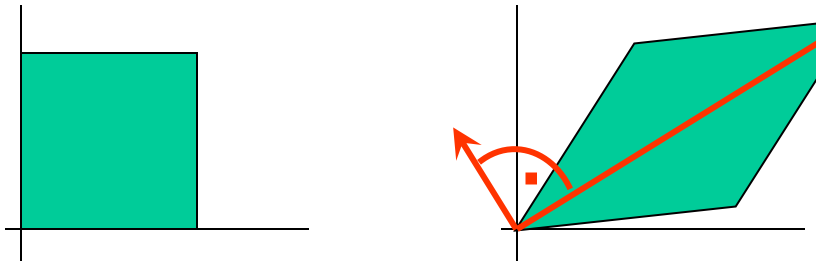
každý násobok vlastného vektoru je
rovnako vhodný ale iba jeden má dĺžku 1



$|\leftarrow| = 1$

Symetrická matica

- $A = A^T$
- zmena bázy má vždy charakter kosoštvorcovej deformácie s otočením a s prevrátením alebo bez:



- vlastné hodnoty sú reálne čísla (A je Hermitovská)

  a vlastné vektory sú na seba kolmé (uhlopriečky kosoštvorca sú na seba kolmé) a preto pre $\alpha = V^T$ $\alpha\alpha^T$ = diag takže ak vlastné vektory znormalizujeme, tak $\alpha\alpha^T = I$ a teda $\alpha^T=\alpha^{-1}$

PCA (Principal component analysis)

- Nech X je centrovaný dataset
- Hľadáme teda zmenu bázy, t.j. regulárnu maticu P rozmerov m x m takú, že

$$XP = Y \quad \text{a} \quad C_Y \text{ je diagonálna}$$

- $E(X) = \overline{X}$
- centrovanie dát sa zmenou bázy nepokazí, ak bolo centrované X, bude centrované aj Y, lebo $E(Y) = E(XP) = E(X)P = 0\,P = 0$

PCA (Principal component analysis)

pre ľubovoľné P platí:

$$C_Y = E(Y^T Y) = E((XP)^T XP) = E(P^T X^T XP) =$$
$$= P^T\, E(X^T X)\, P = P^T\, C_X\, P$$

Pre C_X sa dajú nájsť jej vlastné hodnoty a vektory (je to symetrická matica):

$$\alpha_i C_X = \lambda_i \alpha_i \qquad C_X = \alpha^{-1} \Lambda\, \alpha \qquad \alpha = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} \qquad \Lambda = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{pmatrix}$$

α_i sú ortogonálne, dajú sa z normalizovať a potom: $\alpha^{-1} = \alpha^T$

Takže keď položíme $P = \alpha^T$ dostávame:

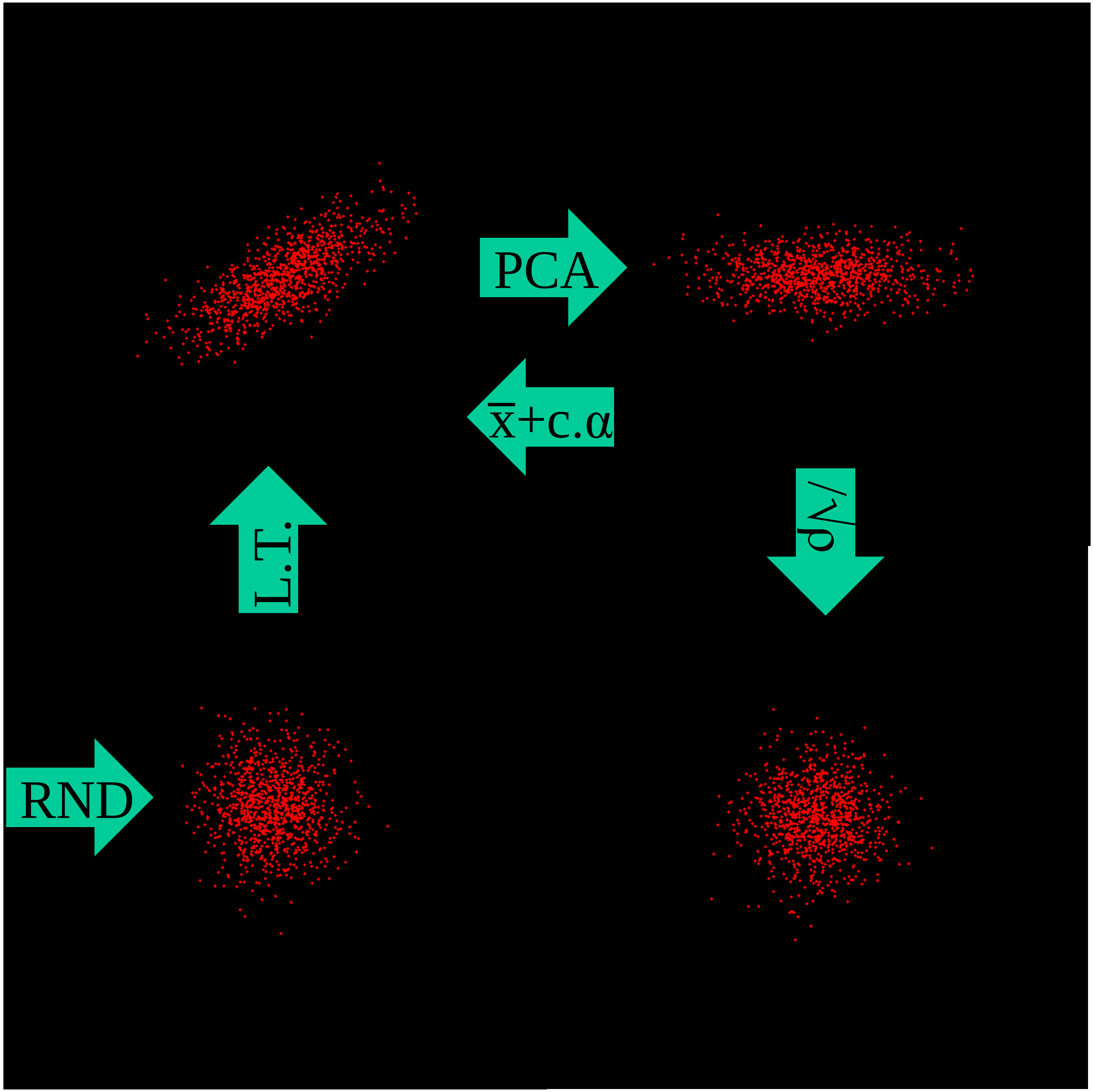
$$C_Y = P^T\, C_X\, P = P^T \alpha^{-1} \Lambda\, \alpha\, P = \alpha\, \alpha^{-1} \Lambda\, \alpha\, \alpha^T = \Lambda$$

