

# Rozpoznávanie obrazcov - 5. cvičenie

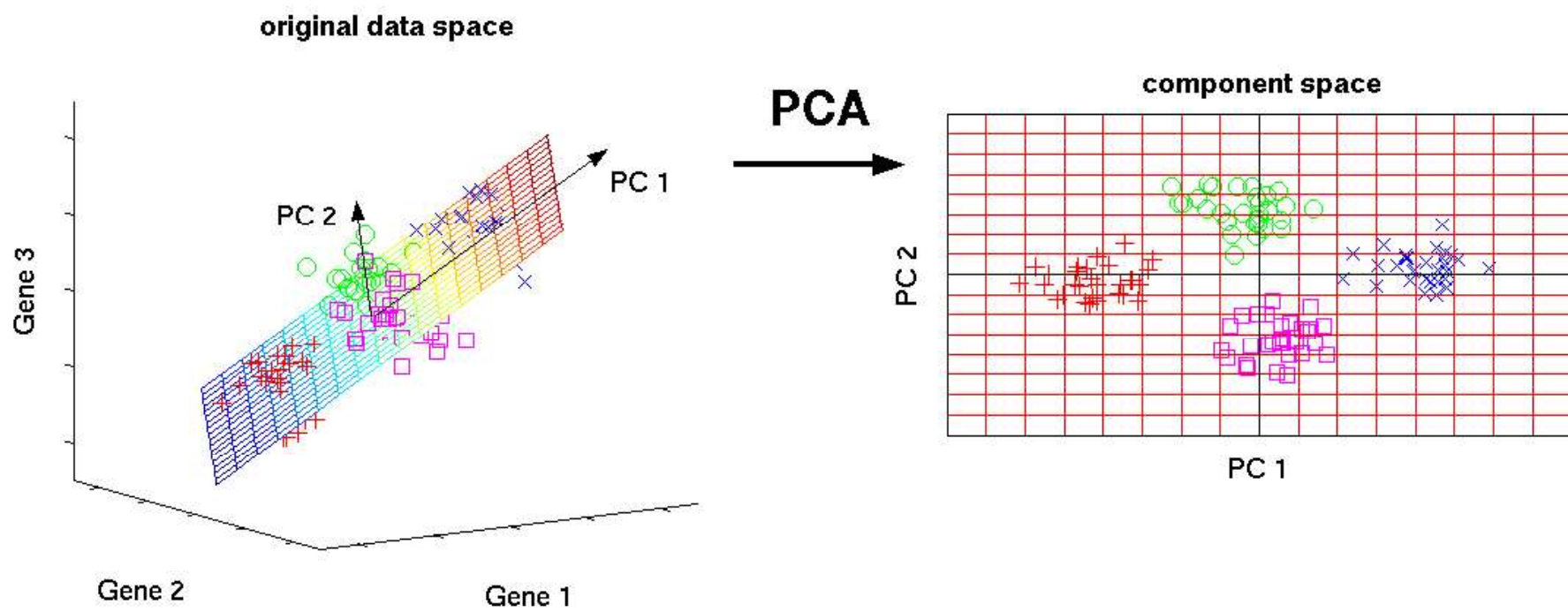
## Redukcia dimensionalita

Viktor Kocur  
viktor.kocur@fmph.uniba.sk

DAI FMFI UK

16.3.2020

# PCA princíp



# PCA matematika

Chceme nájsť ortogonálnu bázu  $W$  (*transformačnú maticu*), ktorou keď dáta prenásobíme, dostaneme nekorelované dáta.

PCA: Ortogonálnu bázu  $W$  pre dáta  $X$  vieme vypočítať ako vlastné vektory kovariančnej matice vstupných dát  $X$ .

1. Zoberieme databázu
2. Odčítame priemer od každého rozmeru prvku databázy
3. Vypočítame kovariančnú maticu
4. Vypočítame vlastné čísla a vlastné vektory tejto kovariančnej matice
5. Vyberieme hlavné komponenty
6. Vypočítame novú databázu

# PCA matematika

## Kovariancia

$$\text{cov}(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{n}$$

## Kovariančná matica

$$\text{COV}(X)_{i,j} = \text{cov}(X_i, X_j)$$

## Kovariančná matica

Kovariančná matica je Hermitovská a pozitívne semi-definitná.

