Machine Learning for Signal Processing

Applications of Linear Gaussian Models

Recap: MAP Estimators

- MAP (*Maximum A Posteriori*): Find most probable value of $\mathbf{y}$ given $\mathbf{x}$

$$\mathbf{y} = argmax_{Y}\, P(\mathrm{Y}|\mathbf{x})$$

MAP estimation

- x and y are jointly Gaussian

$$z = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$E[z] = \mu_z = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$$

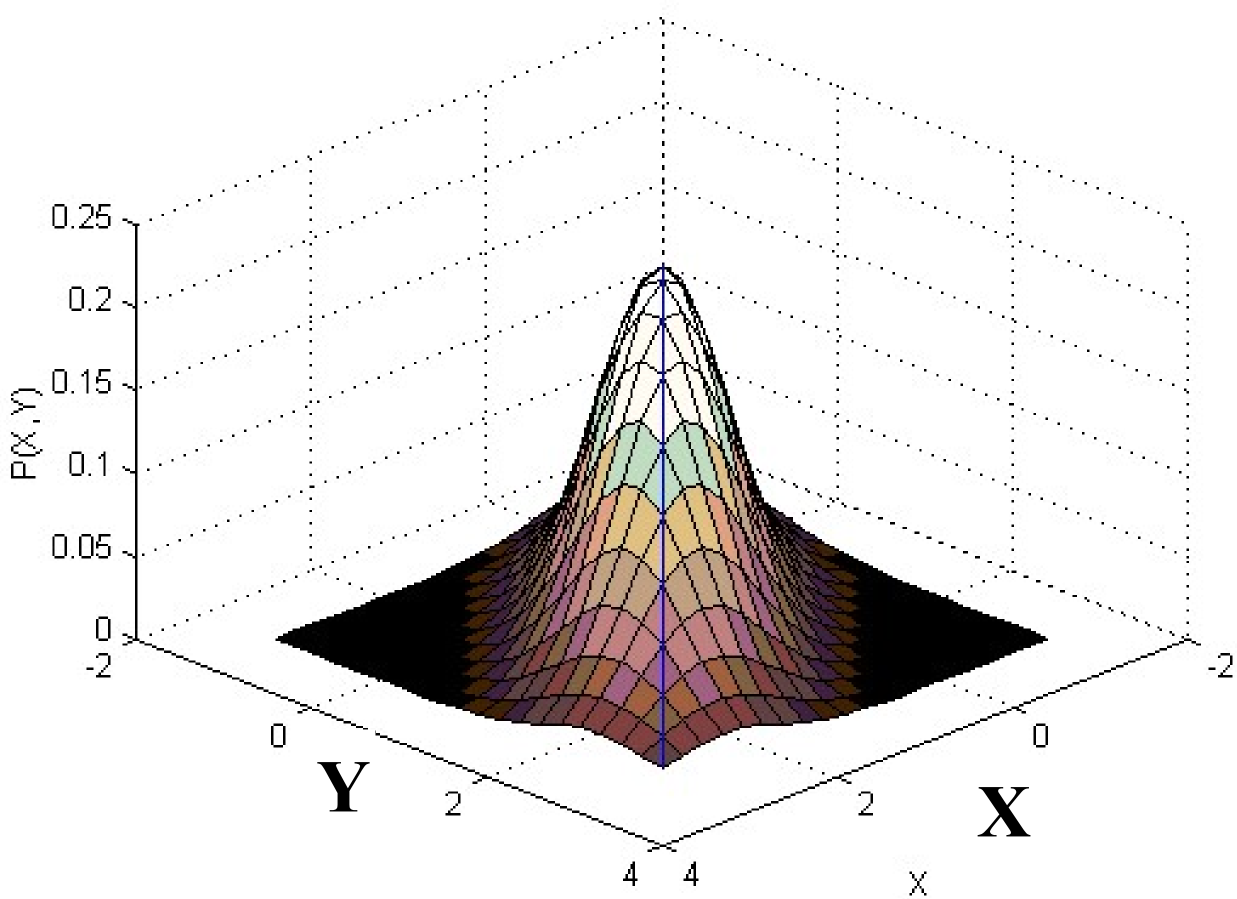
$$Var(z) = C_{zz} = \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix}$$

$$C_{xy} = E[(x - \mu_x)(y - \mu_y)^T]$$

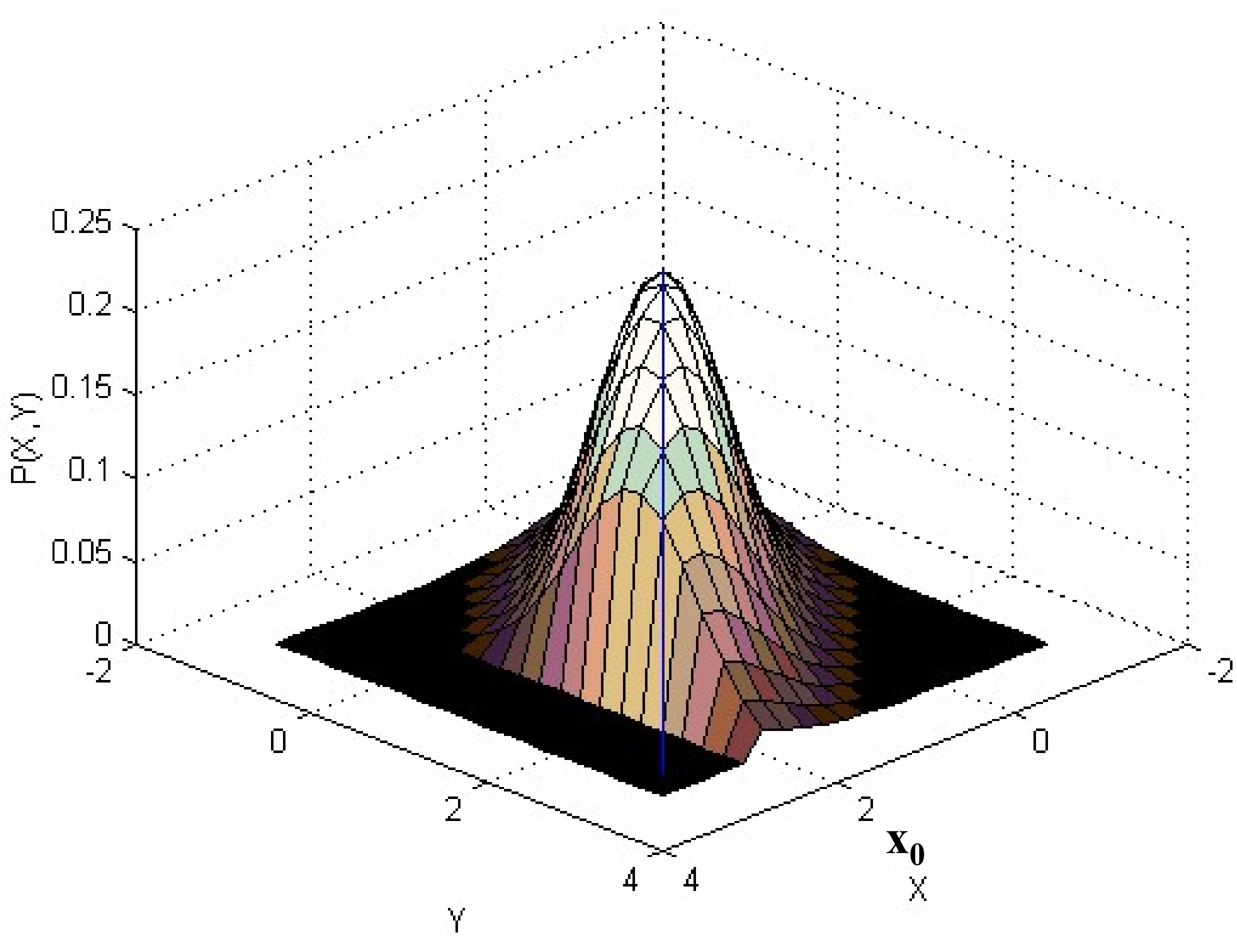
$$P(z) = N(\mu_z, C_{zz}) = \frac{1}{\sqrt{2\pi |C_{zz}|}} \exp\left(-0.5(z - \mu_z)^T C_{zz}^{-1}(z - \mu_z)\right)$$

- **z is Gaussian**

MAP estimation: Gaussian PDF



MAP estimation: The Gaussian at a particular value of X

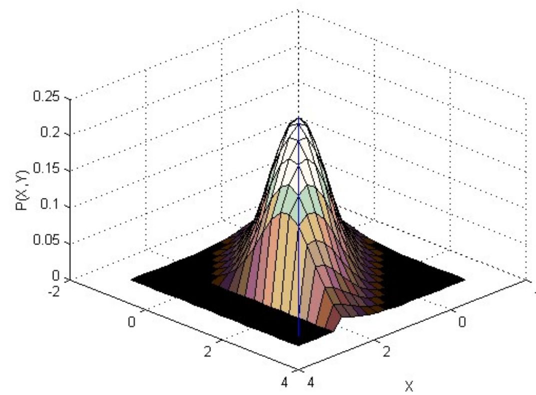


Conditional Probability of y|x

$$P(y \mid x) = N(\mu_y + C_{yx}C_{xx}^{-1}(x-\mu_x), C_{yy} - C_{yx}C_{xx}^{-1}C_{xy})$$

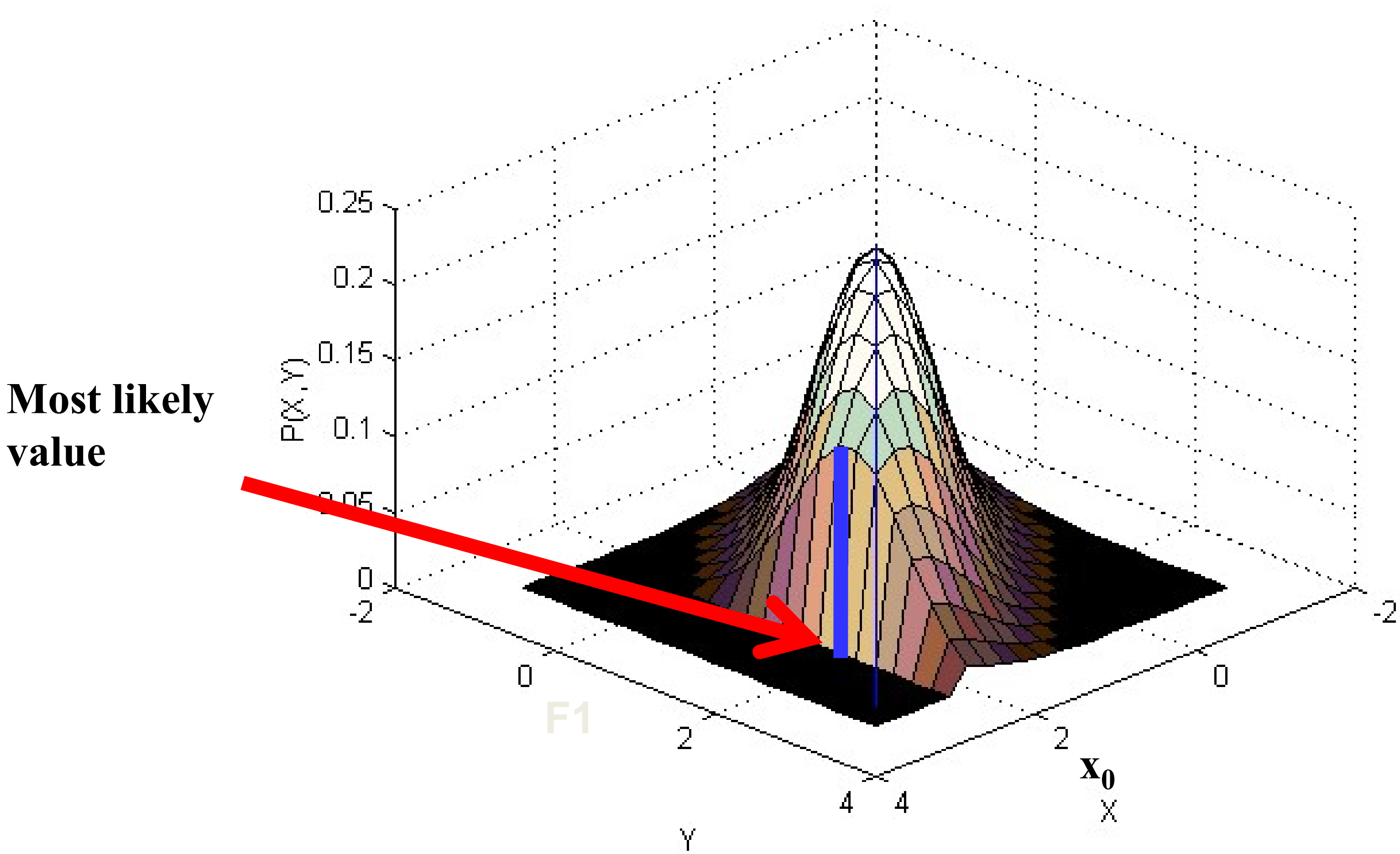
$$E_{y|x}[y] = \mu_{y|x} = \mu_y + C_{yx}C_{xx}^{-1}(x-\mu_x)$$

$$Var(y \mid x) = C_{yy} - C_{yx}C_{xx}^{-1}C_{xy}$$



- The conditional probability of y given x is also Gaussian
  - The slice in the figure is Gaussian
- The mean of this Gaussian is a function of x
- The variance of y reduces if x is known
  - Uncertainty is reduced

MAP estimation: The Gaussian at a particular value of X



Its also a *minimum-mean-squared error* estimate

- Minimize error:

$$Err = E[\|\mathbf{y} - \hat{\mathbf{y}}\|^2 \mid \mathbf{x}] = E[(\mathbf{y} - \hat{\mathbf{y}})^T (\mathbf{y} - \hat{\mathbf{y}}) \mid \mathbf{x}]$$

$$Err = E[\mathbf{y}^T\mathbf{y} + \hat{\mathbf{y}}^T\hat{\mathbf{y}} - 2\hat{\mathbf{y}}^T\mathbf{y} \mid \mathbf{x}] = E[\mathbf{y}^T\mathbf{y} \mid \mathbf{x}] + \hat{\mathbf{y}}^T\hat{\mathbf{y}} - 2\hat{\mathbf{y}}^T E[\mathbf{y} \mid \mathbf{x}]$$

- Differentiating and equating to 0:

$$d.Err = 2\hat{\mathbf{y}}^T d\hat{\mathbf{y}} - 2E[\mathbf{y} \mid \mathbf{x}]^T d\hat{\mathbf{y}} = 0$$

$$\hat{\mathbf{y}} = E[\mathbf{y} \mid \mathbf{x}]$$

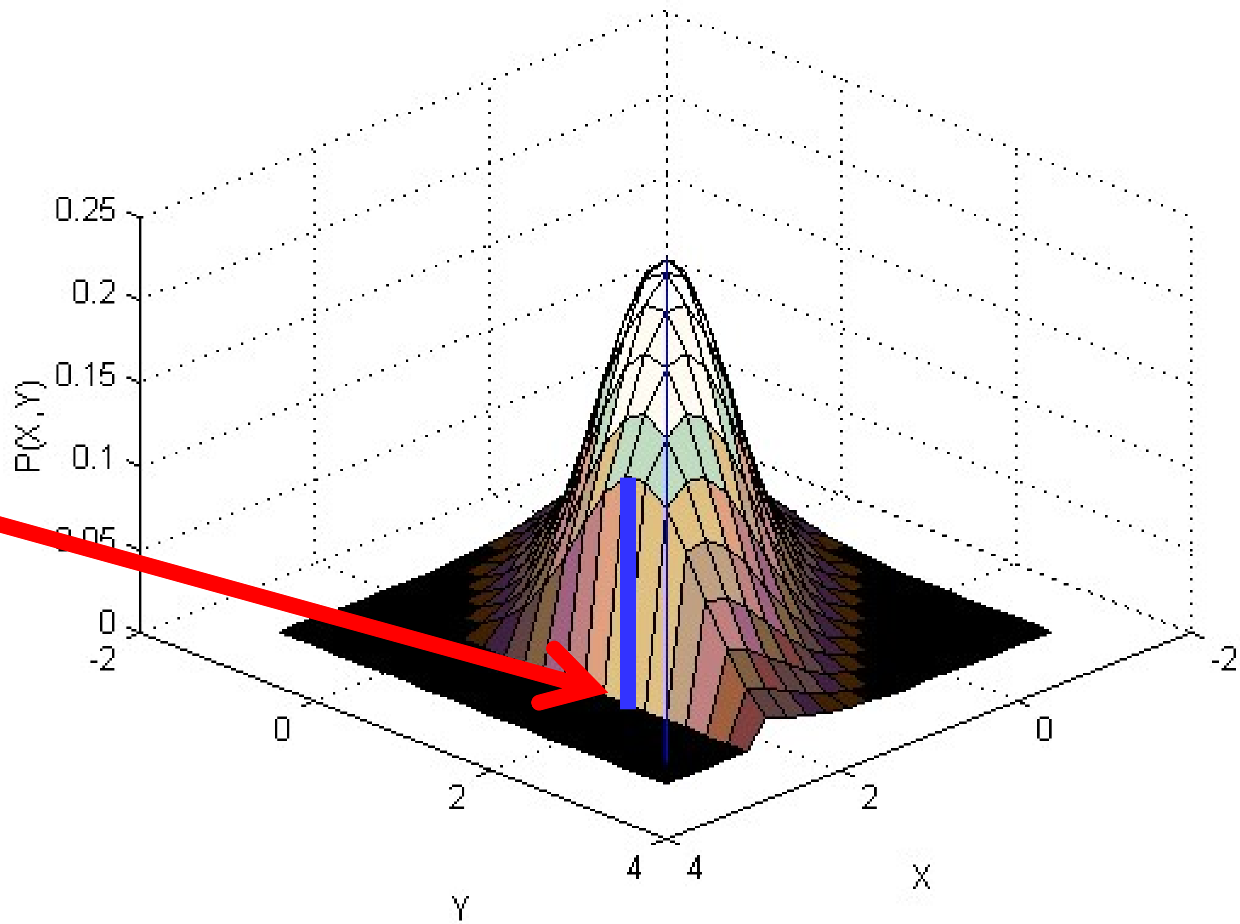
The MMSE estimate is the mean of the distribution

For the Gaussian: MAP = MMSE

Most likely value

is also

The MEAN value



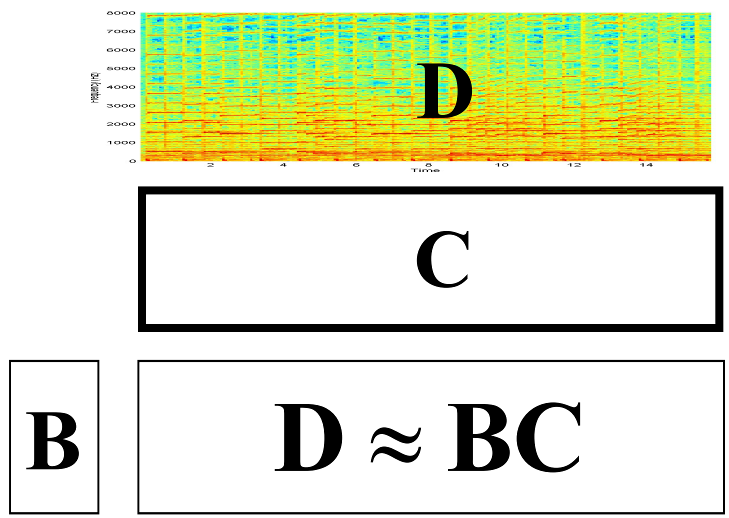
- Would be true of any symmetric distribution

Gaussians and more Gaussians..