Hľadanou transformáciou je matica, kde sú po stĺpcoch normalizované vlastné vektory C_X

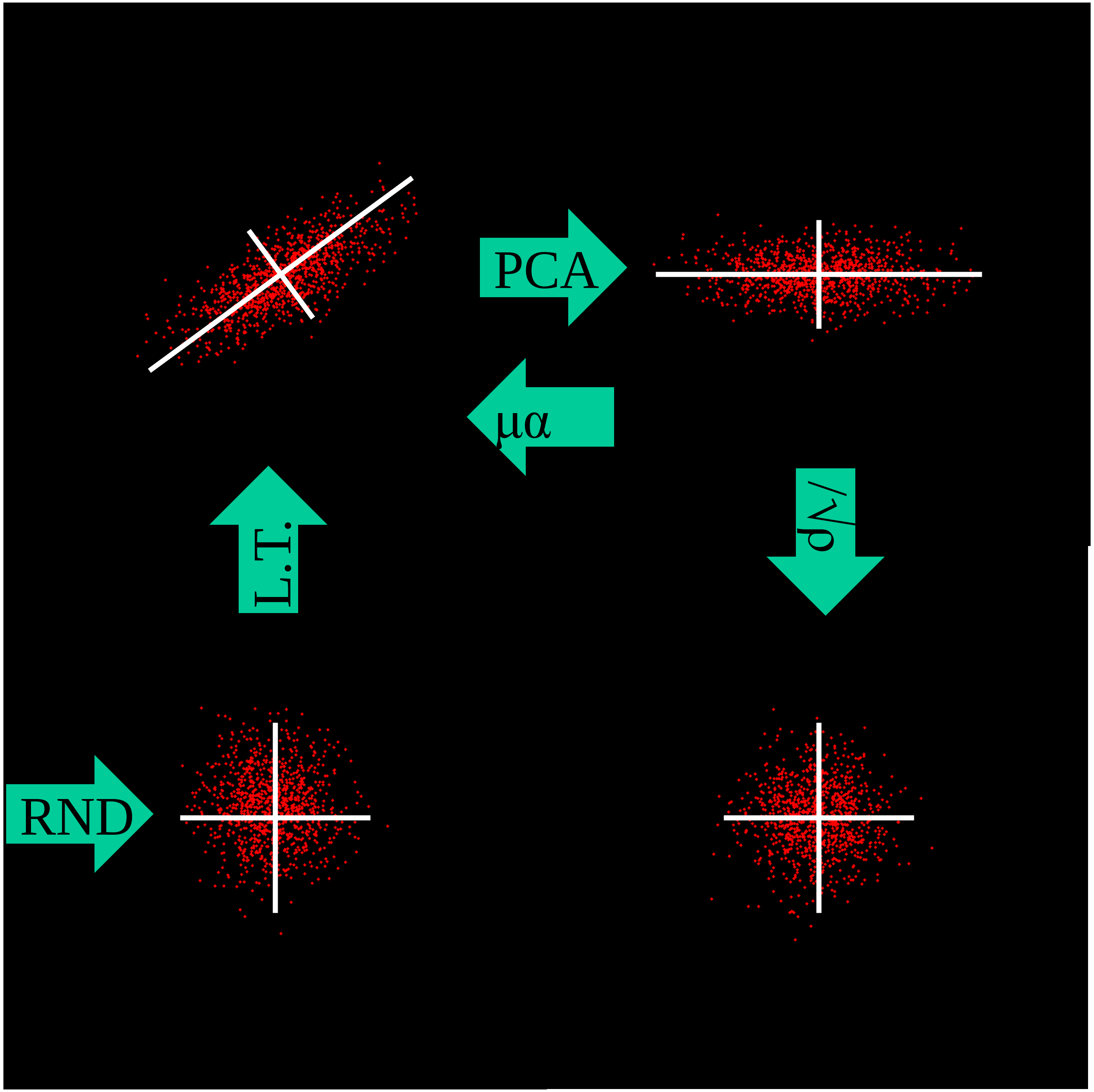
PCA (Principal component analysis)

- Matica P, kde sú po stĺpcoch normalizované vlastné vektory C_X transformuje dáta to priestoru, kde sú ich jednotlivé zložky nekorelujúce
- absolútna hodnota vlastných hodnôt C_X dáva navyše informáciu aký vplyv má daný component (rovná sa jeho rozptylu), čo súvisí s jeho významnosťou

Principal Component Analysis (PCA)



