Rozpoznávanie obrazcov

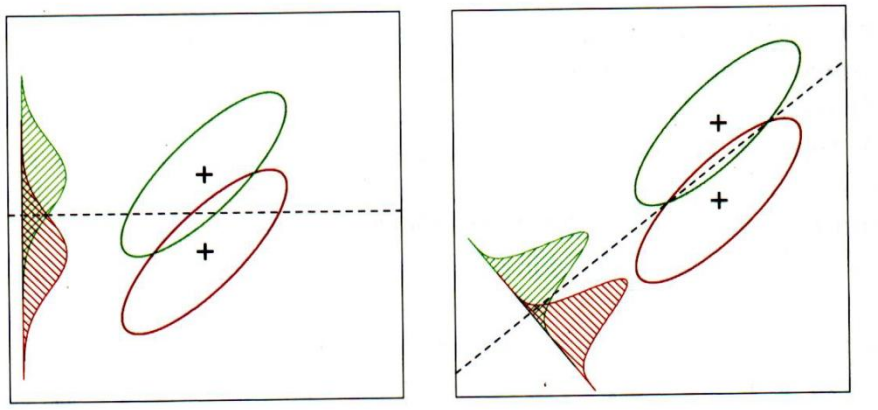
PCA, LDA, ICA

17.3.2014

- **Principal Component Analysis**
- Linear Discriminant Analysis
- Independent Component Analysis

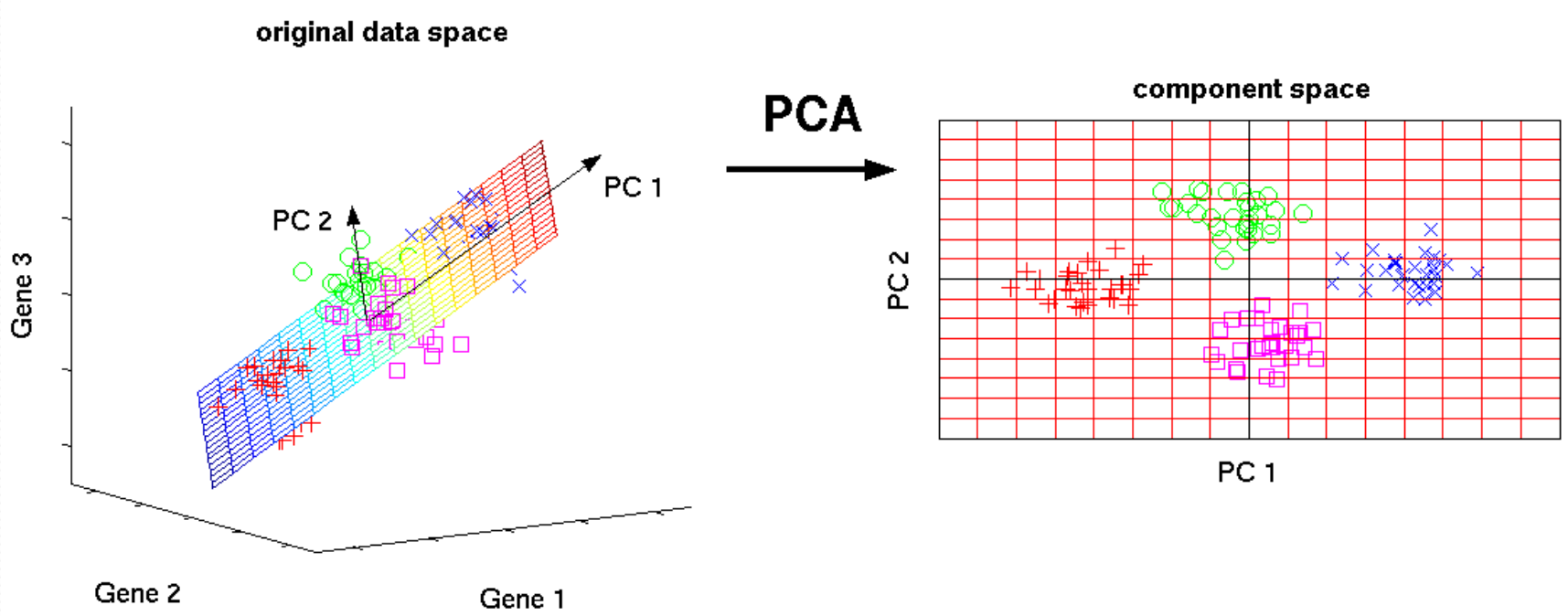
LDA

- Linear Discriminant Analysis
- maximalizuje medzitriednu a minimalizuje vnútrotriednu variabilitu