- Linear Gaussian Models..

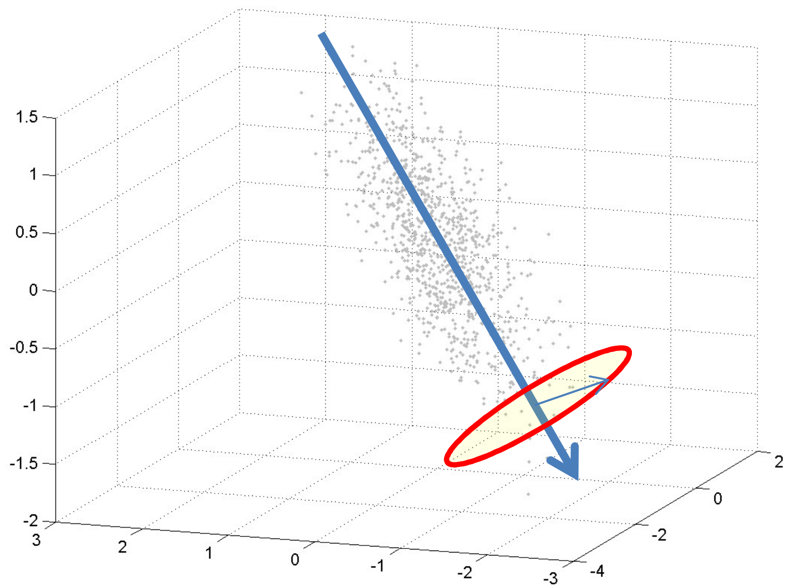
- PCA to develop the idea of LGM

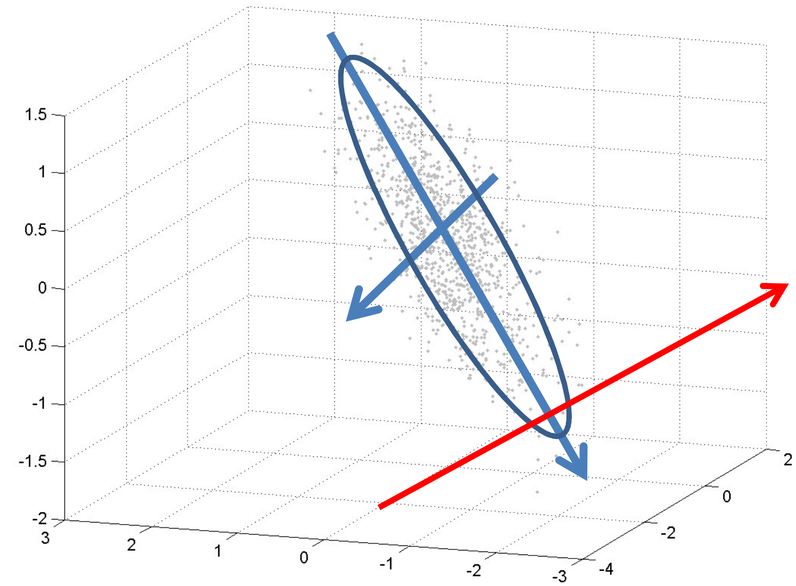
A Brief Recap



- Principal component analysis: Find the *K* bases that best explain the given data
- Find **B** and **C** such that the difference between **D** and **BC** is minimum
  - While constraining that the columns of **B** are orthonormal

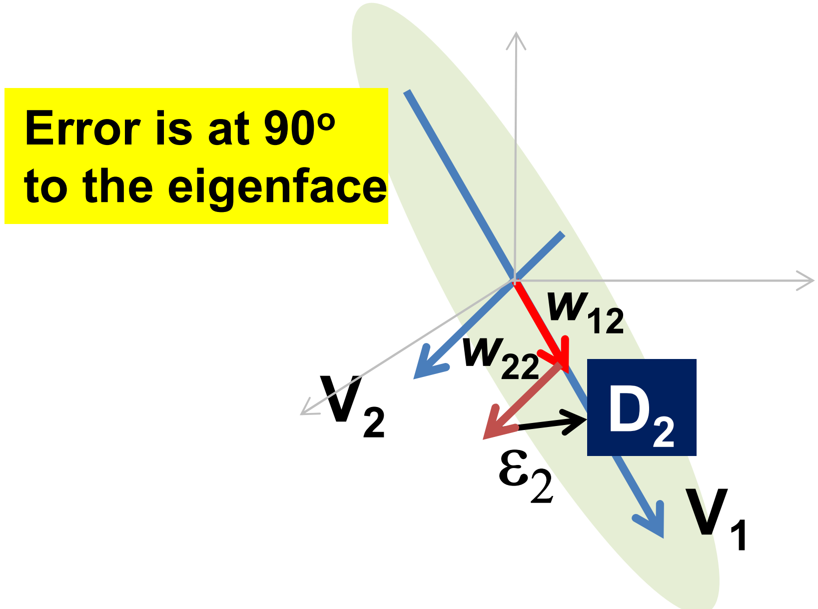
Learning PCA





- For the given data: find the K-dimensional subspace such that it captures most of the variance in the data
  - Variance in remaining subspace is minimal

A Statistical Formulation of PCA



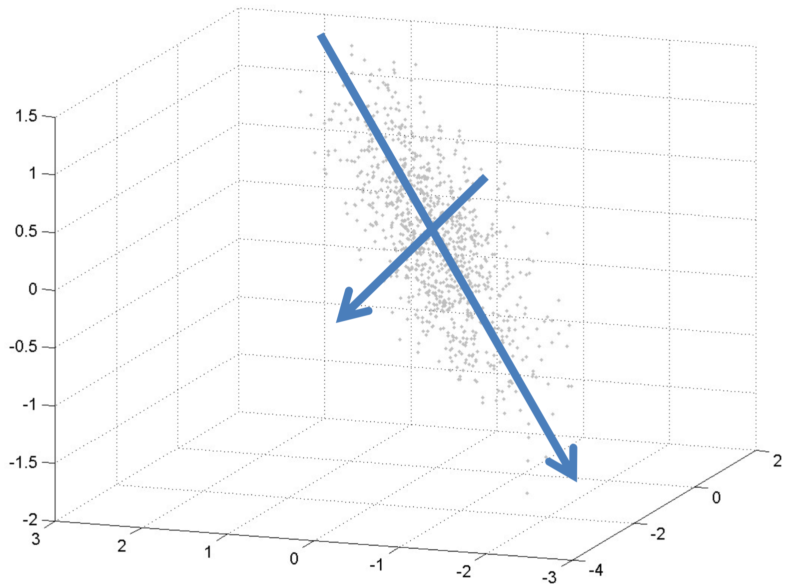
$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

$$\mathbf{w} \sim N(0, B)$$

$$\mathbf{e} \sim N(0, E)$$

- $\mathbf{x}$ is a random variable generated according to a linear relation
- $\mathbf{w}$ is drawn from an K-dimensional Gaussian with diagonal covariance
- $\mathbf{e}$ is drawn from a 0-mean (D-K)-rank D-dimensional Gaussian
- Estimate $\mathbf{V}$ (and B) given examples of $\mathbf{x}$

Linear Gaussian Models!!



$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

$$\mathbf{w} \sim N(0, B)$$

$$\mathbf{e} \sim N(0, E)$$

- $\mathbf{x}$ is a random variable generated according to a linear relation
- $\mathbf{w}$ is drawn from a Gaussian
- $\mathbf{e}$ is drawn from a 0-mean Gaussian
- Estimate $\mathbf{V}$ given examples of $\mathbf{x}$
  - In the process also estimate $\mathbf{B}$ and $\mathbf{E}$

Estimating the variables of the model

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

$$\mathbf{w} \sim N(0, I)$$
$$\mathbf{e} \sim N(0, E)$$

$$\mathbf{x} \sim N(0, \mathbf{V}\mathbf{V}^T + E)$$

- Estimating the variables of the LGM is equivalent to estimating P($\mathbf{x}$)
  - The variables are $\mathbf{V}$, and E
  - *Assuming "centered" (0-mean) data*

LGM: The complete EM algorithm

- Initialize $\mathbf{V}$ and E
- E step:

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}] = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{x}_i$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

- $$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

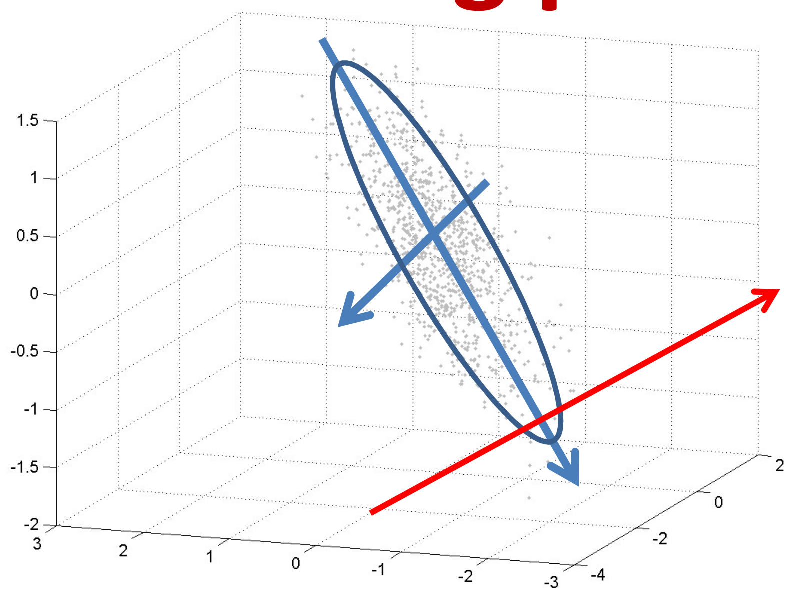
So what have we achieved

- Employed a complicated EM algorithm to learn a *Gaussian* PDF for a variable x
- What have we gained???

- Example uses:
  - PCA
    - Sensible PCA
    - EM algorithms for PCA
  - Factor Analysis
    - FA for feature extraction

LGMs : Application 1

Learning principal components



$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

$$\mathbf{w} \sim N(0, I)$$

$$\mathbf{e} \sim N(0, E)$$

- Find directions that capture most of the variation in the data
- **Error is orthogonal to principal directions**
  - $\mathbf{V}^T\mathbf{e} = 0; \ \mathbf{e}^T\mathbf{V} = 0$

Some Observations: 1

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{e} \sim N(0, E)$$

$$E = \mathrm{E}[\mathbf{e}\mathbf{e}^T]$$

$$\mathbf{V}^T E = \mathrm{E}[\mathbf{V}^T\mathbf{e}\mathbf{e}^T] = \mathrm{E}[0\mathbf{e}^T] = 0$$

- The covariance $\mathbf{E}$ of $\mathbf{e}$ is orthogonal to $\mathbf{V}$
  - $\mathbf{V}$ is in the null space of $\mathbf{E}$

Observation 2

$$\mathbf{V}^T E = 0$$

$$\mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1} = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$$

- Proof

$$\mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}(\mathbf{V}\mathbf{V}^T + E) = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)$$

$$\mathbf{V}^T = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\mathbf{V}\mathbf{V}^T + (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T E$$

$$\mathbf{V}^T = \mathbf{I}\mathbf{V}^T + (\mathbf{V}^T\mathbf{V})^{-1}0$$

$$\mathbf{V}^T = \mathbf{V}^T$$

Observation 3

$$\mathbf{V}^T E = 0$$

$$\mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1} = (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$$

$$= pinv(\mathbf{V})$$

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

MLSP

LGM: The complete EM algorithm

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{X} \approx \mathbf{V}\mathbf{W}$$

- Initialize $\mathbf{V}$ and E
- E step: $E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}] = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{x}_i$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

LGM: The complete EM algorithm

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{X} \approx \mathbf{V}\mathbf{W}$$

- Initialize $\mathbf{V}$ and E
- E step: $$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}] = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{x}_i$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{X} \approx \mathbf{V}\mathbf{W}$$

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{x}_i = pinv(\mathbf{V})\mathbf{x}_i$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{X} \approx \mathbf{V}\mathbf{W}$$

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

$$\mathbf{X} \approx \mathbf{V}\mathbf{W}$$

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i \qquad \mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

$$\mathbf{X} \approx \mathbf{V}\mathbf{W}$$

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i \qquad \mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i \qquad \mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i$$

$$\mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1} = \mathbf{X}\mathbf{W}^T(\mathbf{W}\mathbf{W}^T)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i \qquad \mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1} = \mathbf{X}\mathbf{W}^T(\mathbf{W}\mathbf{W}^T)^{-1} = \mathbf{X}pinv(\mathbf{W})$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{w}_i = pinv(\mathbf{V})\mathbf{x}_i \qquad \mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \mathbf{X}\, pinv(\mathbf{W})$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \mathbf{X}\, pinv(\mathbf{W})$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

- Initialize $\mathbf{V}$ and E
- E step:

$$\mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - V^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

irrelevant

$$\mathbf{V} = \mathbf{X}\, pinv(\mathbf{W})$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

EM for PCA

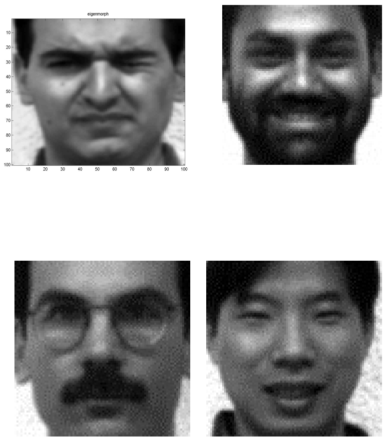
- Initialize $\mathbf{V}$
- Iterate

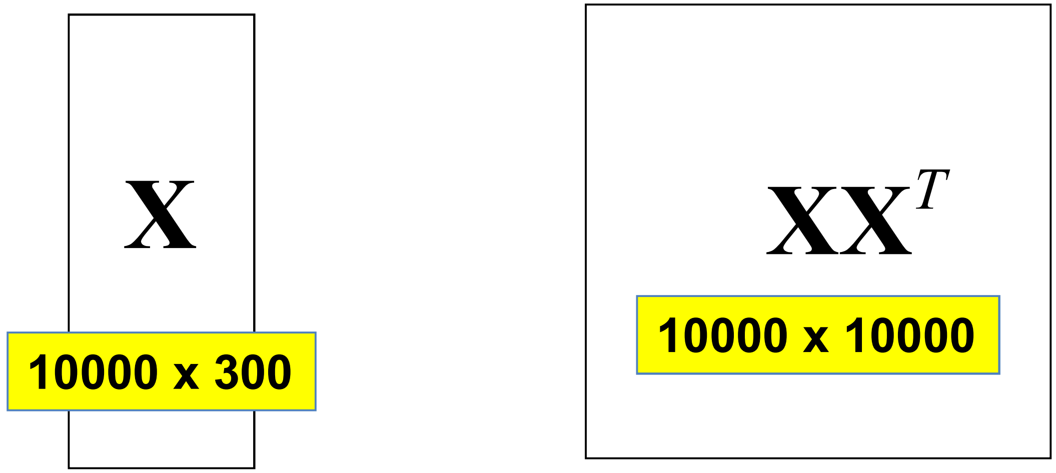
$$\mathbf{W} = pinv(\mathbf{V})\mathbf{X}$$

$$\mathbf{V} = \mathbf{X}\, pinv(\mathbf{W})$$

- Note: $\mathbf{V}$ will not be actual eigenvectors, but a set of bases in space spanned by principal eigenvectors
  – Additional decorrelation within PC space may be needed

Why EM PCA?



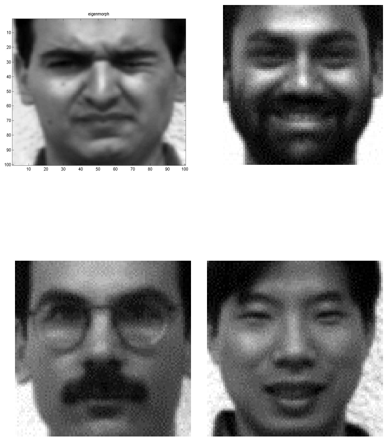


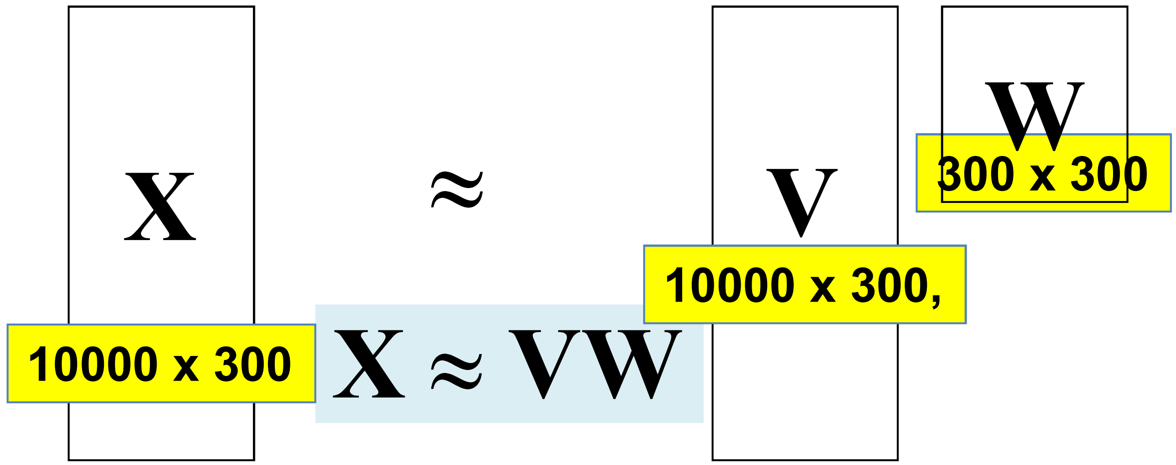
- Example: Computing eigenfaces
- Each face is 100x100 : 10000 dimensional
- But only 300 examples
  - $\mathbf{X}$ is 10000 x 300
- What is the size of the covariance matrix?
- What is its rank?

PCA on illconditioned data

- Few instances of high-dimensional data
  - No. instances < dimensionality
- Covariance matrix is very large
  - Eigen decomposition is expensive
  - E.g. 1000000-dimensional data: Covariance has 10^{12} elements
- But the rank of the covariance is low
  - Only the no. of instances of data

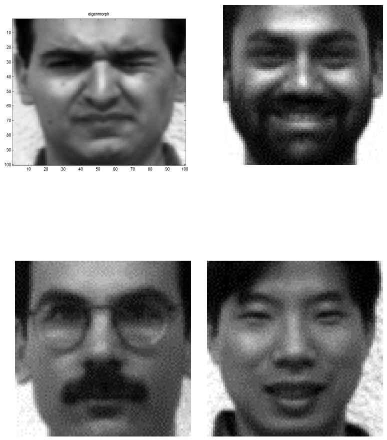
Why EM PCA?