# PCA matematika

## PCA - postup

Naše dáta najprv centrumjeme  $\vec{x}' = \vec{x} - \bar{x}$ . Potom vypočítame  $W$ . Nové dáta dostaneme maticovým násobením  $y = \vec{x}'W^T$ , ak je vektor  $\vec{x}$  riadkový. Vlastné čísla korešpondujú k podielu celkovej variancie v danom smere. Podiel vlastného čísla so súčtom vlastných čísiel určuje aký podiel variancie vie daný smer vysvetliť.

# Postup matlab

## Načítanie dát

```
load data.mat
```

## Úloha

Zobrazte si dáta v 2D plote.

# Postup matlab

## cov

`cov(A)` - vráti kovariančnú maticu  $A$

## eig

`[W, vals] = eig(A)` - vráti maticu  $W$  s vlastnými vektormi a maticu `vals` s vlastnými číslami na diagonále.

## Úloha

Aplikujte na dáta PCA a zobrazte si nový plot. Ktorá zložka zodpovedá akej variancii.

# Riešenie

```
load data.mat
plot(data(:,1), data(:,2), 'r*');
centered = data - mean(data);
[W, eigenvals] = eig(cov(centered))
newdata = centered * W'
plot(newdata(:,1), newdata(:,2), 'r*');
ylim([-2 2]);
xlim([-2 2]);
disp(diag(eigenvals)/sum(diag(eigenvals)))
```



# Matlab - pca

## pca

`[coeff,score,~,~,explained,mu] = pca(X)` - vráti transformačnú maticu `coeff` (naše  $W^T$ ), transformované dáta `score`, percentá na koľko vysvetľujú varianciu jednotlivé smery a stredné hodnoty  $X$ .

## Platí

$\text{score} == (X - \mu) * \text{coeff}$

### Octave:

`[coeff, score, eigvals,tsquare] = princomp(X);`  
`[coeff, eigvals, explained] = pcacov(cov(X));`

## Platí

$X == \text{score} * \text{coeff}' + \mu$

## Úloha

Otestujte túto funkciu na `data.mat`

# Matlab - pca

## Dáta

load ovariancancer

Octave nemá gscatter, takže musíte **scatter** zavolať na každú kategóriu dát zvlášť (kreslite do rovnakej plochy použitím príkazu **hold on** medzi kreslením jednotlivých kategórií)

gscatter

gscatter(obs(:,1), obs(:,2), grp) - zobrazí body z prvého a druhého stĺpca pre dáta a pridelí im farbu podľa príslušnosti v grp

## Úloha

Zistite koľko príznakov potrebujete po aplikácii PCA, aby ste s dát v obs dostali 95 percent variancie. Čo ak niektoré vlastné čísla sú nulové?

## Úloha

Zobrazte si prvé dva smery po PCA pomocou gscatter.

# PCA - úlohy

## Úloha

Pre dáta z data.mat zobrazte smery do ktorých PCA premetie dáta v originálnej súradnicovej sústave.

## Úloha

Vytvorte funkciu ktorá zoberie obrázok a ako dáta vezme jednotlivé trojice RGB pixelov. Urobí PCA na týchto dátach a nastaví na posledný (alebo posledné dva) stĺpce nulu a prekonvertuje obraz naspäť do RGB.

## Úloha

Pre PCA z druhej úlohy zobrazte v novom obrázku rôzne farby, ktoré môžete v tejto reprezentácii používať.

# LDA

## LDA.m

$[Y, W, \text{lambdas}] = \text{LDA}(X, T)$  - pre dáta  $X$  a triedy  $T$  vráti nové hodnoty po transformácii  $Y$ , maticu  $W$  a hodnotu jednotlivých vlastných čísel.

## Pozn

$$Y == X * W$$

## Úloha

Načítajte si Fisherovu databázu (`load fisheriris`). A porovnajte LDA a PCA. Pozn.: ako  $T$  v LDA použite `categorical(species)`.

## Úloha

Pre rôzne dvojice stĺpcov nakreslite do grafu smer do ktorého bude LDA premietat' dáta.

# LDA

## LDA.m

`MdlLinear = fitcdiscr(X,T)` - vráti lineárny klasifikátor, ktorý využíva LDA. V projekte asi používajte toto.

## Tutorial

<https://www.mathworks.com/help/stats/create-and-visualize-discriminant-analysis-classifier.html>