PCA

- Principal Component Analysis
  - otočí súradnicovú sústavu tak, aby prvá os bola v smere najväčšej variability a ďalšie boli na ňu kolmé v smeroch najväčšej zvyšnej variability



Dáta

↓

Normalizovanie dát

↓

Vlastné vektory , Vlastné hodnoty

↓

Zoradenie Vlastných vektorov podľa Vlastných hodnôt

↓

Finálne dáta

PCA MATLAB

- [COEFF,SCORE] = princomp(X)
- [COEFF,SCORE,latent] = princomp(X)
- [COEFF,SCORE,latent,tsquare] = princomp(X)
- [...] = princomp(X,'econ')

PCA MATLAB – príklad

```
data=rand(10,1000);
data=data-repmat(mean(data,2),1,size(data,2));
[W, EvalueMatrix] = eig(cov(data'));
Evalues = diag(EvalueMatrix);
Evalues = Evalues(end:-1:1);
W = W(:,end:-1:1); W=W';
pc = W * data;
figure, plot(pc(1,:),pc(2,:),'.')
```

PCA MATLAB – príklad

```
X = [2.5 2.4; 0.5 0.7; 2.2 2.9; 1.9 2.2; 3.1 3.0; 2.3 2.7; 2 1.6; 1
    1.1; 1.5 1.6; 1.1 0.9];
figure, plot(X(:,1), X(:,2),'.');
[COEFF,SCORE,latent] = princomp(X)
figure, plot(SCORE(:,1), SCORE(:,2),'r.');
```

Toolboxy

- Matlab Toolbox for Dimensionality Reduction
  - http://homepage.tudelft.nl/19j49/Matlab_Toolbox_for_Dimensionality_Reduction.html
- Statistical Pattern Recognition Toolbox
  - http://cmp.felk.cvut.cz/cmp/software/stprtool/

Matlab Toolbox for Dimensionality Reduction

```
[X, labels] = generate_data('helix', 2000);
figure, scatter3(X(:,1), X(:,2), X(:,3), 5, labels); title('Original dataset'), drawnow
no_dims = round(intrinsic_dim(X, 'MLE'));
disp(['MLE estimate of intrinsic dimensionality: ' num2str(no_dims)]);
[mappedX, mapping] = compute_mapping(X, 'PCA', no_dims);
figure, scatter(mappedX(:,1), mappedX(:,2), 5, labels); title('Result of PCA');
[mappedX, mapping] = compute_mapping(X, 'Laplacian', no_dims, 7);

figure, scatter(mappedX(:,1), mappedX(:,2), 5, labels(mapping.conn_comp)); title('Result of Laplacian Eigenmaps');
drawnow
```

Statistical Pattern Recognition Toolbox - príklad

```
% Generate data
distrib.Mean = [[5;4] [4;5]]; % mean vectors
distrib.Cov(:,:,1) = [1 0.9; 0.9 1]; % 1st covariance
distrib.Cov(:,:,2) = [1 0.9; 0.9 1]; % 2nd covariance
distrib.Prior = [0.5 0.5]; % Gaussian weights
data = gmmsamp(distrib,250); % sample data

lda_model = lda(data,1); % train LDA
lda_rec = pcarec(data.X,lda_model);
lda_data = linproj(data,lda_model);

pca_model = pca(data.X,1); % train PCA
pca_rec = pcarec(data.X,pca_model);
pca_data = linproj( data,pca_model);

figure; hold on; axis equal; % visualization
ppatterns(data);
```

Statistical Pattern Recognition Toolbox - príklad

```
h1 = plot(lda_rec(1,:),lda_rec(2,:),'r');
h2 = plot(pca_rec(1,:),pca_rec(2,:),'b');
legend([h1 h2],'LDA direction','PCA direction');

figure; hold on;
subplot(2,1,1); title('LDA'); ppatterns(lda_data);
pgauss(mlcgmm(lda_data));
subplot(2,1,2); title('PCA'); ppatterns(pca_data);
pgauss(mlcgmm(pca_data));
```