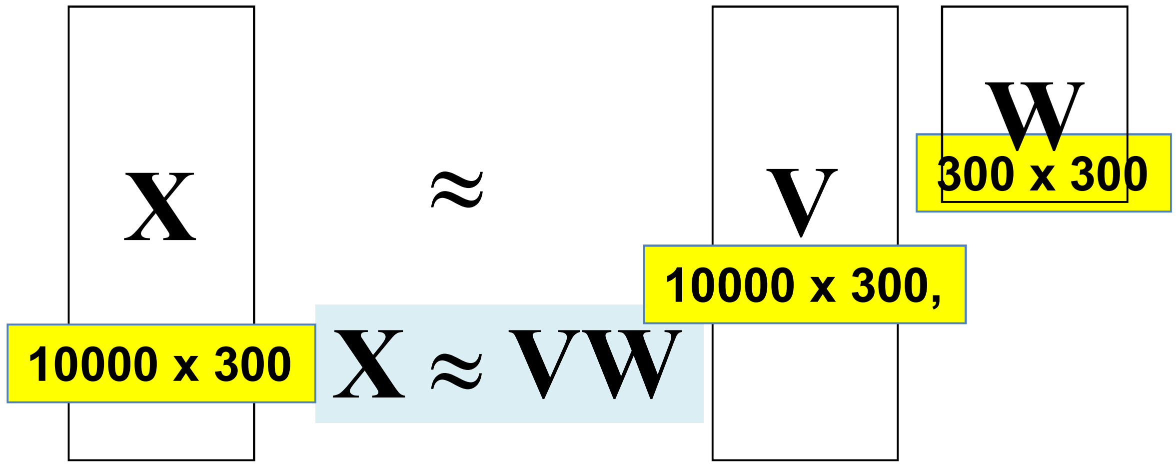




- Consequence of low rank $\mathbf{X}$
  - The actual number of bases is limited to the rank of $\mathbf{X}$
- Note actual size of $\mathbf{V}$
  - Max number of columns = min(dimension, no. data points)
  - No. of columns = rank of ($\mathbf{X}\mathbf{X}^T$)
- Note size of $\mathbf{W}$
  - Max number of rows = min(dimension, no. of data points)

Why EM PCA?





- If **X** is high dimensional
  - Particularly if the number of vectors in **X** is smaller than the dimensionality

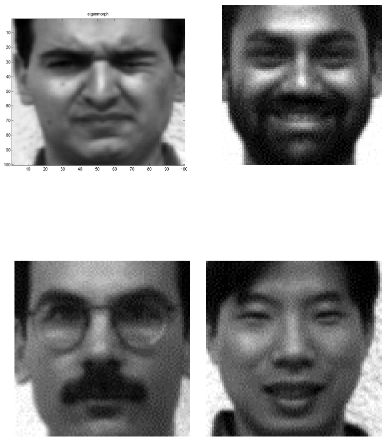
- Pinv(**V**) and pinv(**W**) are efficient to compute
  - **V** will have a max of 300 columns in the example
  - **W** will have a max of 300 rows

PCA as an instance of LGM

- Viewing PCA as an instance of linear Gaussian models leads to EM solution
- Very effective in dealing with high-dimensional and/or data poor situations

- An aside: Another simpler solution for the same situation..

An Aside: The GRAM trick



$\mathbf{X}$

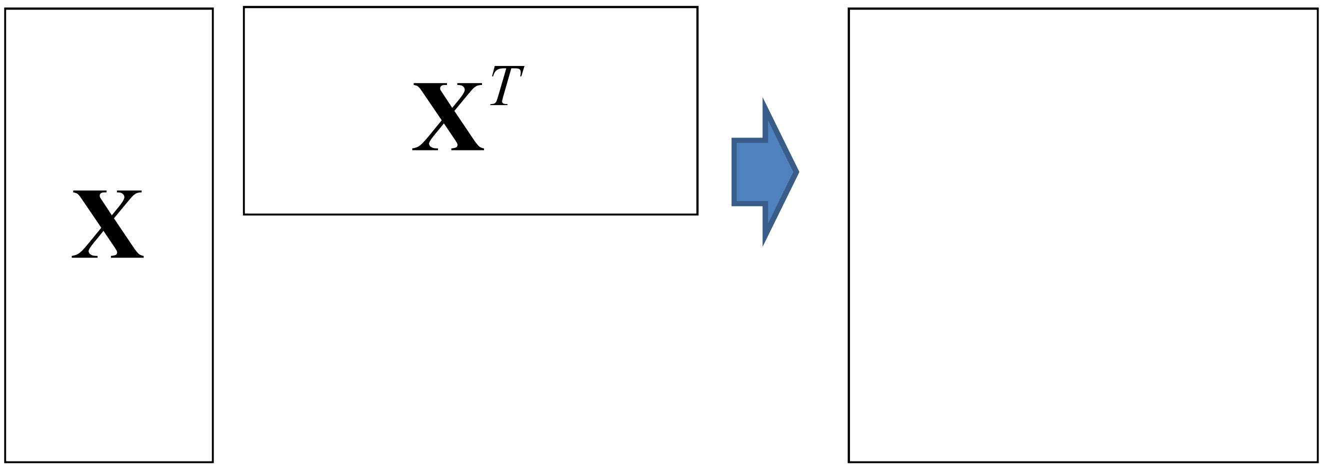
10000 x 300

$\mathbf{XX}^T$

10000 x 10000

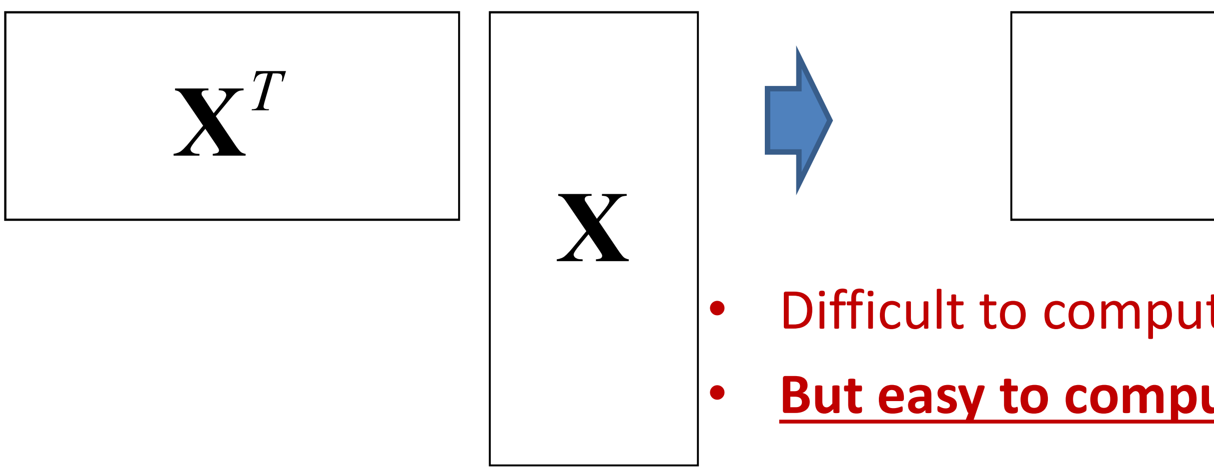
- The number of non-zero Eigen values is no more than the length of the smallest "edge" of **X**
  - 300 in this case
- This leads to the "gram" trick..
- Assumption $\mathbf{X}^T\mathbf{X}$ is invertible: the instances are linearly independent

An Aside: The GRAM trick



If X is 10000 x 300,
XX^T = 10000 x 10000

- $\mathbf{XX^T}$ is large but $\mathbf{X^TX}$ is not



If X is 10000 x 300,
X^TX = 300 x 300

- Difficult to compute Eigen vectors of $\mathbf{XX^T}$
- **But easy to compute Eigen vectors of $\mathbf{X^TX}$**

The Gram Trick

- To compute principal vectors we Eigendecompose $\mathbf{X}\mathbf{X}^T$

$$\left(\mathbf{X}\mathbf{X}^T\right)\mathbf{E} = \mathbf{E}\Lambda$$

- Let us find the Eigen vectors of $\mathbf{X}^T\mathbf{X}$ instead

$$\left(\mathbf{X}^T\mathbf{X}\right)\hat{\mathbf{E}} = \hat{\mathbf{E}}\hat{\Lambda}$$

- Manipulating it slightly

Note that for a diagonal matrix:
$\Lambda\Lambda^{-0.5} = \Lambda^{-0.5}\Lambda$

$$\mathbf{X}^T\mathbf{X}\hat{\mathbf{E}}\hat{\Lambda}^{-0.5} = \hat{\mathbf{E}}\hat{\Lambda}^{-0.5}\hat{\Lambda}$$

The Gram Trick

- Eigendecompose $\mathbf{X}^T\mathbf{X}$ instead of $\mathbf{X}\mathbf{X}^T$

$$\left(\mathbf{X}^T\mathbf{X}\right)\hat{\mathbf{E}} = \hat{\mathbf{E}}\hat{\Lambda}$$

$$\mathbf{X}^T\mathbf{X}\hat{\mathbf{E}}\hat{\Lambda}^{-0.5} = \hat{\mathbf{E}}\hat{\Lambda}^{-0.5}\hat{\Lambda}$$

$$\left(\mathbf{X}\mathbf{X}^T\right)\left(\mathbf{X}\hat{\mathbf{E}}\hat{\Lambda}^{-0.5}\right) = \left(\mathbf{X}\hat{\mathbf{E}}\hat{\Lambda}^{-0.5}\right)\hat{\Lambda}$$

- Letting: $\mathbf{X}\hat{\mathbf{E}}\hat{\Lambda}^{-0.5} = \mathbf{E}$

$$\left(\mathbf{X}\mathbf{X}^T\right)\mathbf{E} = \mathbf{E}\hat{\Lambda}$$

- E is the matrix of Eigenvectors of $\mathbf{X}\mathbf{X}^T$**!!!**

The Gram Trick

- **When X is low rank or XX^T is too large:**
- Compute $\mathbf{X}^T\mathbf{X}$ instead
  - Will be manageable size
- Perform Eigen Decomposition of $\mathbf{X}^T\mathbf{X}$

$$\left(\mathbf{X}^T\mathbf{X}\right)\hat{\mathbf{E}} = \hat{\mathbf{E}}\hat{\Lambda}$$

- **Compute Eigenvectors of XX^T as**

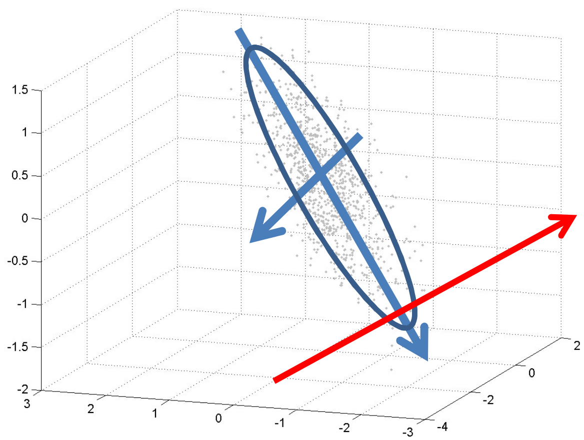
$$\mathbf{X}\hat{\mathbf{E}}\hat{\Lambda}^{-0.5} = \mathbf{E}$$

- **These are the principal components of X**

Why EM PCA

- Dimensionality / Rank has alternate potential solution
  - Gram Trick
- Other uses?
  - Noise
  - Incomplete data

PCA with noisy data



$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} + \mathbf{n}$$

$$\mathbf{w} \sim N(0, I)$$

$$\mathbf{e} \sim N(0, E)$$

$$\mathbf{n} \sim N(0, B)$$

- Error is orthogonal to principal directions
  - $\mathbf{V}^T\mathbf{e} = \mathbf{0}; \; \mathbf{e}^T\mathbf{V} = \mathbf{0}$
- Noise is isotropic
  - *B* is diagonal
  - Noise is not orthogonal to either **V** or **e**

LGM: The complete EM algorithm

- Initialize $\mathbf{V}$ and E
- E step:

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}] = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{x}_i$$

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

- M step:

$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T$$

PCA with Noisy Data

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} + \mathbf{n}$$
$$\mathbf{w} \sim N(0, I)$$
$$\mathbf{e} \sim N(0, E)$$
$$\mathbf{n} \sim N(0, B)$$

- Initialize $\mathbf{V}$ and B
- E step:

$$\beta = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + B)^{-1}$$

$$\mathbf{W} = \beta\mathbf{X}$$

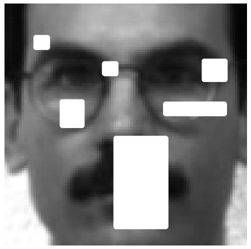
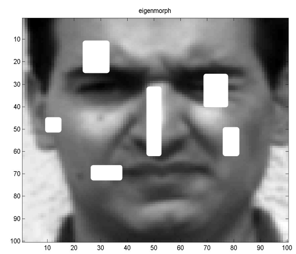
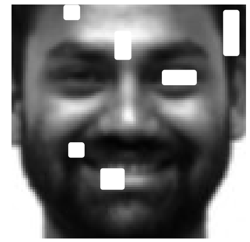
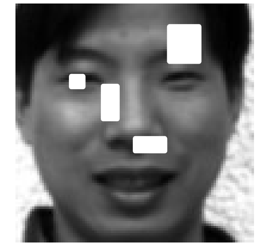
$$\mathbf{C} = NI - N\beta\mathbf{V} + \mathbf{W}\mathbf{W}^T$$

- M step:

$$\mathbf{V} = \mathbf{X}\mathbf{W}^T\mathbf{C}^{-1}$$

$$B = \frac{1}{N} diag\left(\mathbf{X}\mathbf{X}^T - \mathbf{V}\mathbf{W}\mathbf{X}^T\right)$$

PCA with *Incomplete* Data

- How to compute principal directions when some components in your training data are missing?

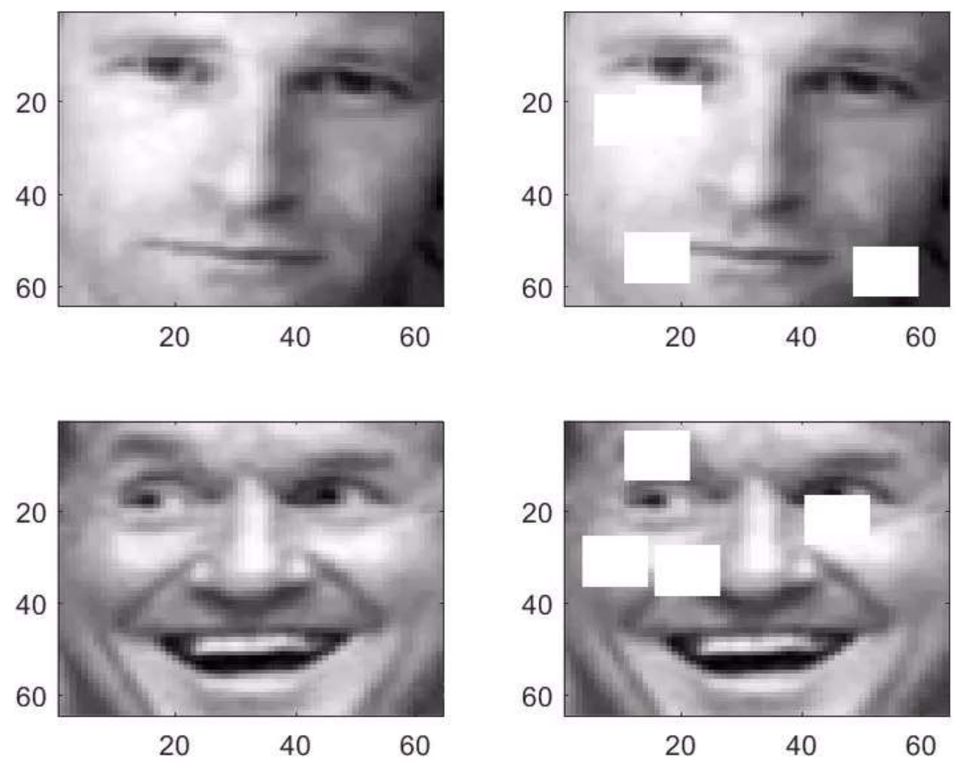
- Eigen decomposition is not possible
  - Cannot compute correlation matrix with missing data

PCA with missing data

- How it goes
- Given : $\mathbf{X} = \{\mathbf{X}_c, \mathbf{X}_m\}$
  - $\mathbf{X}_\mathbf{m}$ are missing components

1. Initialize: Initialize $\mathbf{X}_m$
2. Build "complete" data $\mathbf{X} = \{\mathbf{X}_c, \mathbf{X}_m\}$
3. PCA ($\mathbf{X} = \mathbf{VW}$): Estimate $\mathbf{V}$
   - $\mathbf{V}$ must have fewer bases than dimensions of $\mathbf{X}$
4. $\mathbf{W} = \mathbf{V}^\mathbf{T}\mathbf{X}$
5. $\hat{\mathbf{X}} = \mathbf{VW}$
6. Select $\mathbf{X}_\mathbf{m}$ from $\hat{\mathbf{X}}$
7. Return to 2

Data imputation example



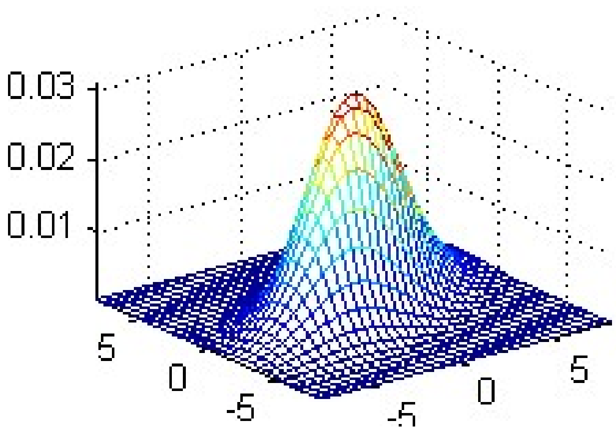
- Filling in holes in facial images
- Using a large number of face images, *all of which have holes*
- PCA will simultaneously "fix" all of them

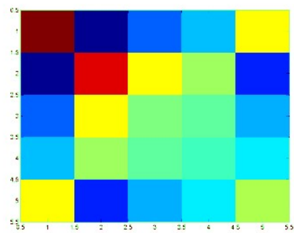
LGM for PCA

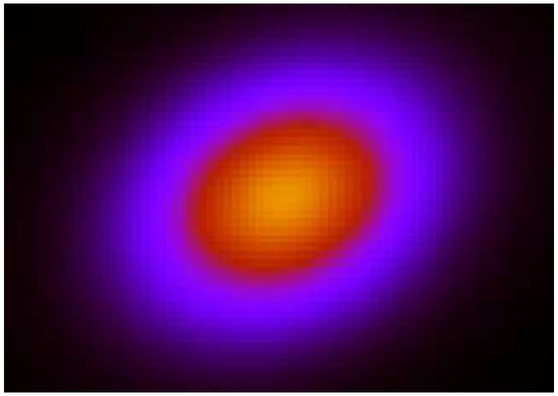
- Obviously many uses:
  - Ill-conditioned data
  - Noise
  - Missing data

  - Any combination of the above..