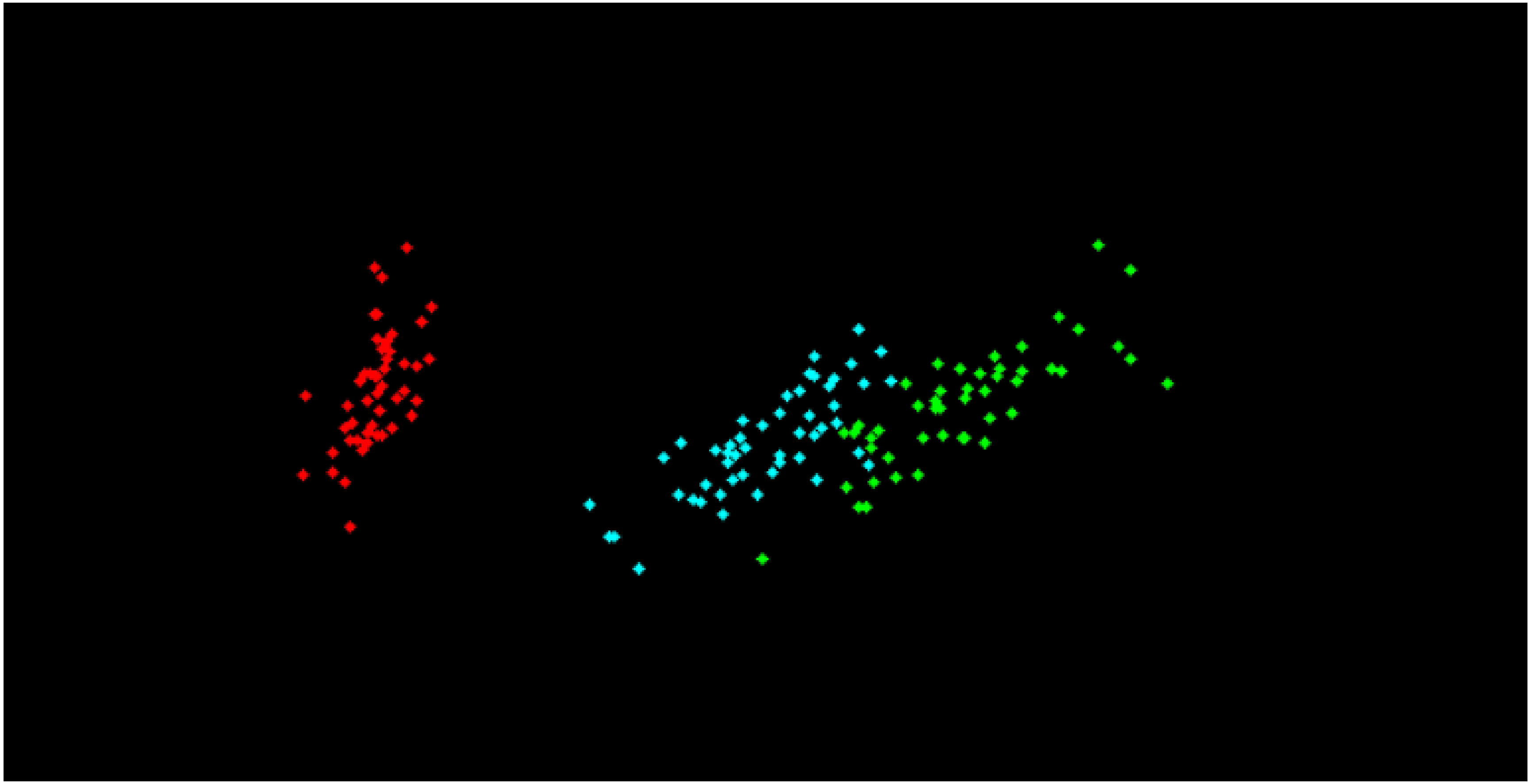
Principal Component Analysis (PCA)



PCA (Principal component analysis)

- PCA vyjadruje pôvodné dáta ako priemer + lineárnu kombináciu vlastných vektorov (= normalizovaných vlastných vektorov vynásobených vlastnými hodnotami)
- Dimenzie s malými vlastnými hodnotami len málo vplývajú na výslednú hodnotu, takže ich môžeme zanedbať a tak PCA sa dá použiť na redukciu dimenzie

Zobrazenie mnohorozmerných dát pomocou PCA



Dataset tvárí

Eigenimages

čo sa dá urobiť s bodkami, dá sa urobiť aj s obrázkami...

Dataset tvárí rozložíme na priemer + násobky vlastných vektorov

rôzne násobky

Eigenimages

Obmedzením sa na významné komponenty poskytuje nám PCA transformáciu obrazu na **feature vector**, ktorý má oveľa menšiu dimenziu

$$c = [c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8]$$

Matica prechodu zložená z vlastných vektorov dokáže takto transformovať akýkoľvek obrázok daných rozmerov

Ako to využiť na rozpoznanie či je niečo fotka z pasu alebo nie?
Z PCA vieme aj priemer μ a vlastné hodnoty a ich odmocniny sú smerodajné odchýlky: $\sigma = [\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6, \sigma_7, \sigma_8,]$

Takže ak $\mu - 3\sigma < c < \mu + 3\sigma$ povieme, že ide o fotku pasu a inak, že nie. (Nefunguje to moc dobre, ale trochu áno)