LGMs : Application 2
Learning with insufficient data

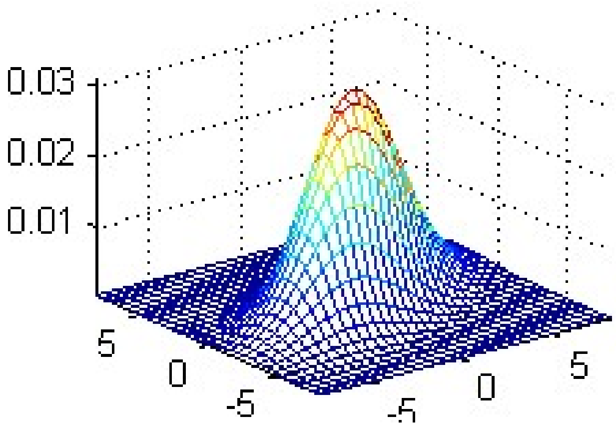


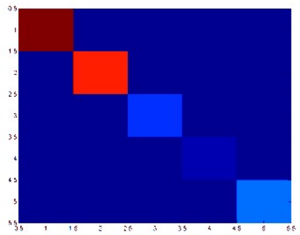


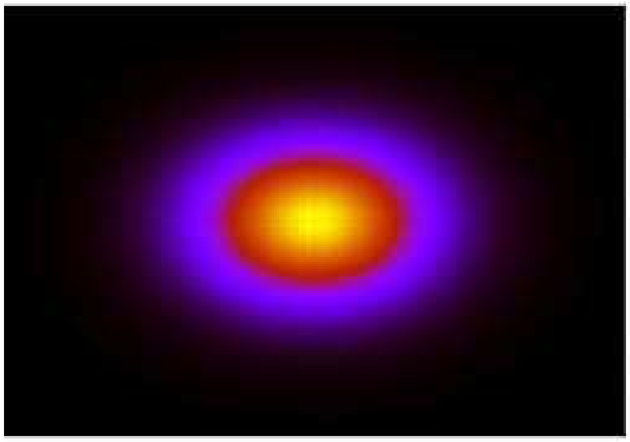


- The full covariance matrix of a Gaussian has D^2 terms
- Fully captures the relationships between variables
- Problem: **Needs a lot of data to estimate robustly**

An Approximation







- Assume the covariance is diagonal
  - Gaussian is aligned to axes : no correlation between dimensions
  - Covariance has only *D* terms
- **Needs less data**
- **Problem : Model loses all information about correlation between dimensions**

Is There an Intermediate

- Capture the most important correlations
- But require less data

- Solution: Find the key subspaces in the data
  - Capture the complete correlations in these subspaces
  - Assume data is otherwise uncorrelated

Factor Analysis

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

$$\mathbf{w} \sim N(0, I)$$
$$\mathbf{e} \sim N(0, E)$$

$$\mathbf{x} \sim N(0, \mathbf{V}\mathbf{V}^T + E)$$

- *E* is a full rank diagonal matrix
- **V** has *K* columns: K-dimensional subspace
  - We will capture all the correlations in the subspace represented by **V**
- Estimated covariance: Diagonal covariance *E* plus the covariance between dimensions in **V**

Factor Analysis

- Initialize $\mathbf{V}$ and E
- E step:

$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}] = \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{x}_i$$

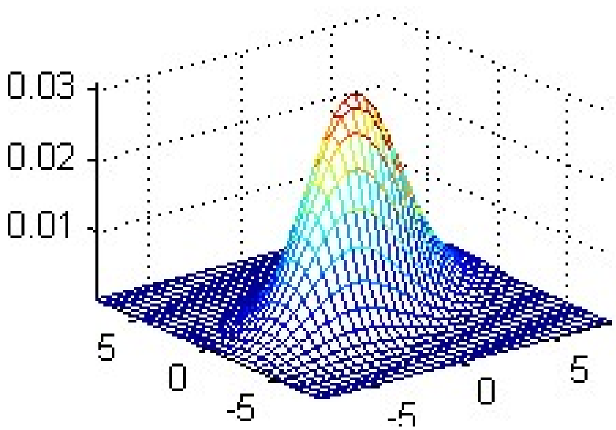
$$E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T] = I - \mathbf{V}^T(\mathbf{V}\mathbf{V}^T + E)^{-1}\mathbf{V} + E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]^T$$

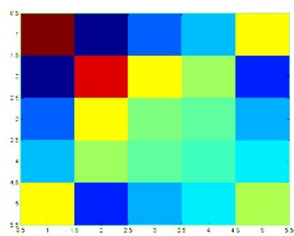
- M step:

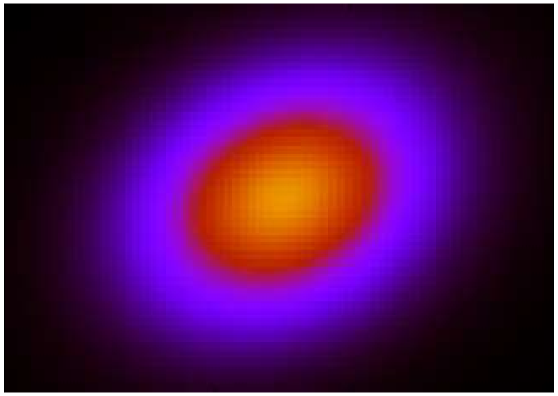
$$\mathbf{V} = \left(\sum_i \mathbf{x}_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}^T]\right)\left(\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}\mathbf{w}^T]\right)^{-1}$$

$$E = \frac{1}{N} diag\left(\sum_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{N}\mathbf{V}\sum_i E_{\mathbf{w}|\mathbf{x}_i}[\mathbf{w}]\mathbf{x}_i^T\right)$$

FA Gaussian







- Will get a full covariance matrix
- But only estimate DK terms
- Data insufficiency less of a problem

The Factor Analysis Model

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

LOADINGS FACTORS

$$\mathbf{w} \sim N(0, I)$$
$$\mathbf{e} \sim N(0, E)$$

- Often used to learn distribution of data when we have insufficient data
- Often used in psychometrics
  - Underlying model: The actual systematic variations in the data are totally explained by a small number of “factors”
  - FA uncovers these factors

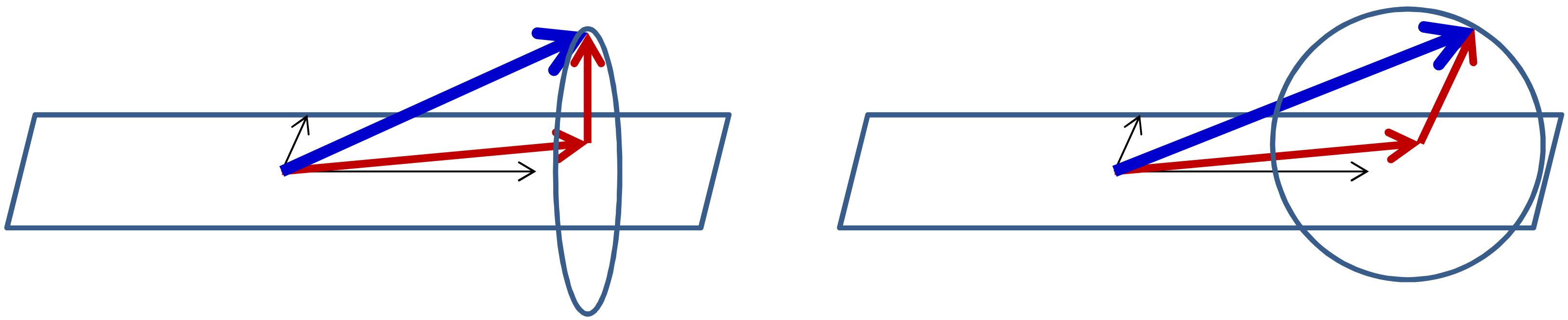
FA, PCA etc.

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e}$$

$$\mathbf{w} \sim N(0, I)$$
$$\mathbf{e} \sim N(0, E)$$

- Note: distinction between PCA and FA is only in the assumptions about **e**
- FA looks a lot like PCA with noise
- FA can also be performed with incomplete data

FA, PCA etc.

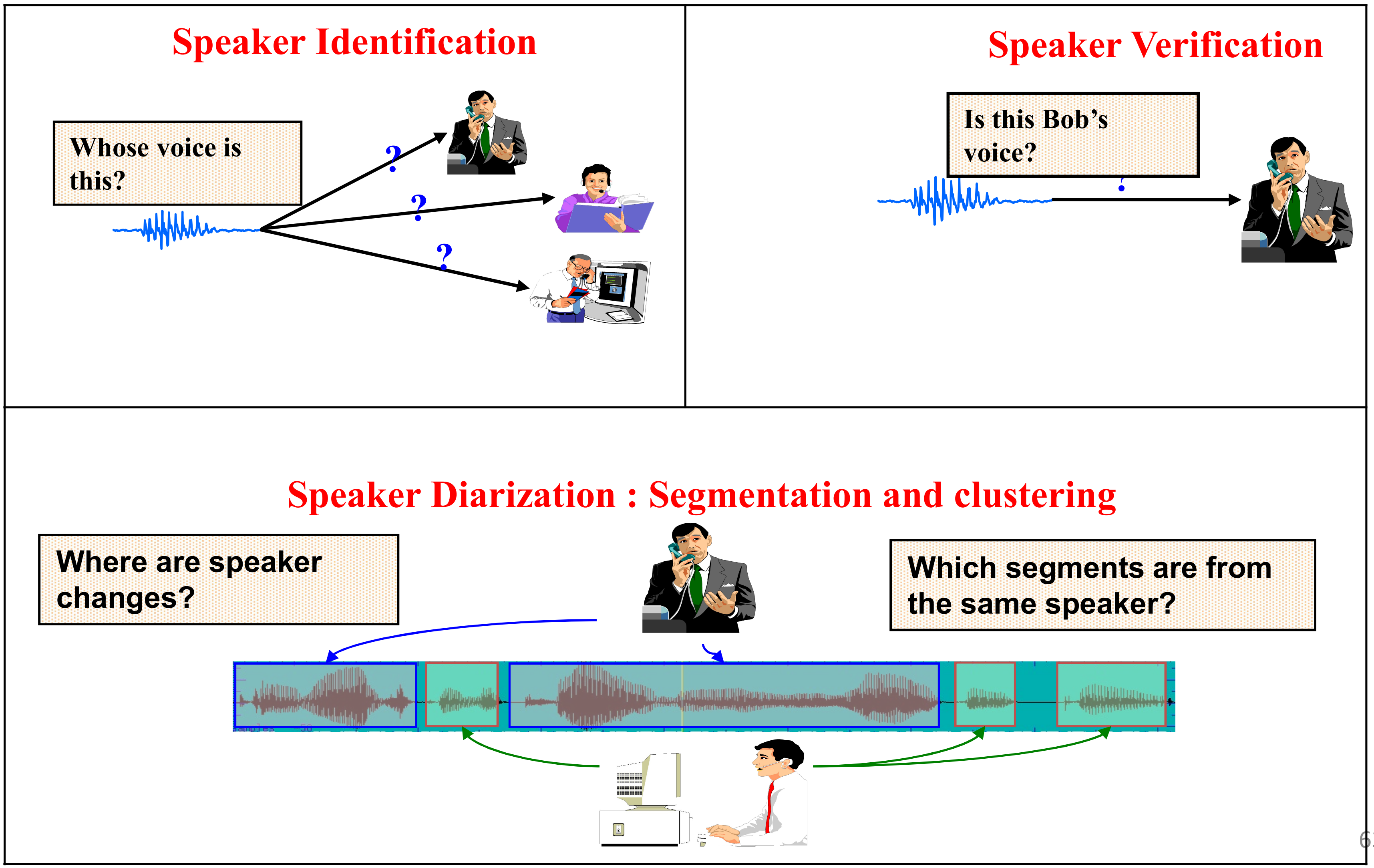


- PCA: Error is always at 90 degrees to the bases in $\mathbf{V}$
- FA: Error may be at any angle
- PCA used mainly to find *principal* directions that capture most of the variance
  - Bases in V will be orthogonal to one another
- FA tries to capture most of the covariance

FA: A very successful use

- Voice biometrics: Speaker recognition
- Given: Only a small amount of training data from a speaker to learn its model
  - Use to verify speaker later
- Problem: Immense variation in ways people can speak
  - Less than 1 minute of training data; totally insufficient!

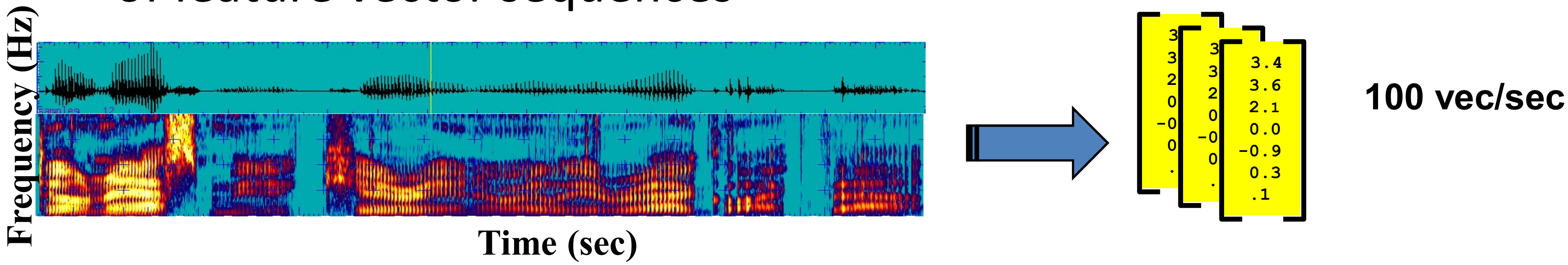
Speaker Recognition



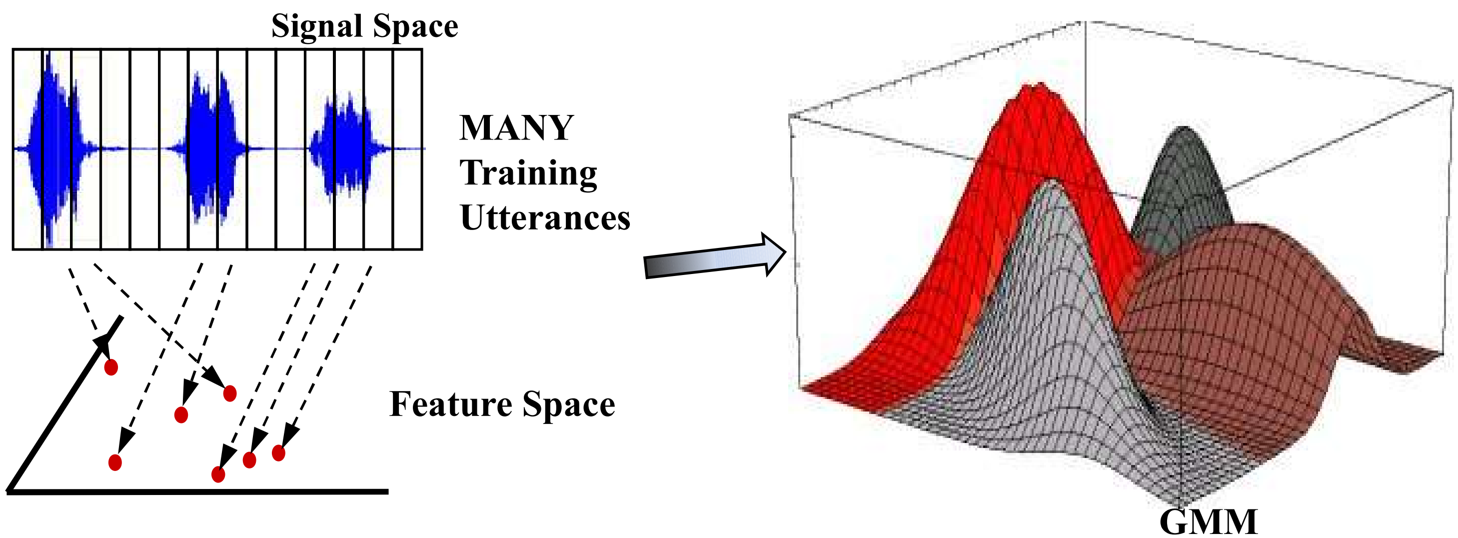
Modeling Sequence of Features

Gaussian Mixture Models

- For most recognition tasks, we need to model the distribution of feature vector sequences



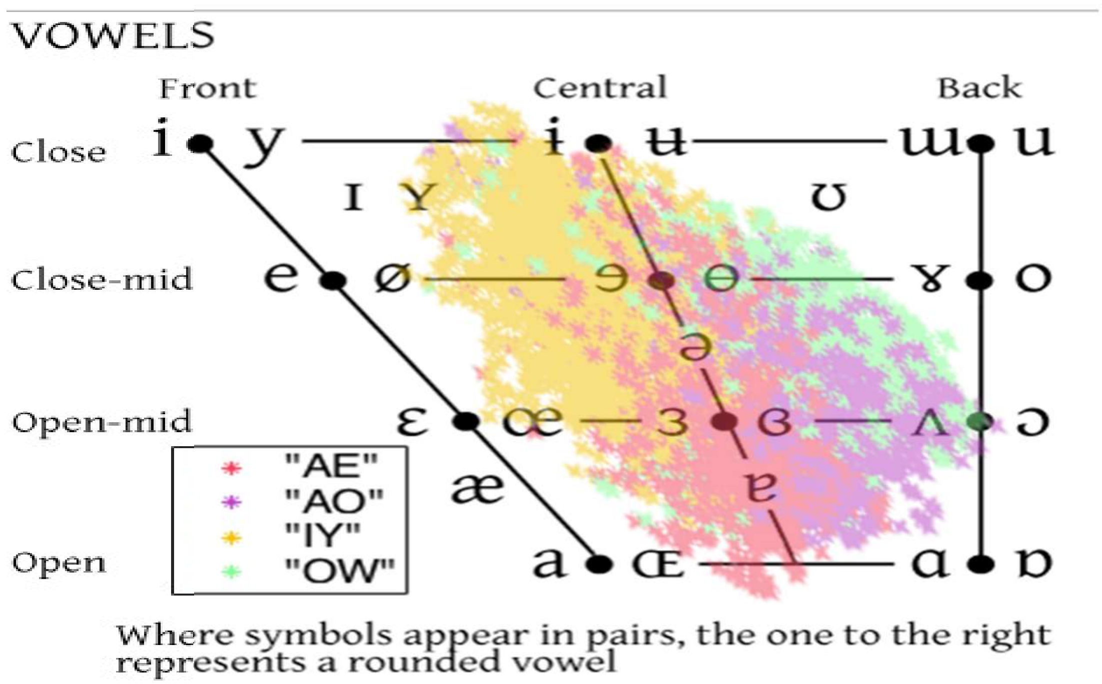
- **In practice, we often use Gaussian Mixture Models (GMMs)**



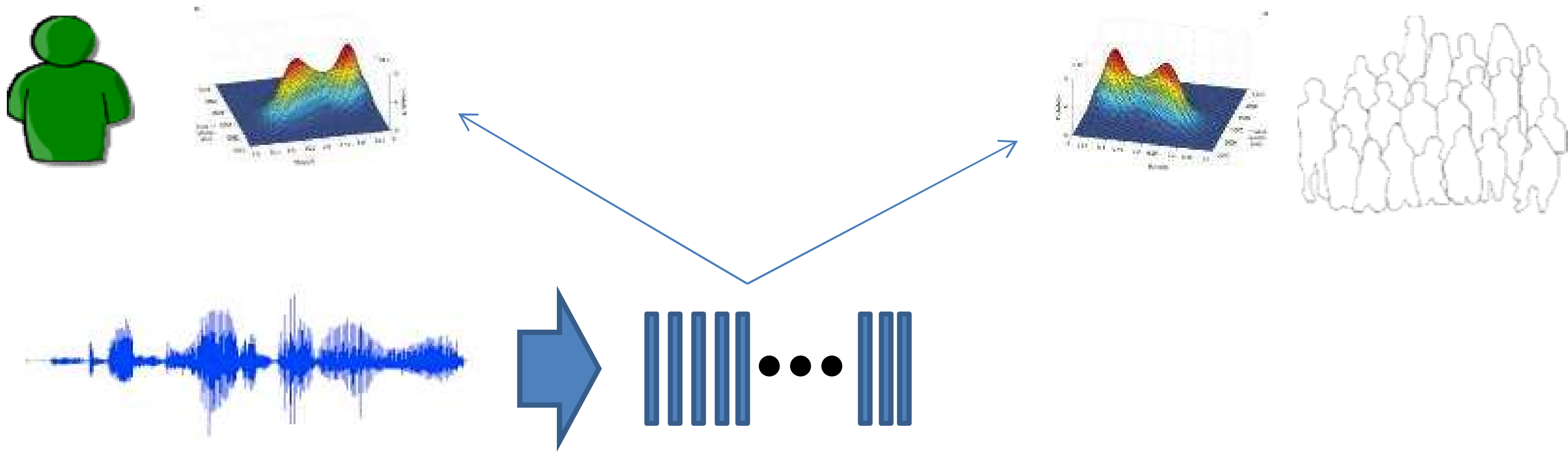
Why GMMs

- Vowel Classification

PCA



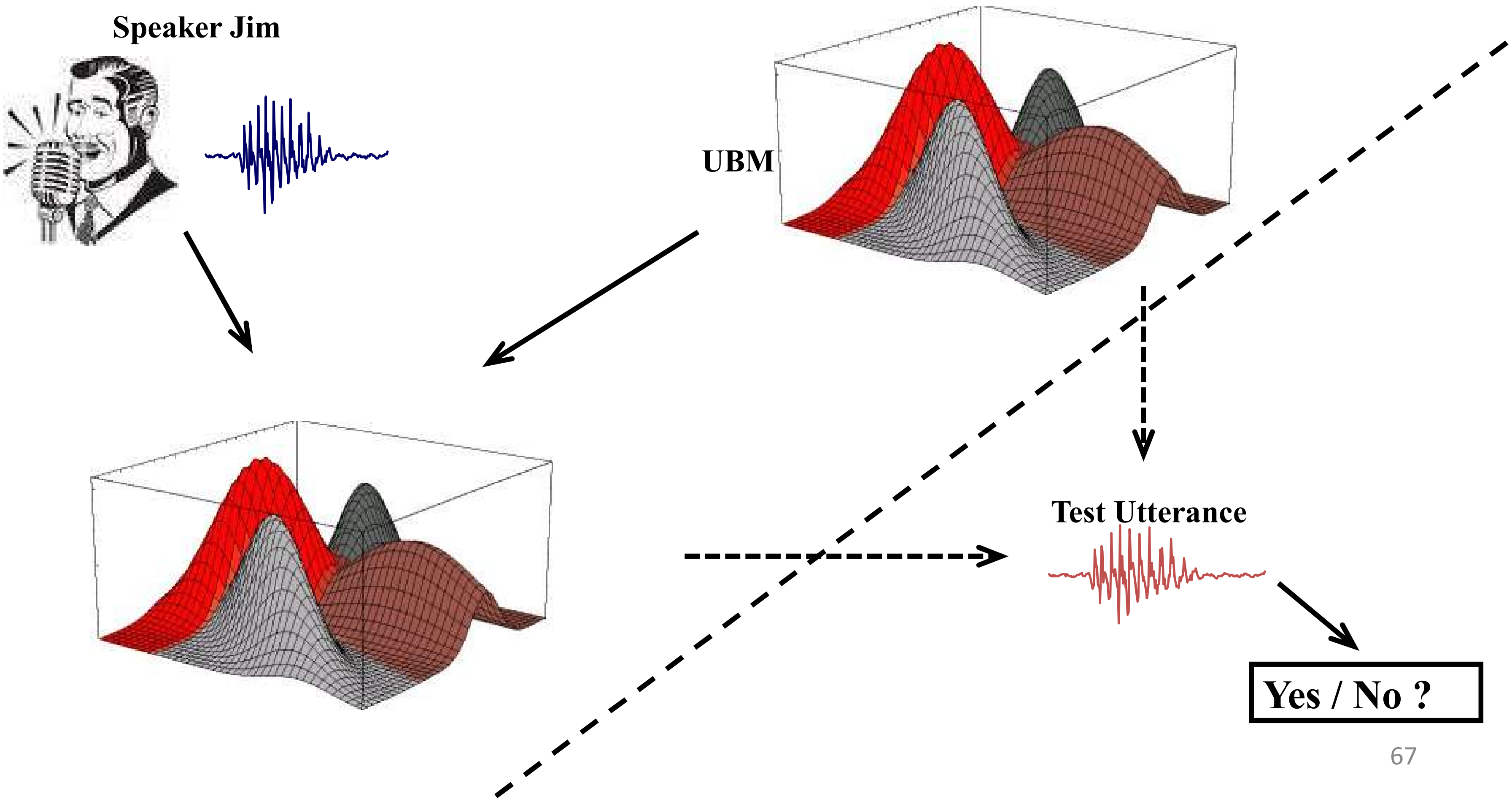
Speaker Verification



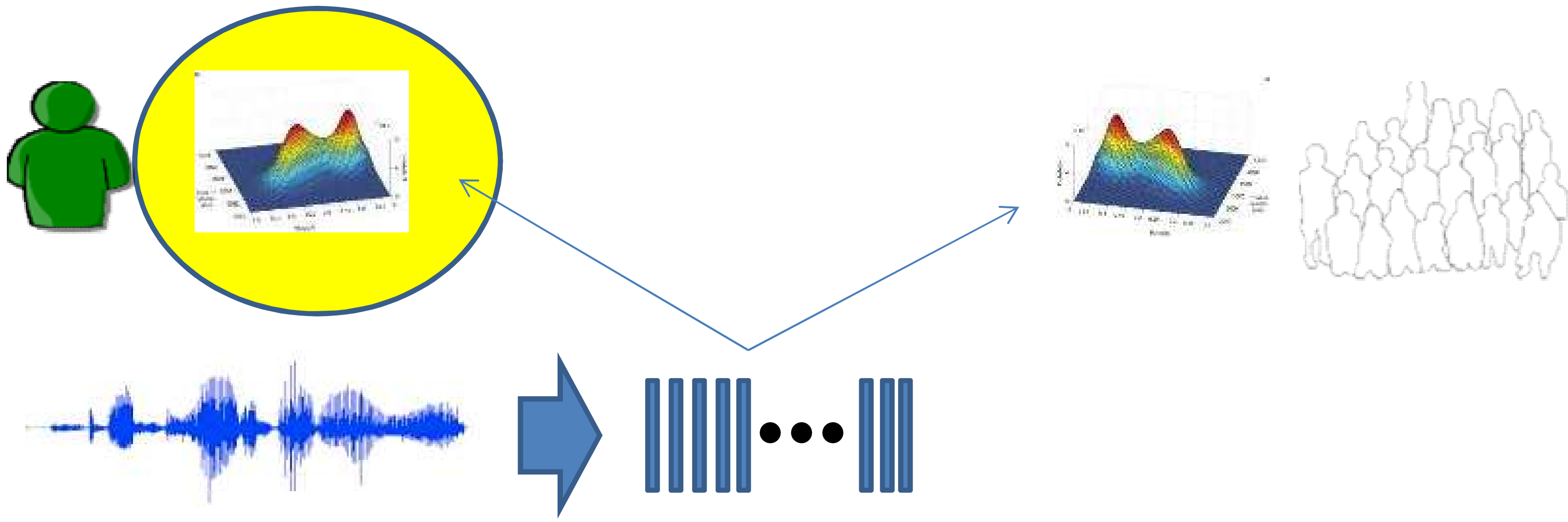
- A model represents distribution of cepstral vectors for the speaker
- A second model represents everyone else (potential imposters)
- The cepstra computed from a test recording are "scored" against both models
  - Accept the speaker if the speaker model scores higher

GMM for speaker verification

- **We enroll a given speaker by adapting the UBM using the speaker's input speech. [Reynolds 2000]**



Speaker Verification

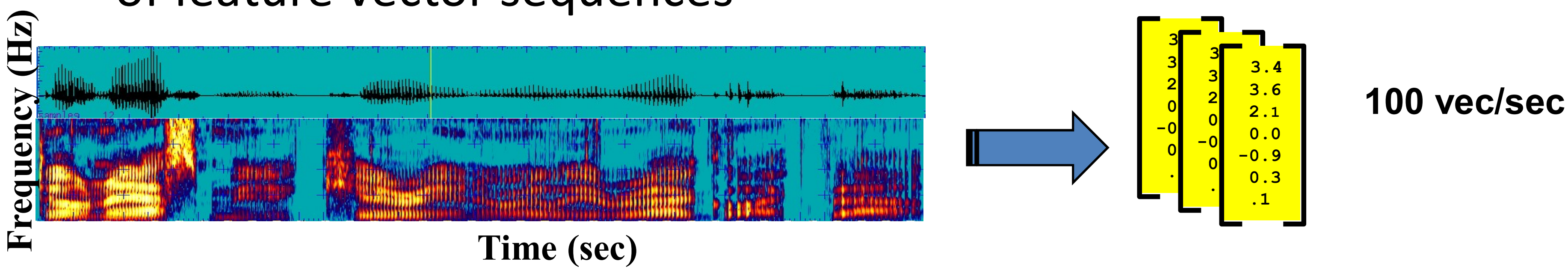


- Problem: One typically has only a few seconds or minutes of training data from the speaker
- Hard to estimate speaker model
- Test data may be spoken differently, or come over a different channel, or in noise
  - Wont really match

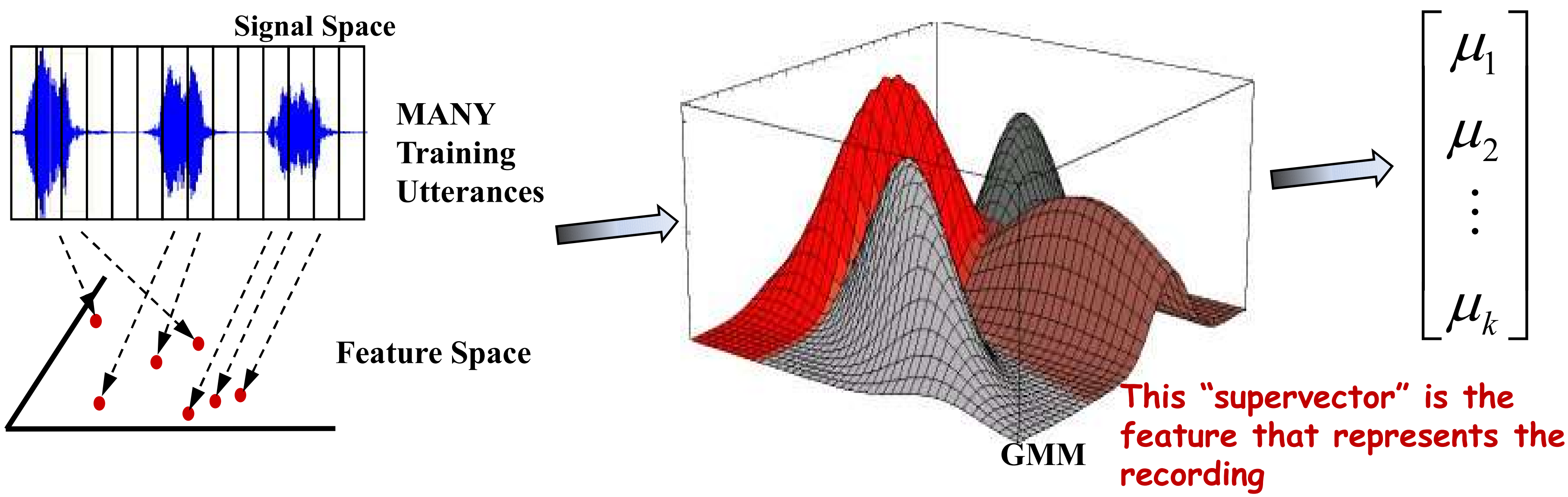
Modeling Sequence of Features

Gaussian Mixture Models

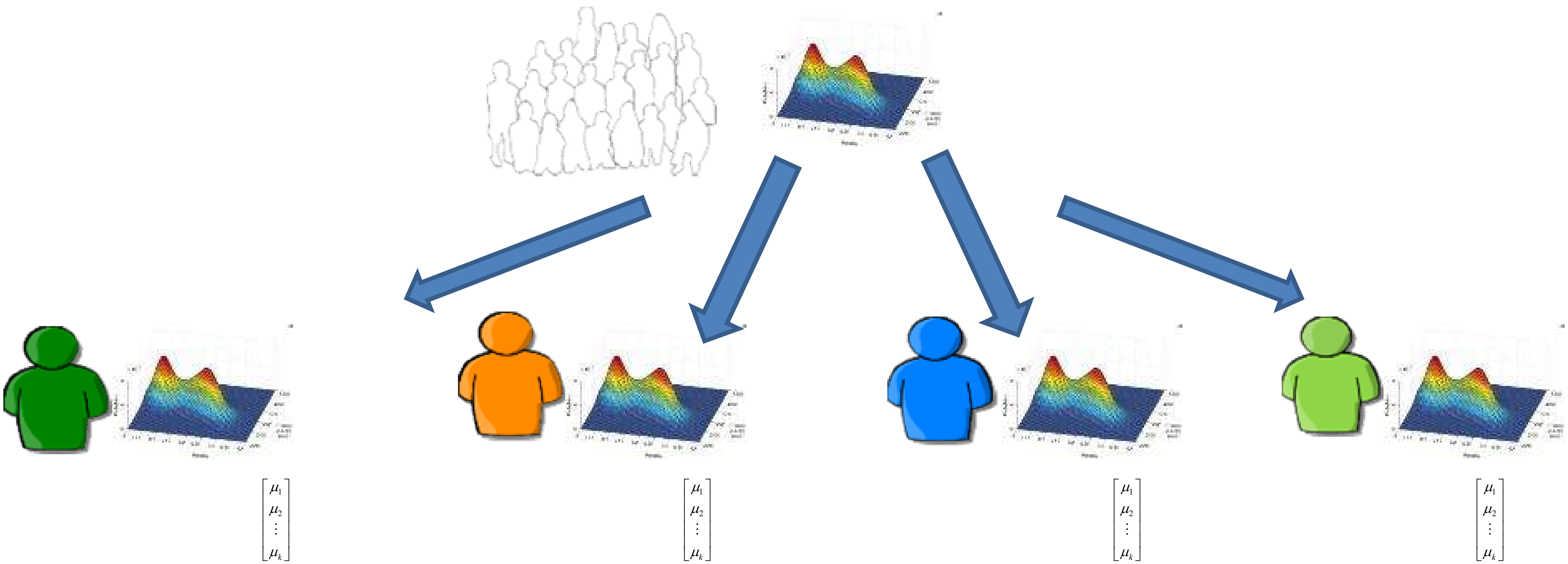
- For most recognition tasks, we need to model the distribution of feature vector sequences



- **In practice, we often use Gaussian Mixture Models (GMMs)**



Training



- Supervectors are obtained for each training speaker by adapting a “Universal background model” trained from large amounts of data
  - Few data by each speaker to train a GMM based on Maximum likelihood

Training the Factor Analyzer

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{w} \sim N(0, I) \quad \mathbf{e} \sim N(0, E)$$

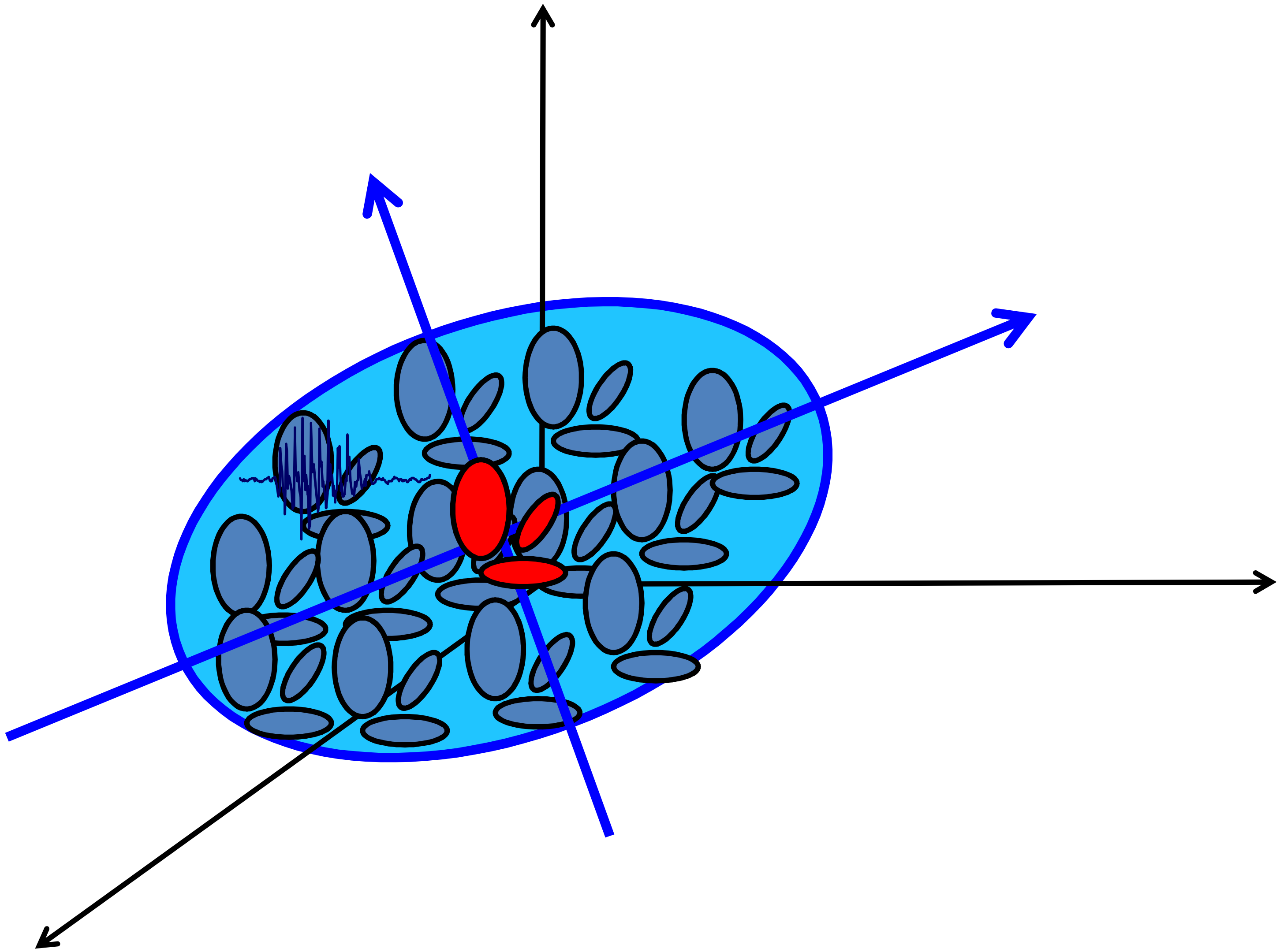
- The supervectors are assumed to be the output of a linear Gaussian process
- Use FA to estimate $\mathbf{V}$
  - $\mathbf{V}$ are the directions of main variations
  - The *real* information is in the factor $\mathbf{w}$