Všeobecná schéma klasifikácie

obraz

Metóda 1 $\rightarrow$

feature vector

$[c_1, c_2, \ldots, c_n]$

Metóda 2 $\rightarrow$

kategória

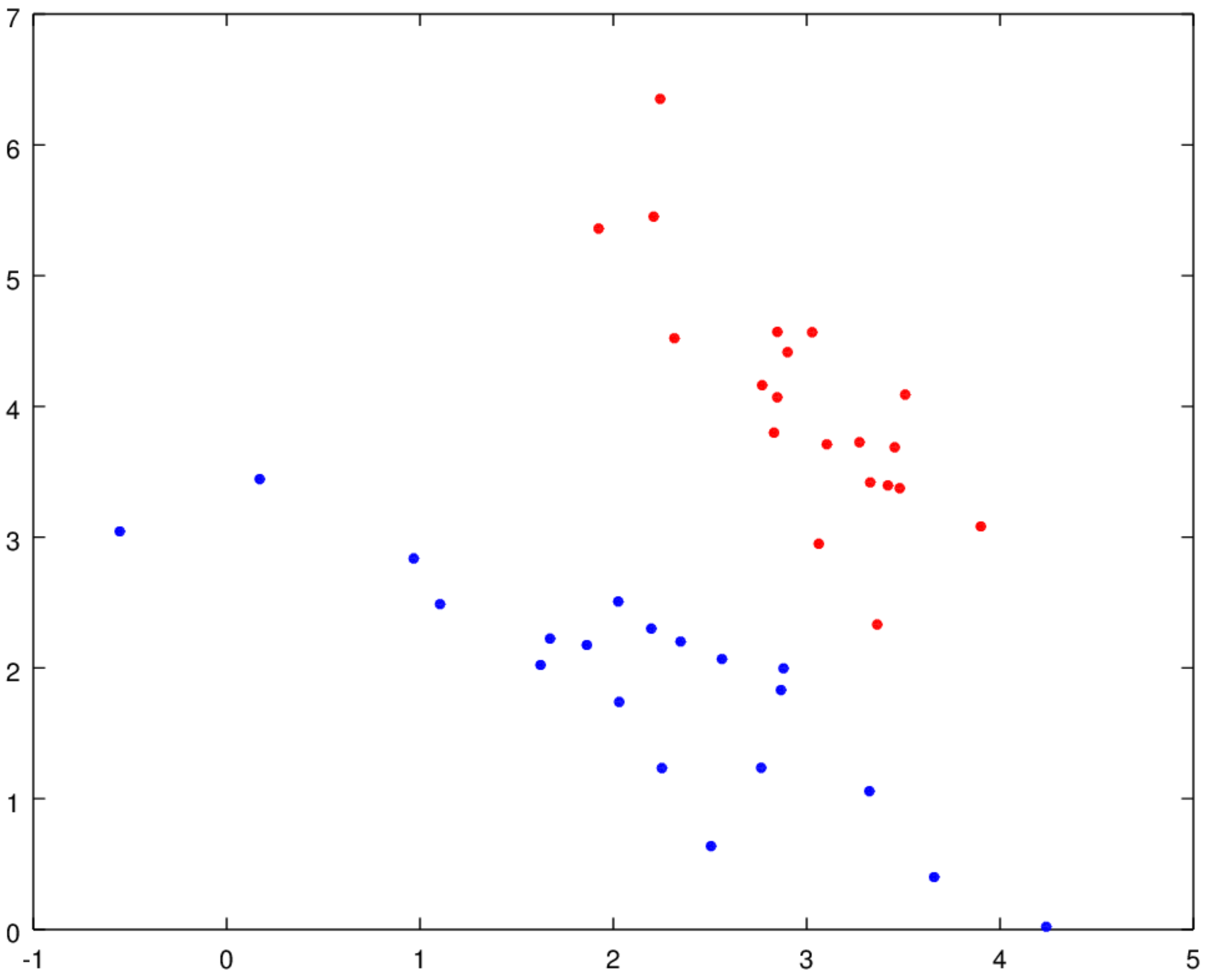
eigenimages	3σ	
eigenimages	LDA	
LBPH	LDA	(Fisher faces)
LBPH	SVM	
HOG	SVM	
Haar	Decision tree	(Viola-Jones)

Linear Discriminant Analysis (LDA)

- metóda strojového učenia
- LDA klasifikuje dáta v n-rozmernom priestore do viacerých kategórii, avšak spôsob jej fungovania si priblížime najprv na dvojrozmernom priestore a dvoch kategóriách
- Podstatou LDA je nájsť takú lineárnu transformáciu priestoru, ktorá najlepšie rozlišuje (diskriminuje) dané vzorky pri redukcii dimenzie priestoru.
- Je založená nájdení hlavných vektorov a hodnôt určitej matice, ktorá sa skonštruje zo vzoriek.

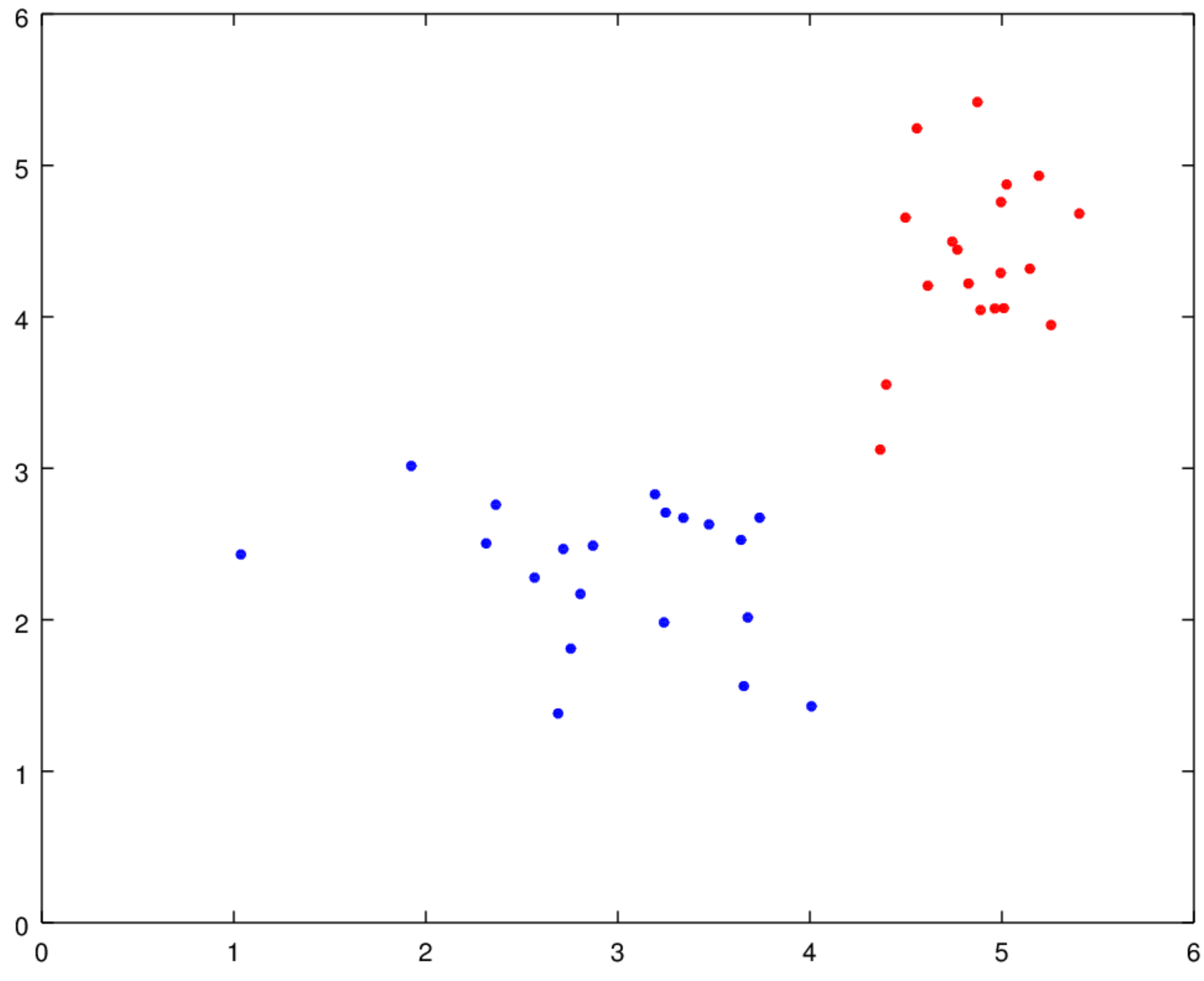
Linear Discriminant Analysis (LDA)