Identification

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{w} \sim N(0, I) \quad \mathbf{e} \sim N(0, E)$$

- Enrollment: Derive one or more $\mathbf{w}_{spkr}$ vectors from speaker recordings
  - Using $\mathbf{V}$ and E learned during the "training phase"
  - Also use $\mathbf{w}_{imposter}$ from recordings from other speakers to train a binary classifier, e.g. an SVM
- Verification: Derive $\mathbf{w}_{verif}$ from test recording
  - Classify using SVM
  - Alternately, compare to $\mathbf{w}_{spkr}$ vectors from enrollment recordings

I-vector : Total variability space

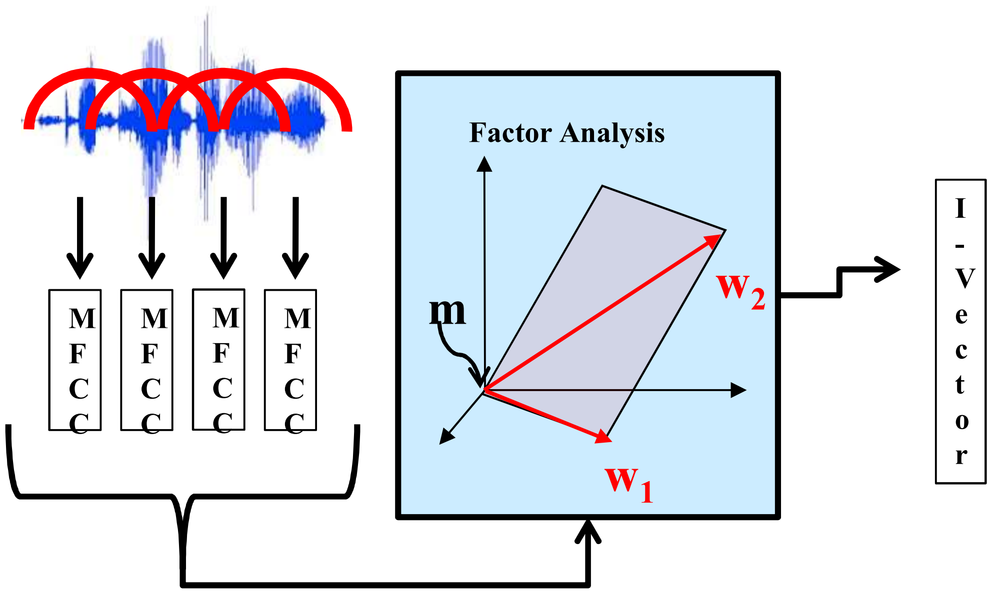


I-Vector

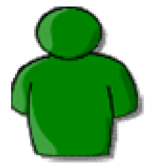
- Factor analysis as feature extractor
- Speaker and channel dependent supervector

$$\mathbf{M} = \boldsymbol{m} + \boldsymbol{T}\boldsymbol{w}$$

  - T is rectangular, low rank (total variability matrix)
  - w standard Normal random (total factors – intermediate vector or i-vector)



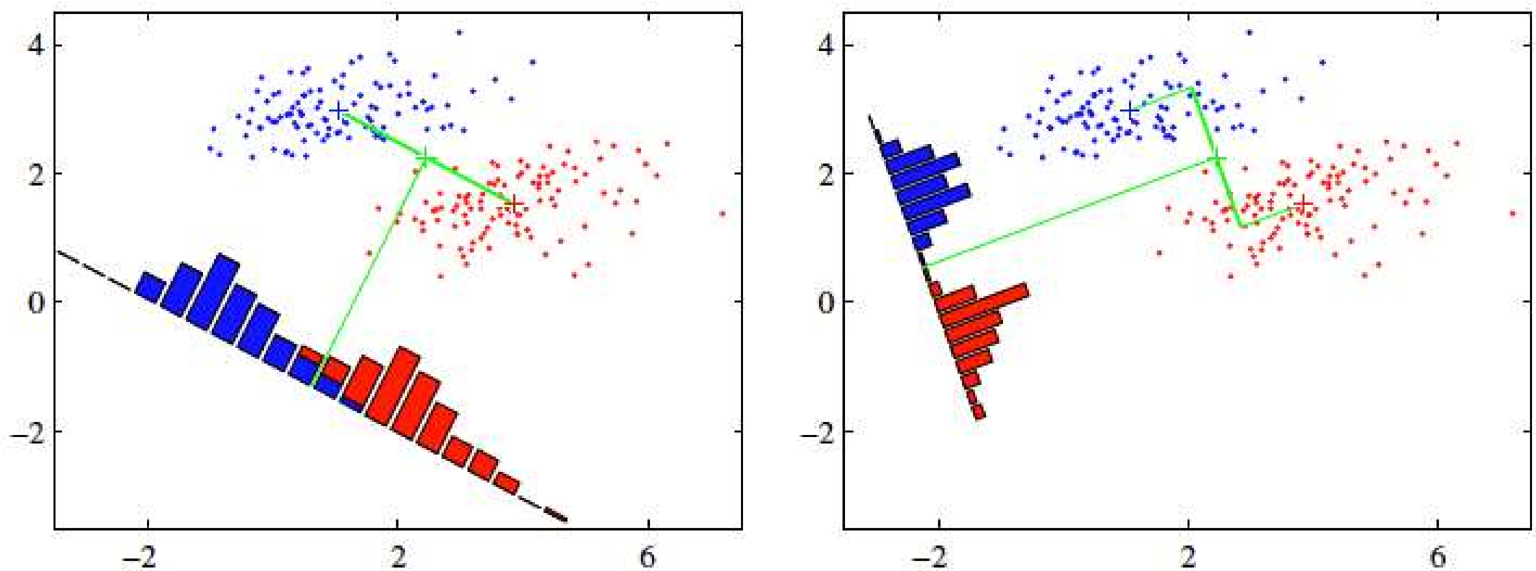
Training models for *a speaker*



$$\begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_k \end{bmatrix}$$

$$\mathbf{x} = \mathbf{V}\mathbf{w} + \mathbf{e} \qquad \mathbf{w} \sim N(0, I) \quad \mathbf{e} \sim N(0, E)$$

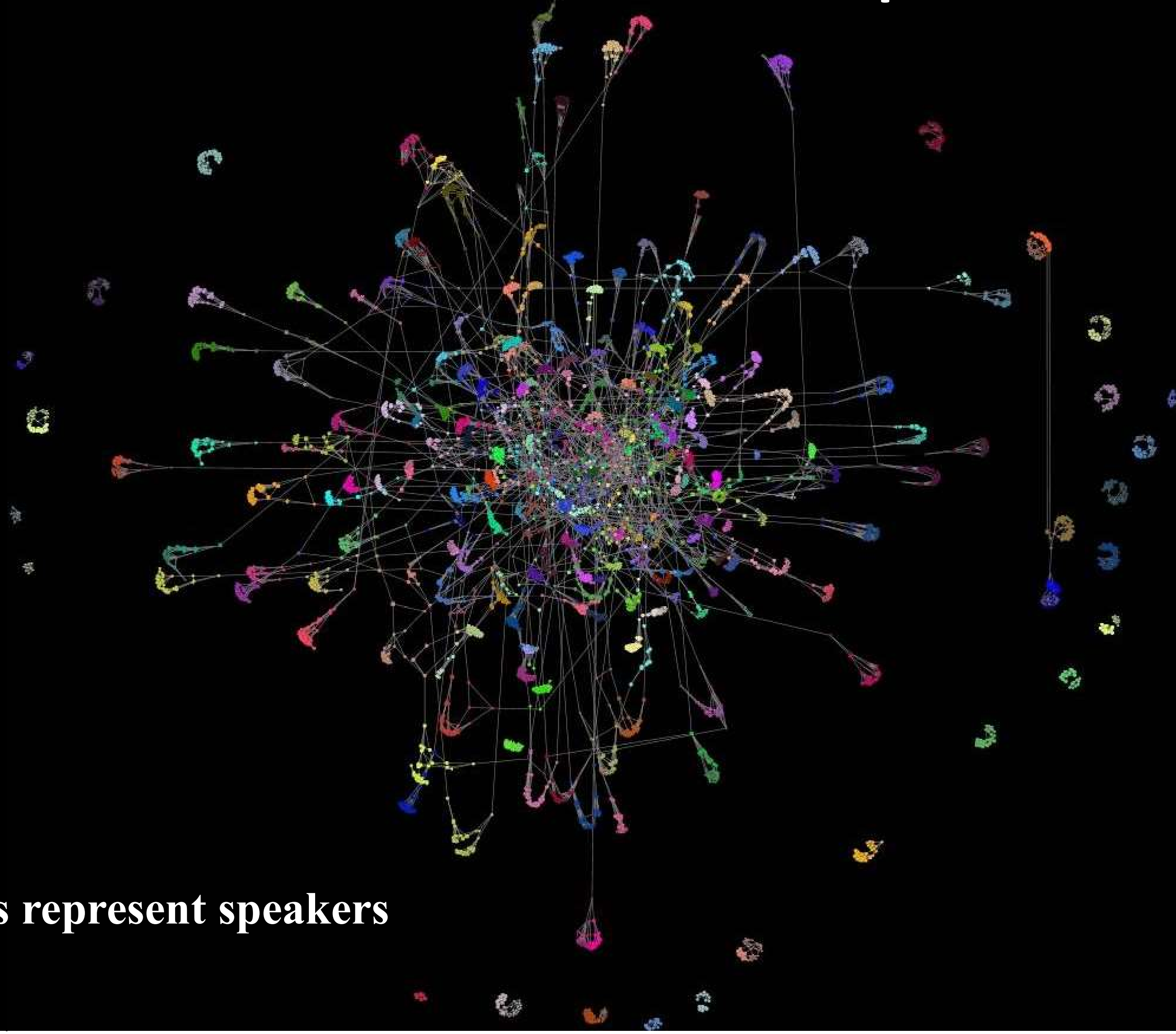
- Use Linear Discriminant Analysis to maximize the discrimination between the speakers



Data Visualization based on Graph

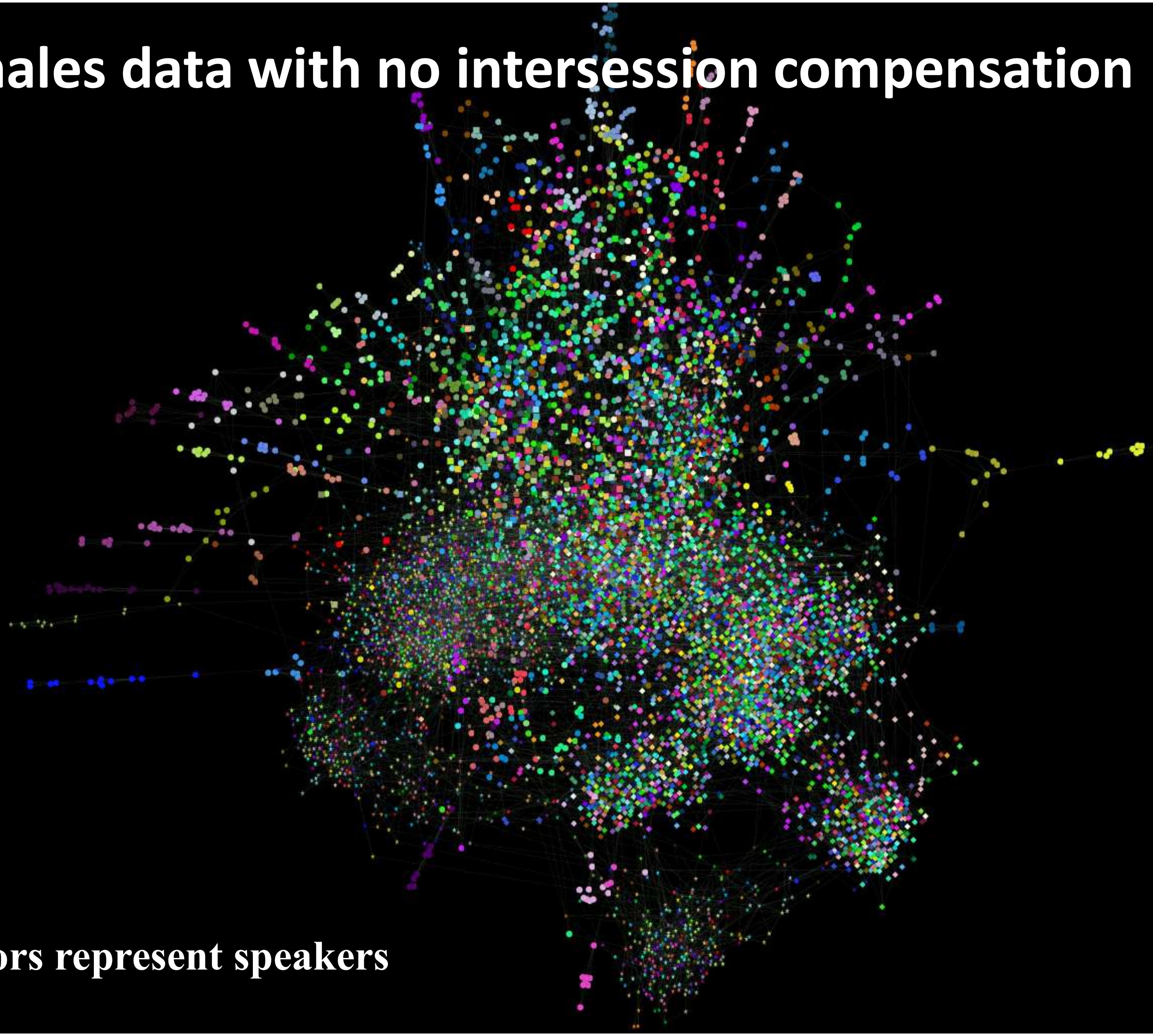
- Nice performance of the cosine similarity for speaker recognition
- **Data visualization using the Graph Exploration System (GUESS)**
- Represent segment as a node with connections (edges) to nearest neighbors (3 NN used)
  - NN computed using blind TV system (with and without channel normalization)
- Applied to 5438 utterances from the NIST SRE10 core
  - Multiple telephone and microphone channels
- Absolute locations of nodes not important
- Relative locations of nodes to one another is important:
  - The visualization clusters nodes that are highly connected together
- Meta data (speaker ID, channel info) not used in layout
- Colors and shapes of nodes used to highlight interesting phenomena

Females data with intersession compensation



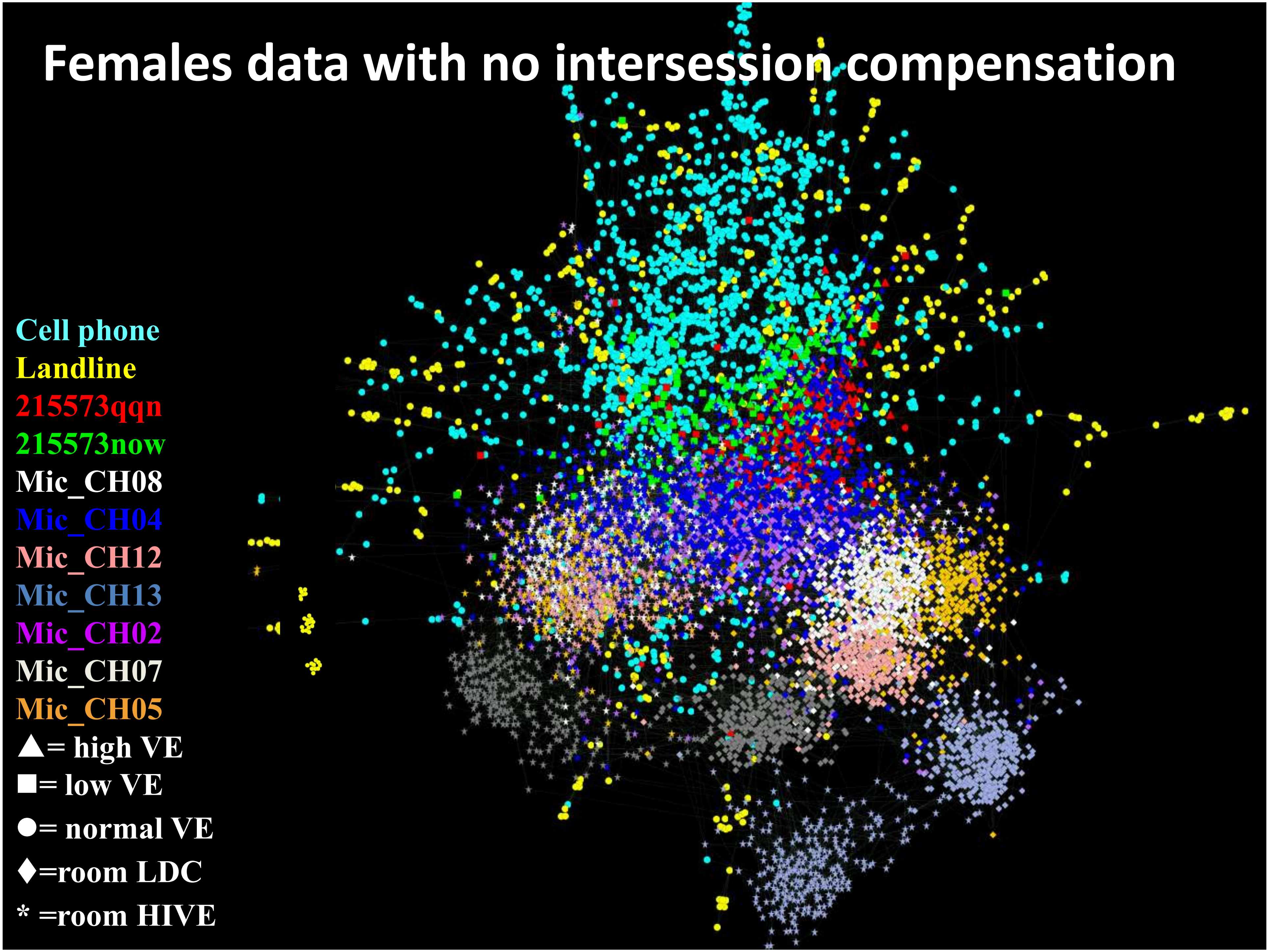
Colors represent speakers

Females data with no intersession compensation



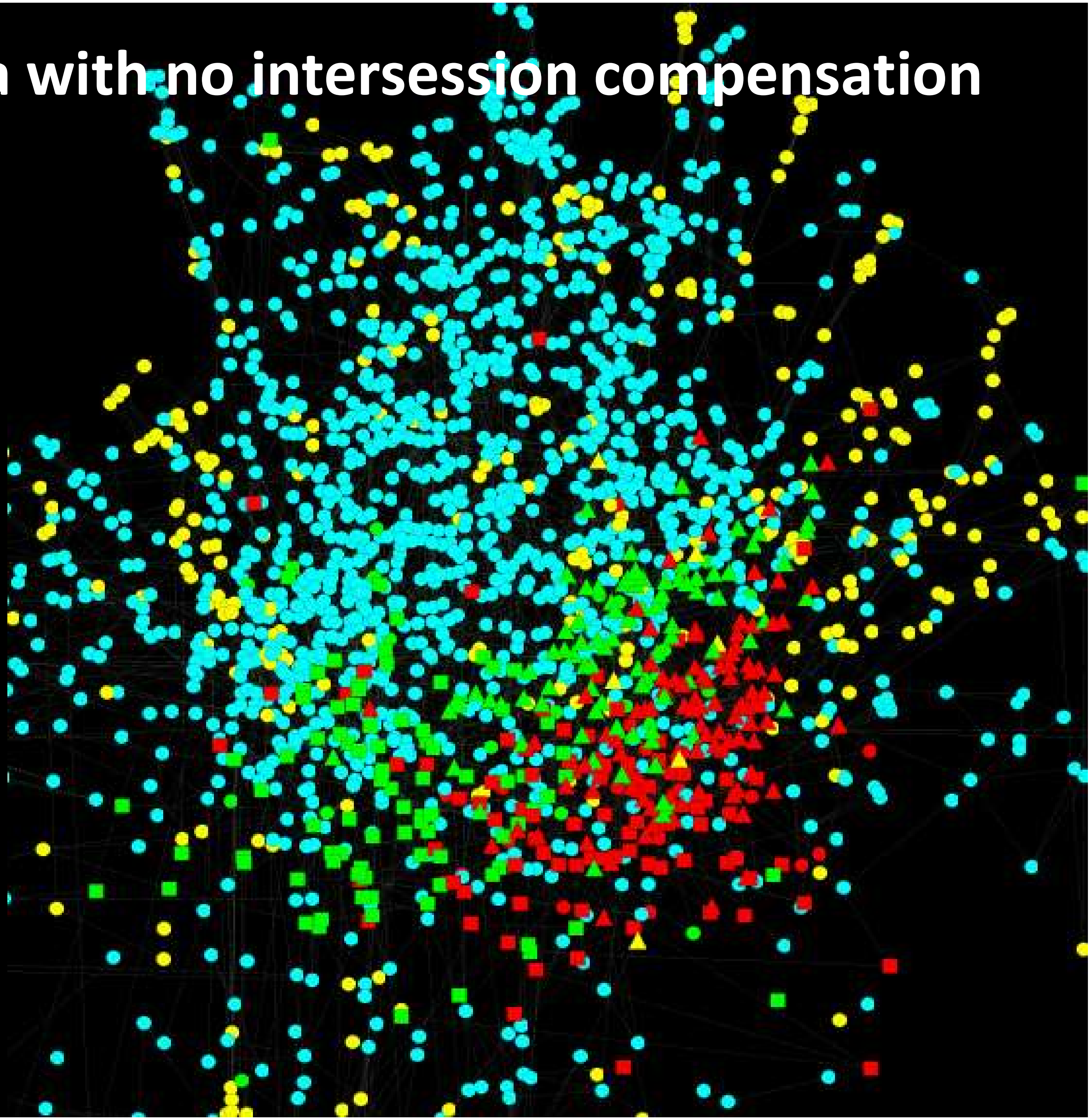
Colors represent speakers

Females data with no intersession compensation

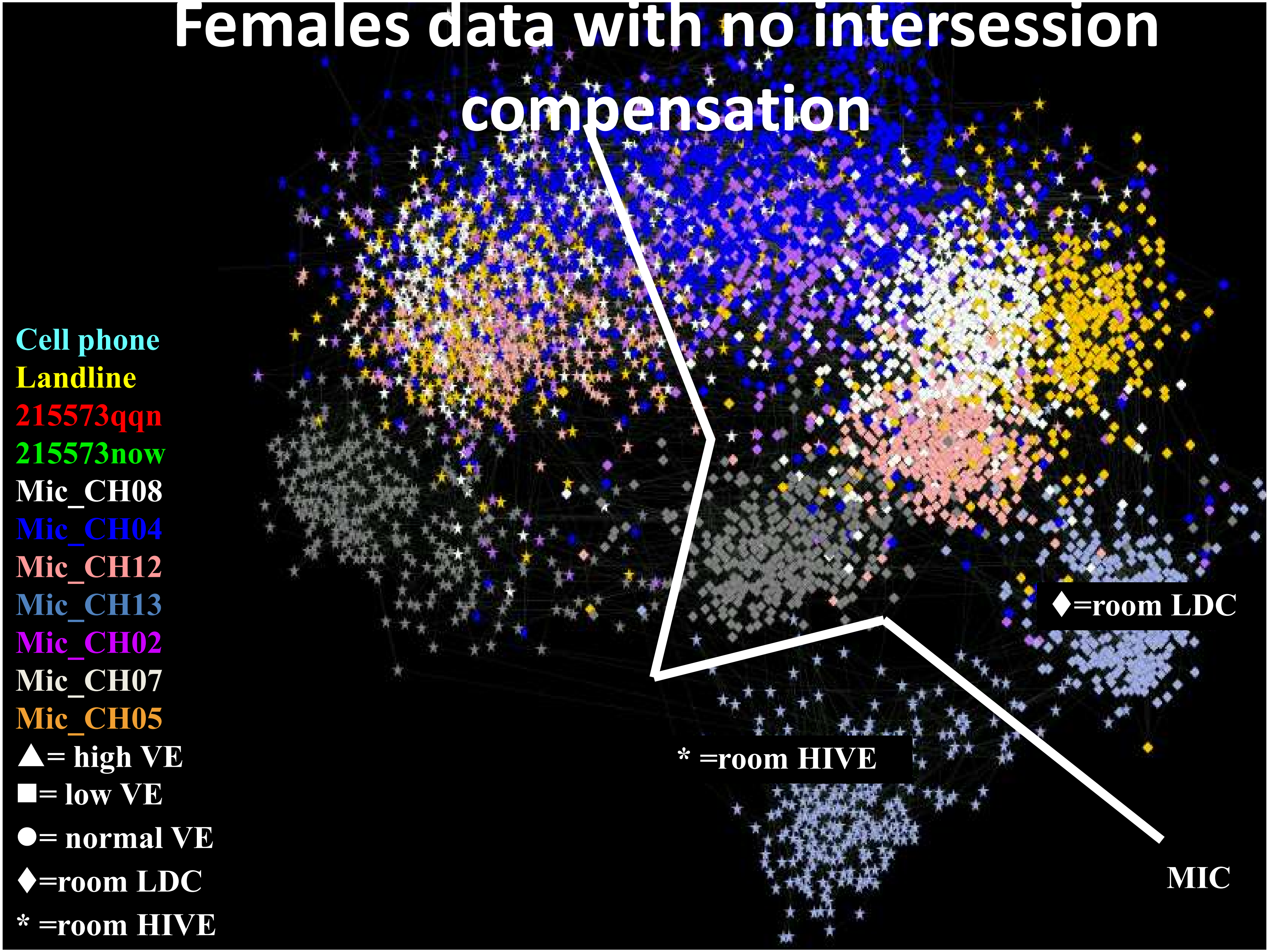


Females data with no intersession compensation

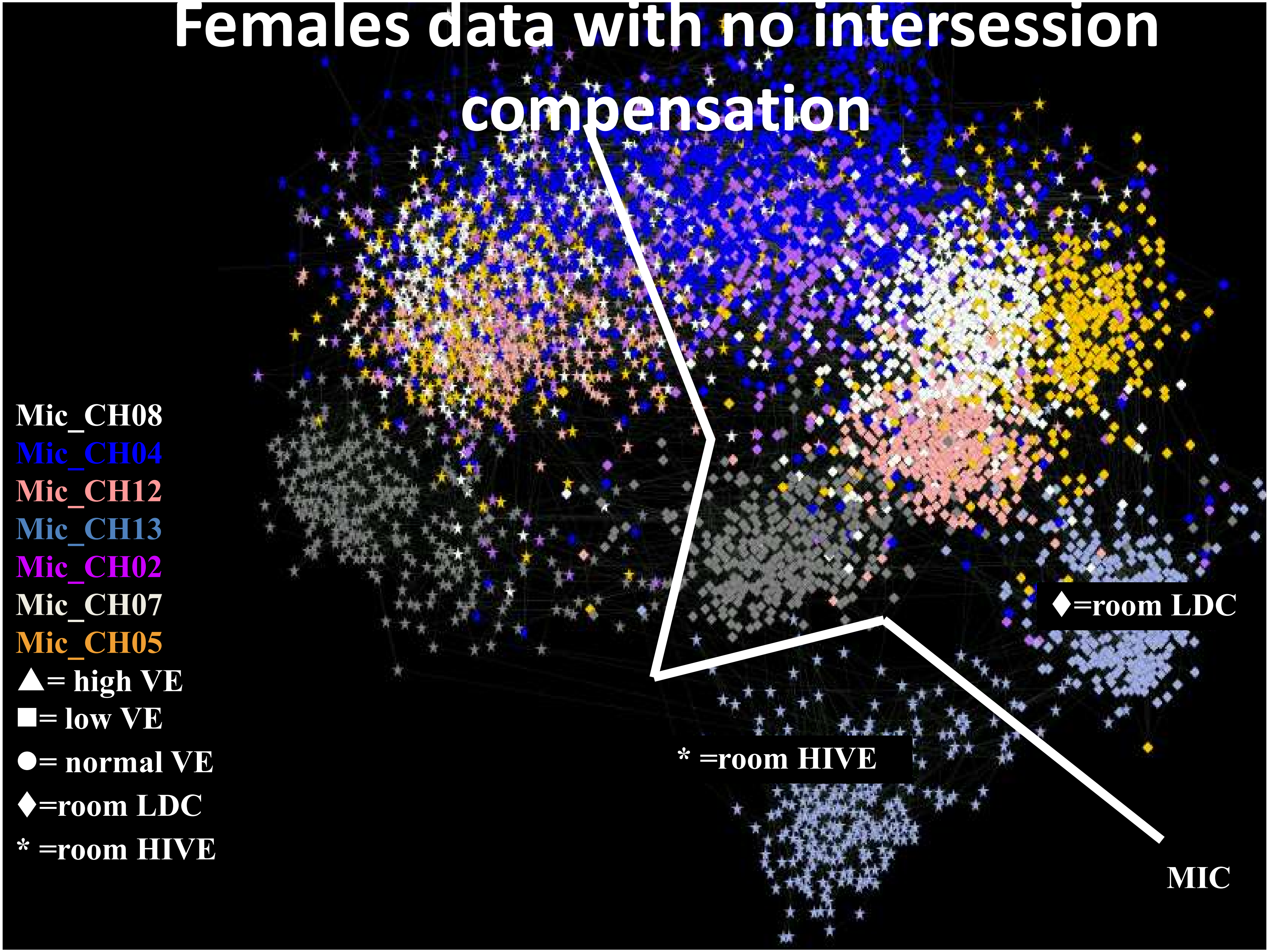
Cell phone
Landline
215573qqn
215573now
Mic_CH08
Mic_CH04
Mic_CH12
Mic_CH13
Mic_CH02
Mic_CH07
Mic_CH05
▲= high VE
■= low VE
●= normal VE
♦=room LDC
* =room HIVE



Females data with no intersession compensation

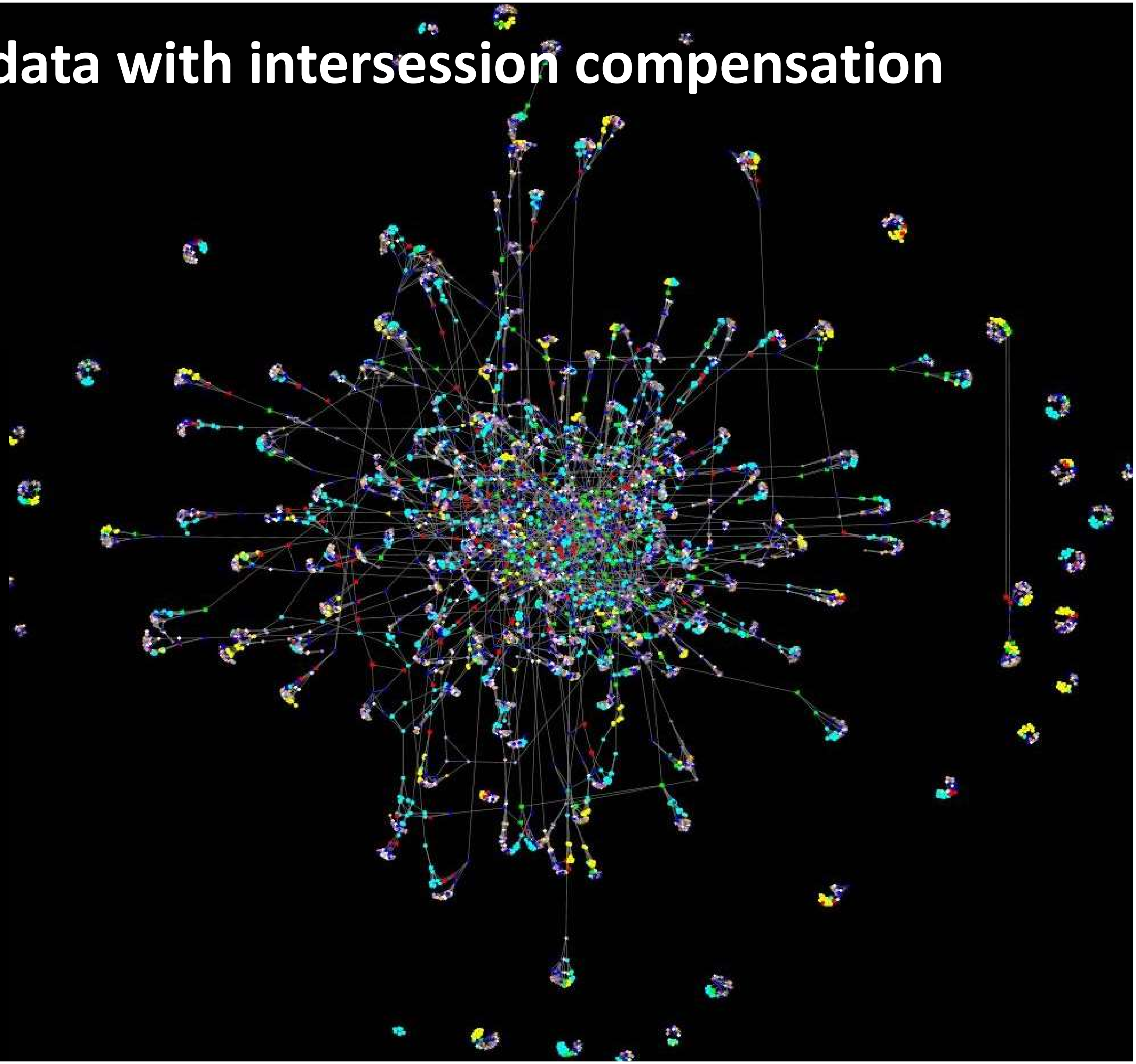


Females data with no intersession compensation

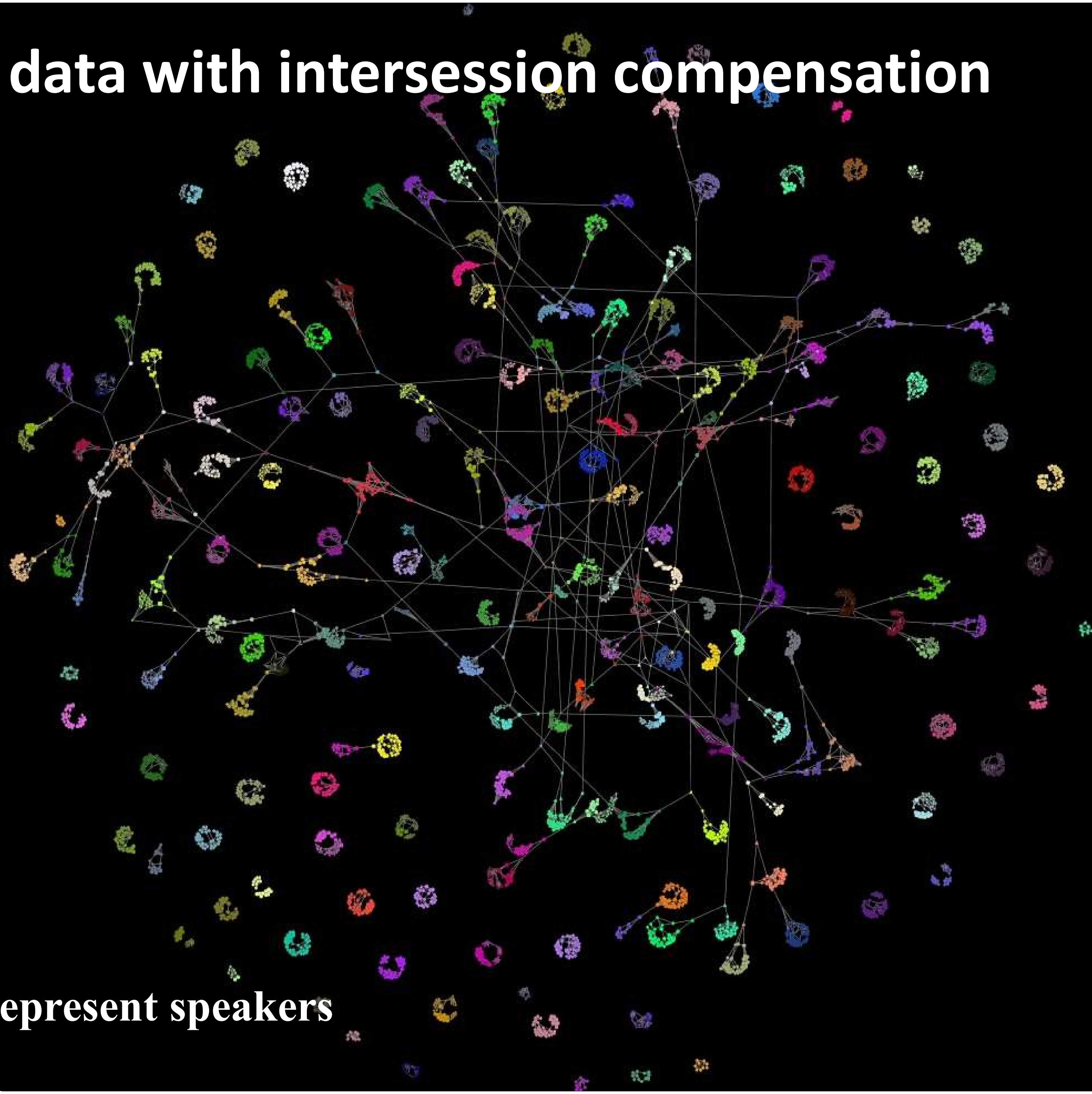


Females data with intersession compensation

Cell phone
Landline
215573qqn
215573now
Mic_CH08
Mic_CH04
Mic_CH12
Mic_CH13
Mic_CH02
Mic_CH07
Mic_CH05
▲= high VE
■= low VE
●= normal VE
♦=room LDC
* =room HIVE