Uvažujme tieto vzorky dvoch kategórii



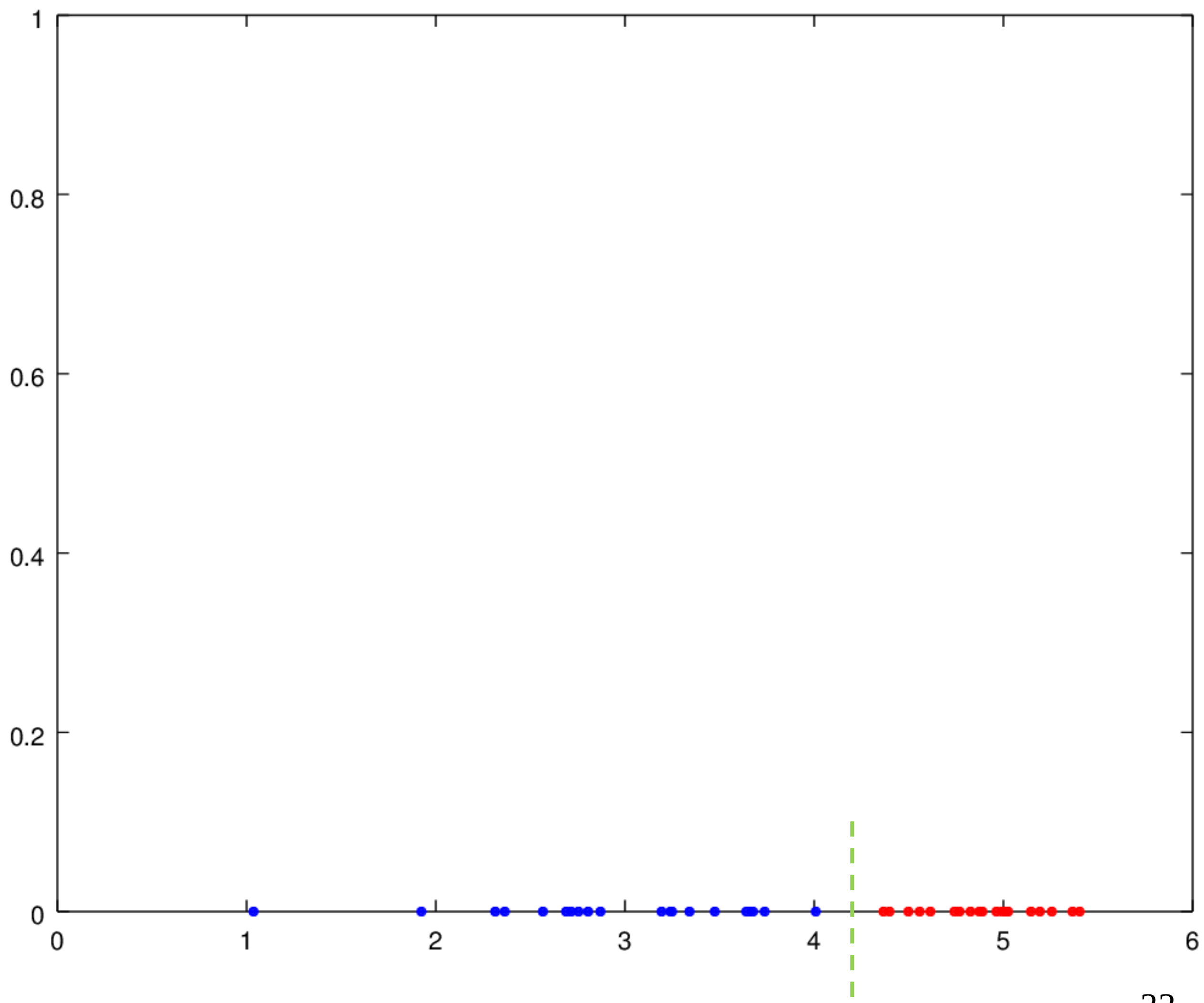
Linear Discriminant Analysis (LDA)

LDA ich dokáže transformovať ich súradnice tak ...



Linear Discriminant Analysis (LDA)

... že po redukcií dimenzie možno kategórie ľahko rozlíšiť



Linear Discriminant Analysis (LDA)

Pri transformácii priestoru dát LDA uvažuje len lineárne transformácie, pri dvojrozmerných dátach ide teda o stĺpcový vektor (označme *W*)

$$(x_1, x_2) \times \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \left(x^{'}\right)$$

čo znamená, že

$$x^{'} = w_1 x_1 + w_2 x_2$$

- Každú zmenená súradnica je určitou lineárnou kombináciou pôvodných súradníc

Linear Discriminant Analysis (LDA)

Ktorá z možných *W* je najlepšia pre dané vzorky?
Tá, ktorá kategórie (*a* a *b*) najlepšie diskriminuje.

LDA hľadá takú *W*, pre ktorú nadobúda maximum:

$$J(W) = \frac{(\mu_a^W - \mu_b^W)^2}{(\sigma_a^W)^2 + (\sigma_b^W)^2}$$

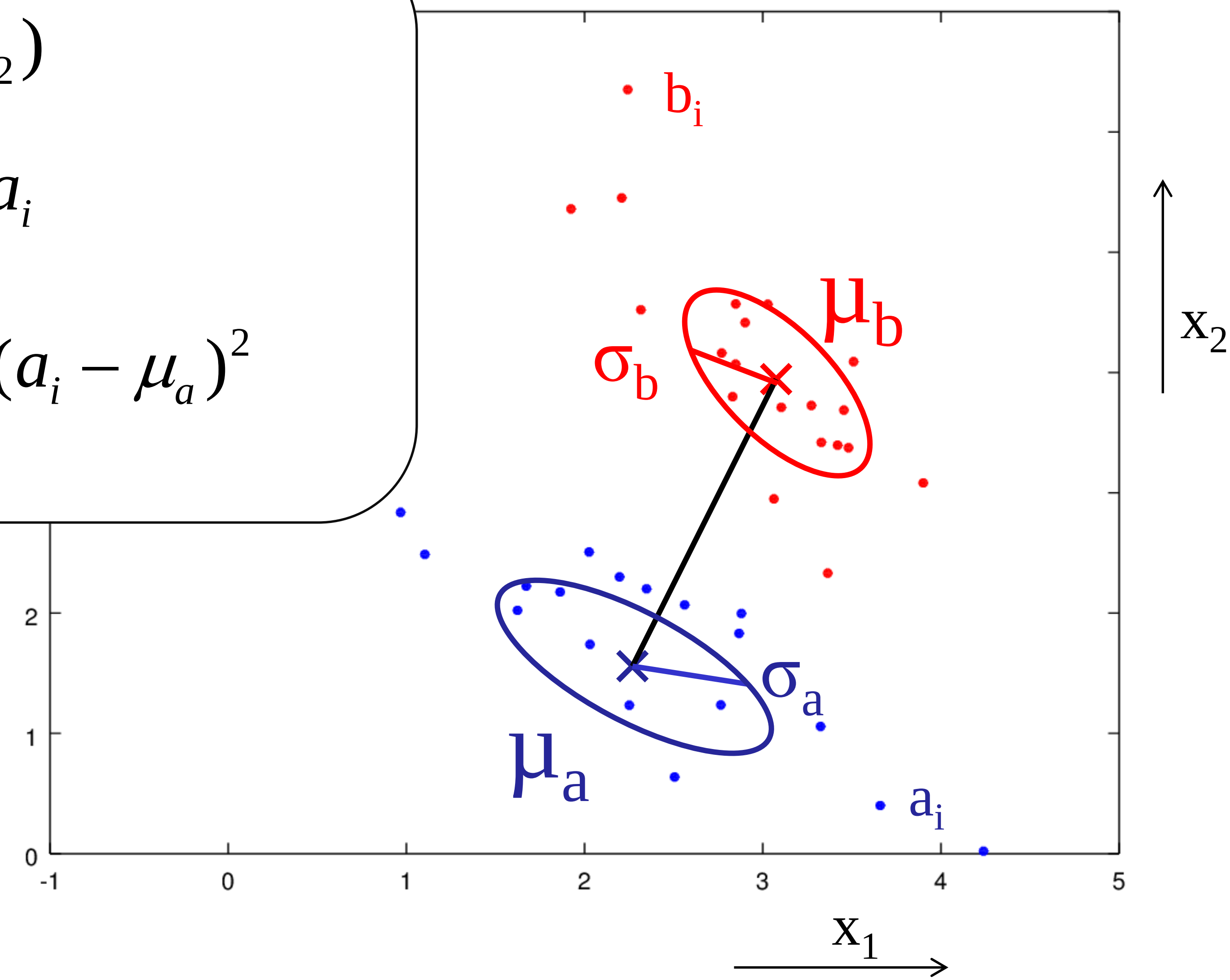
čo je približne vtedy, keď je vzdialenosť priemerov kategórii μ čo najväčšia a rozptyl vzoriek σ^2 čo najmenší

Linear Discriminant Analysis (LDA)

$$a_i = (x_1, x_2)$$

$$\mu_a = \frac{1}{n}\sum_i a_i$$

$$\sigma_a^2 = \frac{1}{n}\sum_i (a_i - \mu_a)^2$$

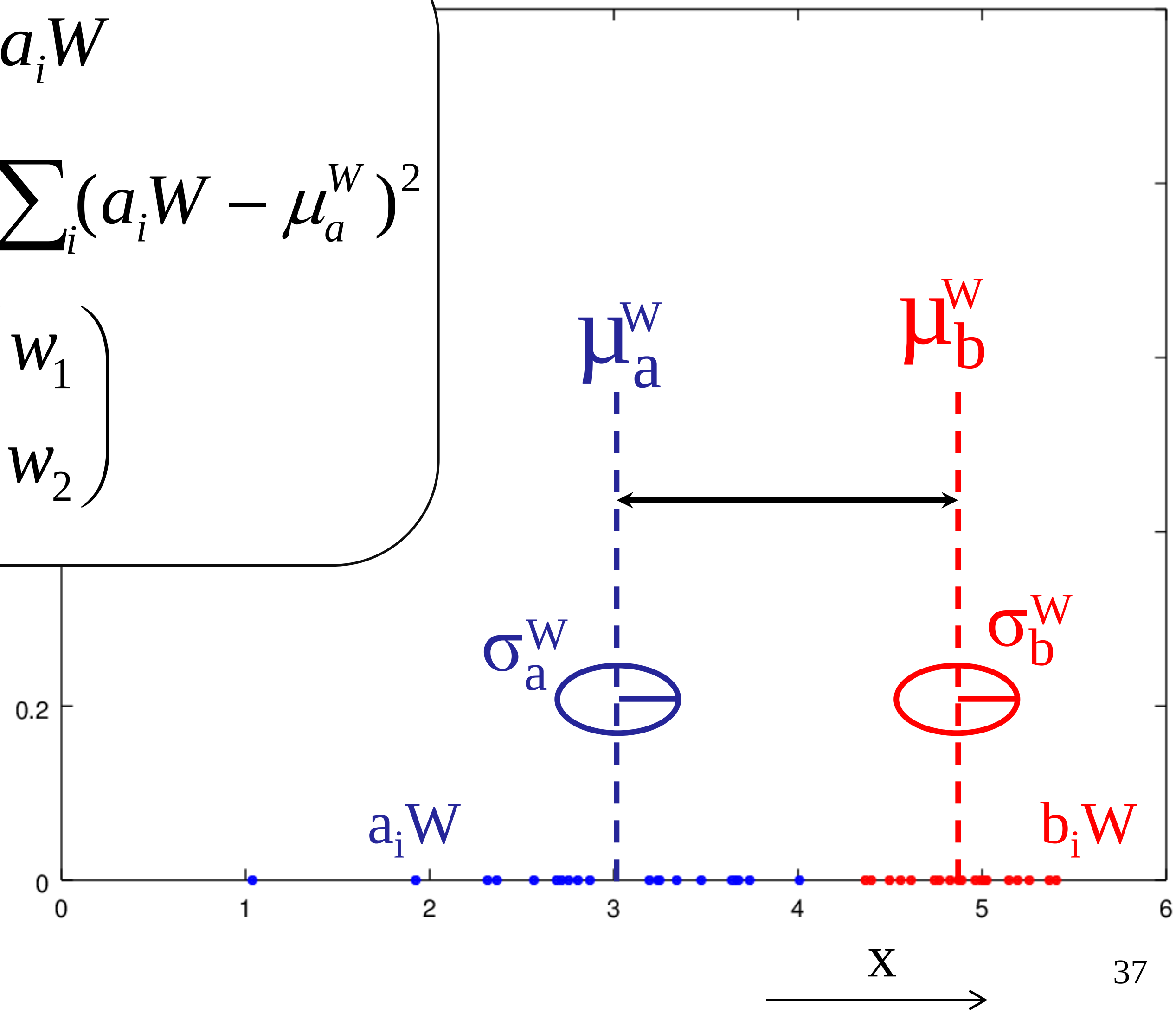


Linear Discriminant Analysis (LDA)

$$\mu_a^W = \frac{1}{n}\sum_i a_i W$$

$$(\sigma_a^W)^2 = \frac{1}{n}\sum_i (a_i W - \mu_a^W)^2$$

$$W = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$



Linear Discriminant Analysis (LDA)

Ako nájsť pre ktorú *W* nadobúda *J(W)* maximum? J(W) je diferencovateľná, možno si ju predstaviť ako kopec na vrchole ktorého je vodorovný v podľa každej zo svojich priestorových súradníc w_1, w_2

Riešime teda sústavu dvoch rovníc o dvoch neznámych:

$$\frac{dJ(W)}{dw_i} = 0$$

To sa v princípe vždy nejako dá, ale v našom špeciálnom prípade sa dá k riešeniu dostať aj (výpočtovo) ľahšie.

Linear Discriminant Analysis (LDA)

Dá sa totiž ukázať, že riešením tejto sústavy rovníc je matica zložená z tzv. vlastných vektorov určitej matice. A vlastné vektory vieme veľmi efektívne spočítať. Ako na to prídeme?

Skúmajme najprv ako sa μ_a k $\mu_a{}^W$ a σ_a ku $\sigma_a{}^W$, t.j. čo robí s priemerom a rozptylom vzoriek hľadaná transformácia W:

$$
\begin{aligned}
\mu_a^W &= \frac{1}{n}\sum_i a_i W = \mu_a W \\
\mu_b^W &= \mu_b W \\
(\mu_a^W - \mu_b^W)^2 &= W^T \overbrace{(\mu_a - \mu_b)^T (\mu_a - \mu_b)}^{S} W
\end{aligned}
$$

$S \longleftarrow$ určuje rozptyl medzi kategóriami

Linear Discriminant Analysis (LDA)

$$(\sigma_a^W)^2 \quad = \quad \frac{1}{n_a}\sum_i (a_i W - \mu_a^W)^2 = \quad \frac{1}{n_a}\sum_i (a_i W - \mu_a W)^2$$

$$= \quad \frac{1}{n_a}\sum_i (a_i W - \mu_a W)^2 = \quad \frac{1}{n_a}\sum_i W^T (a_i - \mu_a)^T (a_i - \mu_a) W$$

$$= \quad \frac{1}{n_a} W^T \underbrace{\left(\sum_i (a_i - \mu_a)^T (a_i - \mu_a)\right)}_{S_a} W \quad = \quad \frac{1}{n_a} W^T S_a W$$

$S_a \leftarrow$ kovariančná matica,
určuje rozptyl vzoriek
v rámci kategórie

$$(\sigma_b^W)^2 \quad = \quad \frac{1}{n_b} W^T S_b W$$

Linear Discriminant Analysis (LDA)

$$J(W) = \frac{(\mu_a^W - \mu_b^W)^2}{(\sigma_a^W)^2 + (\sigma_b^W)^2} = \frac{W^T S W}{\frac{1}{n_a} W^T S_a W + \frac{1}{n_b} W^T S_b W}$$

$$= \frac{W^T S W}{W^T (\underbrace{S_a / n_a + S_b / n_b}_{S_{ab}}) W} = \frac{W^T S W}{W^T S_{ab} W}$$

Linear Discriminant Analysis (LDA)

Riešime: $$\frac{dJ(W)}{dW} = 0 \qquad \frac{d}{dW}\left(\frac{W^T S W}{W^T S_{ab} W}\right) = 0$$

$$W^T S_{ab} W \frac{d}{dW}\left(W^T S\, W\right) - W^T S\, W \frac{d}{dW}\left(W^T S_{ab} W\right) = 0$$

$$W^T S_{ab} W\left(2S\, W\right) - W^T S\, W\left(2S_{ab} W\right) = 0$$

$$S\, W - \frac{W^T S\, W}{W^T S_{ab} W} S_{ab} W = 0$$

vlastná hodnota M

$$S\, W = J(W) S_{ab} W \qquad \underbrace{S_{ab}^{-1} S}_{M}\, W = J(W)\, W$$

vlastný vektor M

vlastné vektory vieme efektívne vypočítať

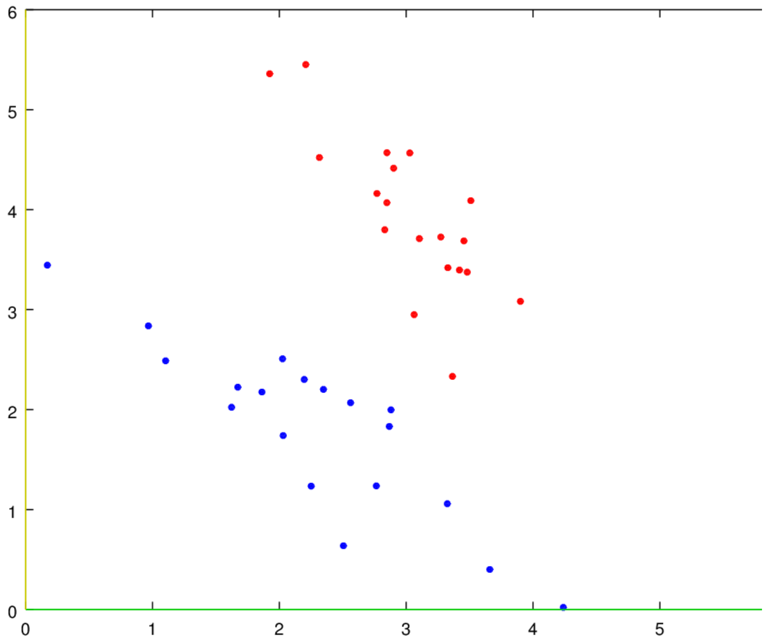
Pre dimenziu 1,2,3 a 4 priamo vzorcom

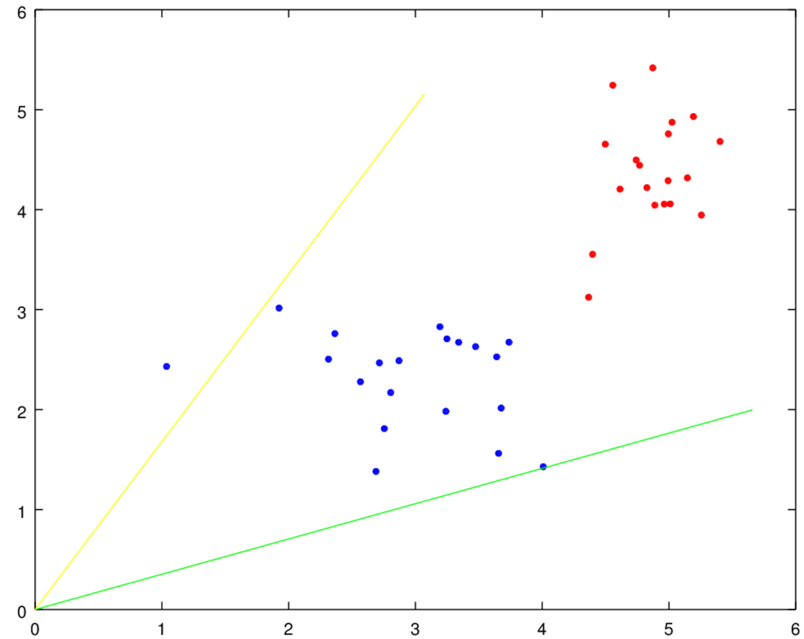
Pre 5 a viac iteračným algoritmom, v najjednoduchšej verzii vezme akýkoľvek vektor a zobrazuje ho maticou, kým sa nezobrazí (skoro) na seba, potom sa v už len z na neho kolmých vektorov hľadá ďalší taký, až sa nájdu všetky. Ale sú k dispozícii algoritmy o mnoho rafinovanejšie a rýchlejšie.

Pomocou vlastných vektorov sa počíta všeličo, napr. aj korene rovníc piateho a vyšších stupňov, ...

A keďže ho používa LDA, fáza trénovania zo vzoriek je pri LDA – v porovnaní s inými metódami strojového učenia - neobvykle rýchla.

Pri LDA teda zo vzoriek spočítame správnu maticu, nájdeme jej vlastné vektory a ten s väčšou vlastnou hodnotou definuje tú správnu transformáciu





Pomocou tohto triku LDA dokáže rozlišovať kategórie ktoré definuje pomocou vzoriek a to aj pri väčšej dimenzii priestoru a väčšom počte kategórií. Samozrejme len natoľko dokonale, ako to povaha vzoriek umožňuje.

LDA – dimenzia > 2

Pri väčšej dimenzii postupujeme analogicky. Nájdeme vlastné vektory a hodnoty príslušnej matice a tie vlastné, ktoré majú významne veľké vlastné hodnoty, zložíme po stĺpcoch do matice, ktorá bude maticou prechodu k priestoru s redukovanou dimenziou.

Prax ukazuje, že výrazných vlastných hodnôt je oveľa menej než je pôvodná dimenzia, takže redukcia dimenzie je výrazná.

LDA – viac kategórii

Pri viacerých kategóriách sa dá vymyslieť podobný postup.

Minimalizujeme pri ňom

$$J(W) = \frac{\sum_i (\mu_i^W - \mu^W)^2}{\sum_i (\sigma_i^W)^2}$$

kde i je kategória a μ^W je priemer všetkých vzoriek