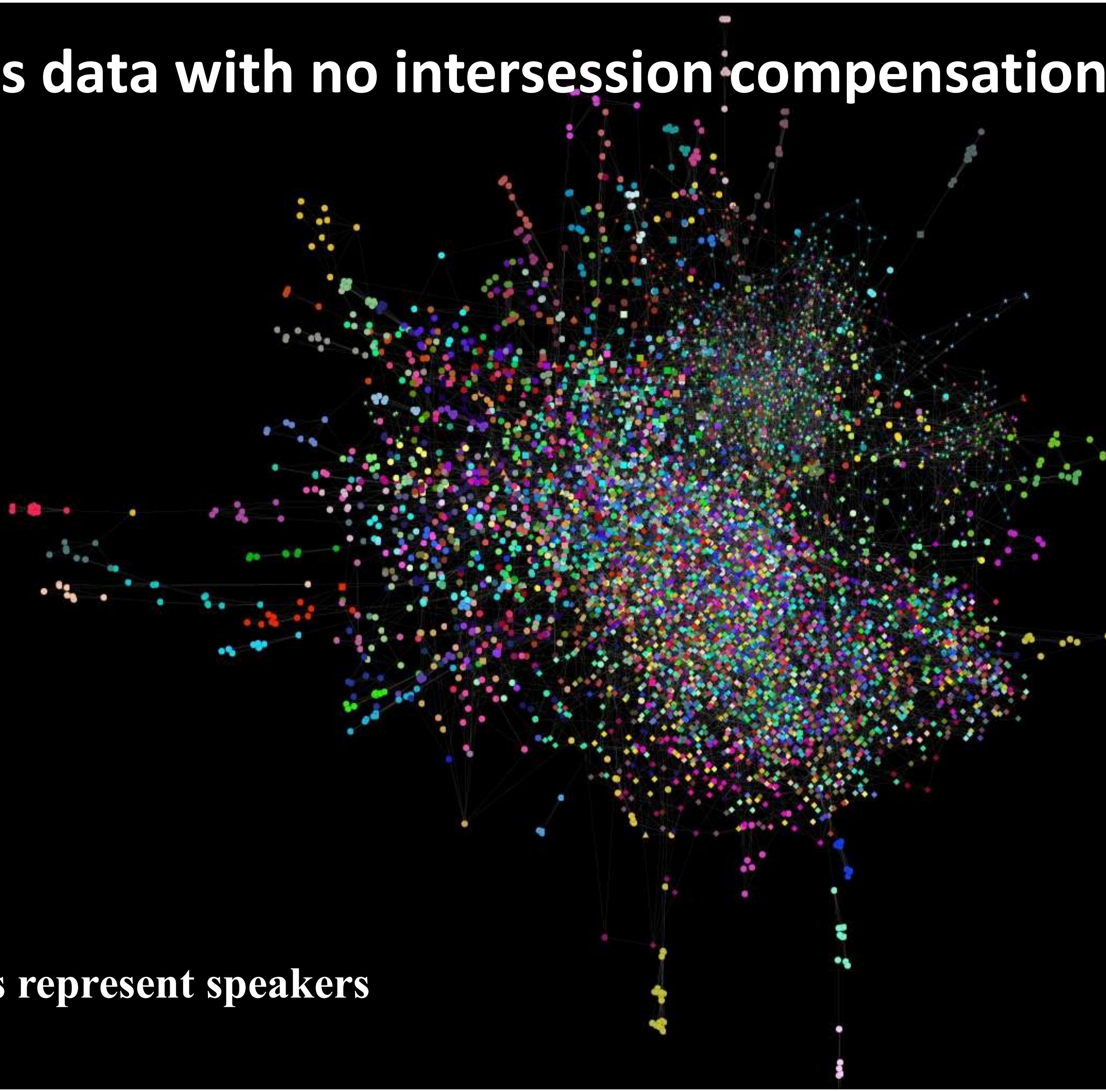


Males data with intersession compensation



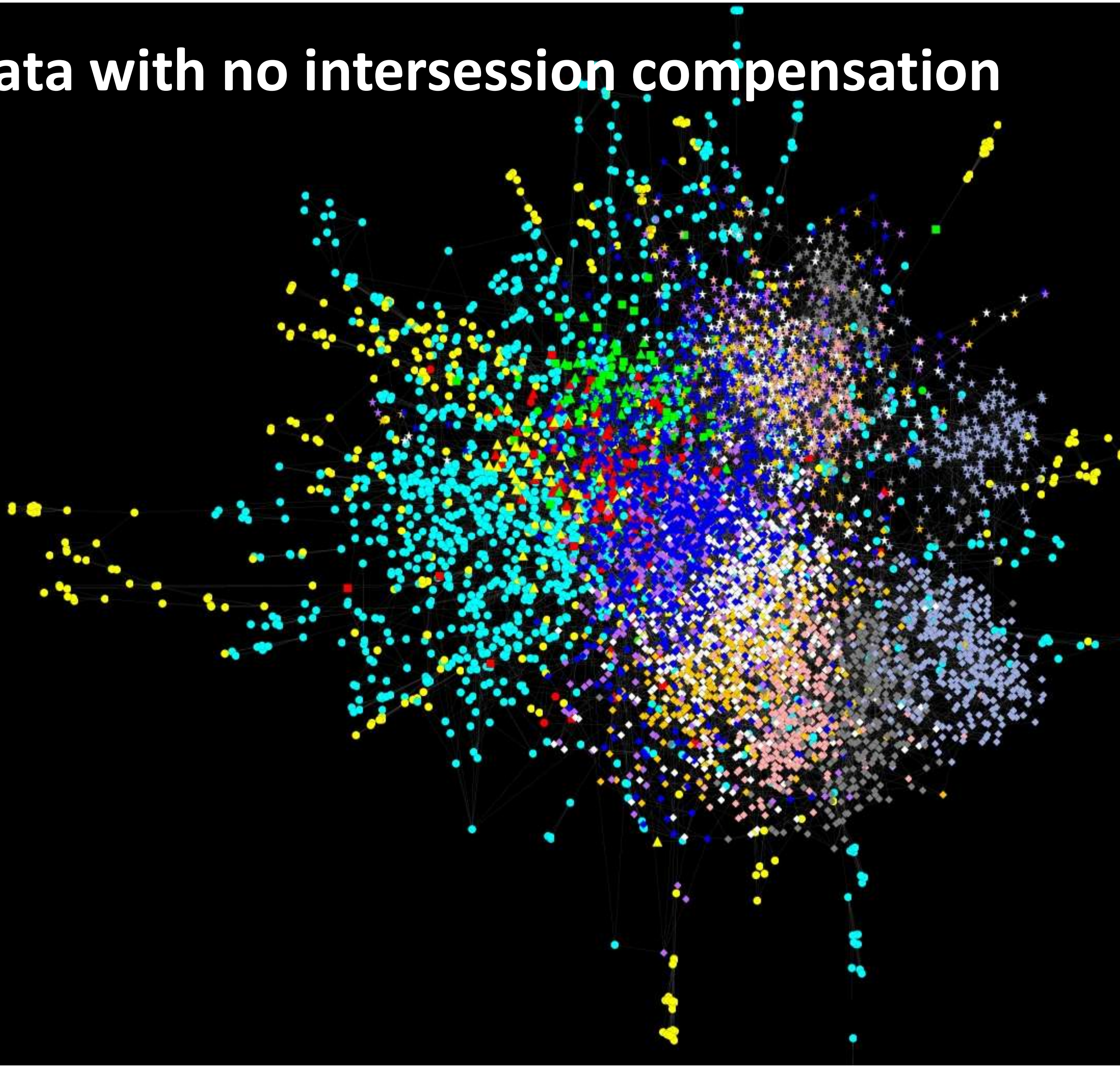
Colors represent speakers

Males data with no intersession compensation



Colors represent speakers

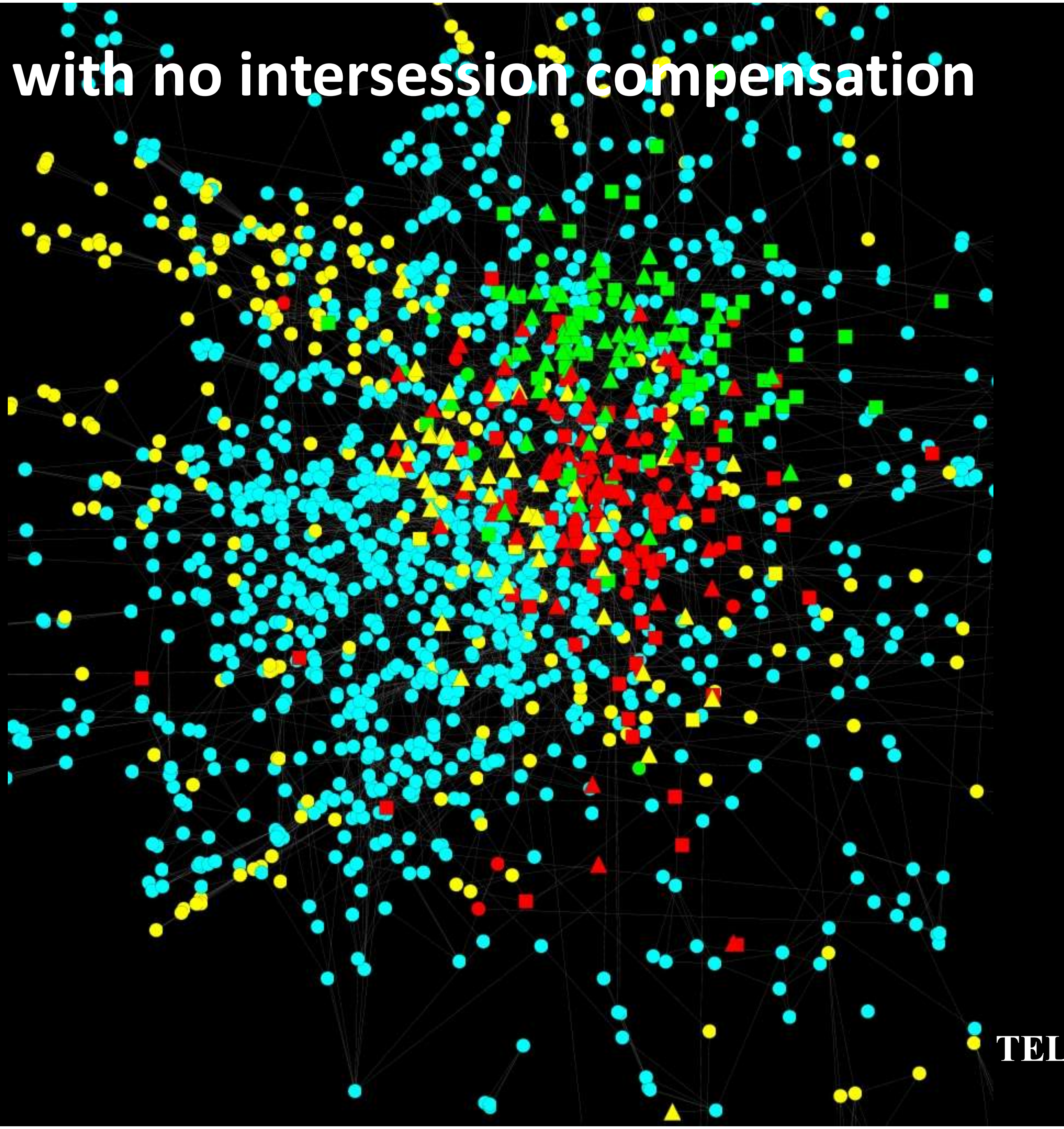
Males data with no intersession compensation

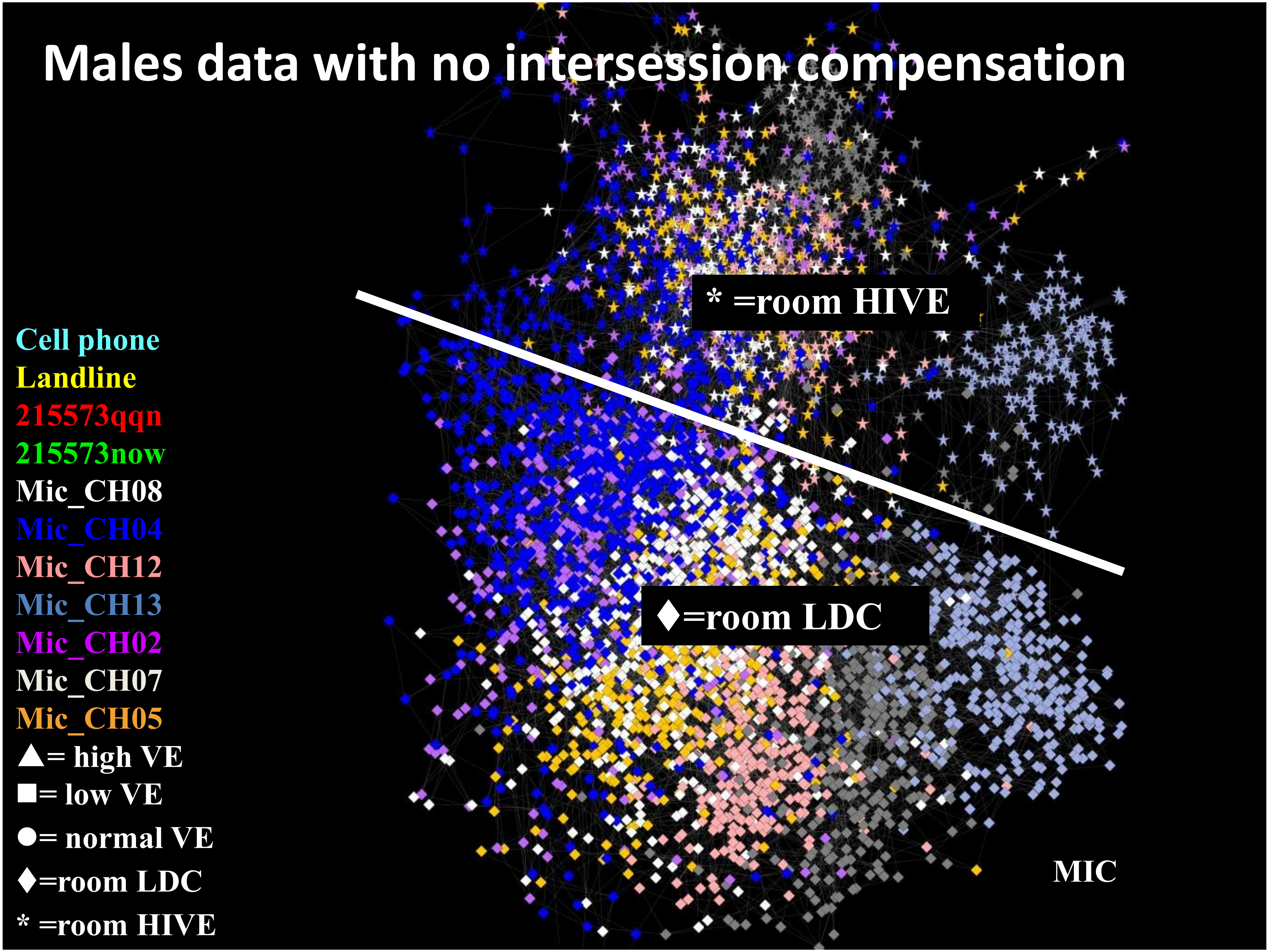


Cell phone
Landline
215573qqn
215573now
Mic_CH08
Mic_CH04
Mic_CH12
Mic_CH13
Mic_CH02
Mic_CH07
Mic_CH05
▲= high VE
■= low VE
●= normal VE
♦=room LDC
* =room HIVE

Males data with no intersession compensation

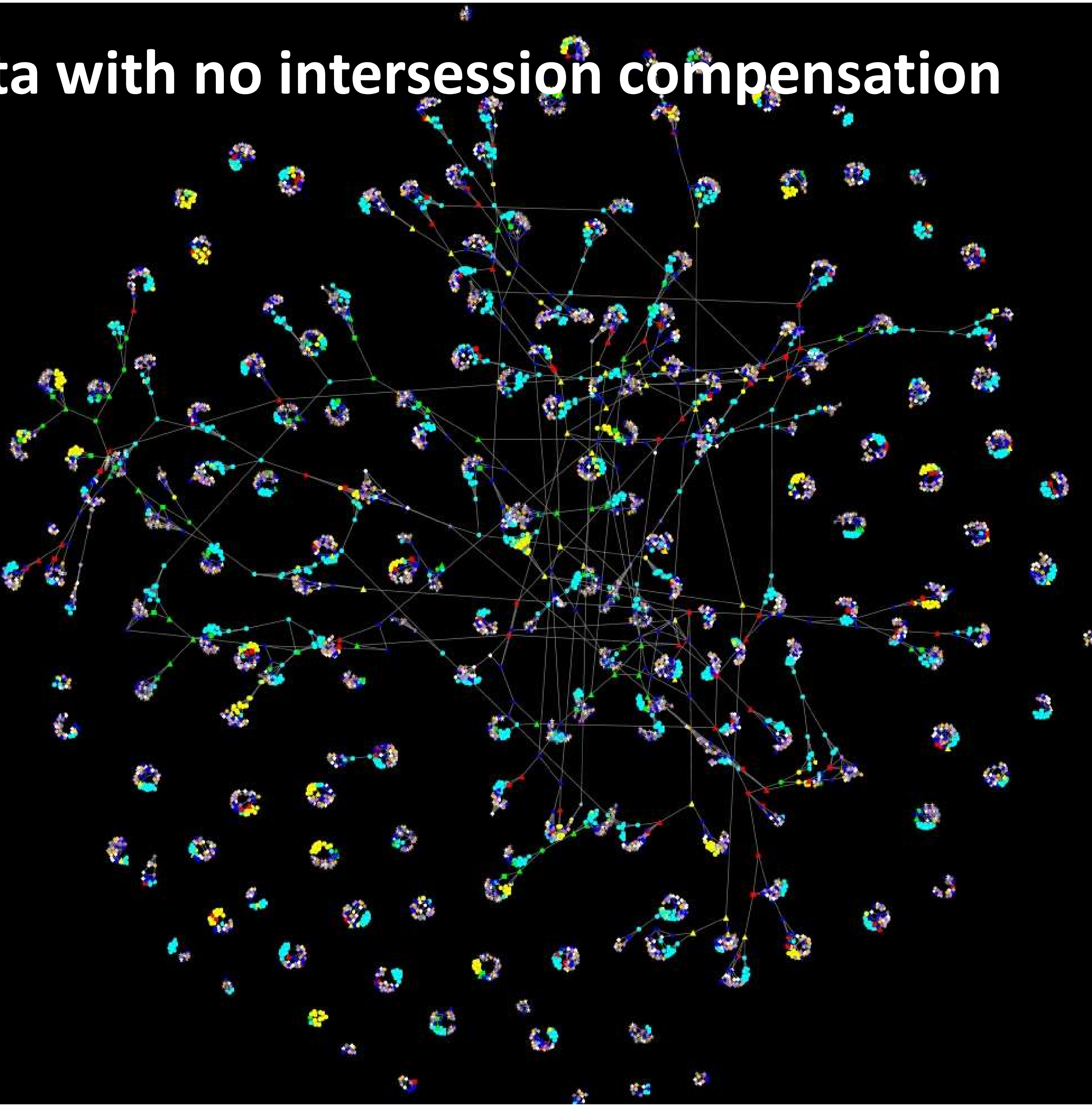
Cell phone
Landline
215573qqn
215573now
Mic_CH08
Mic_CH04
Mic_CH12
Mic_CH13
Mic_CH02
Mic_CH07
Mic_CH05
▲= high VE
■= low VE
●= normal VE
♦=room LDC
* =room HIVE



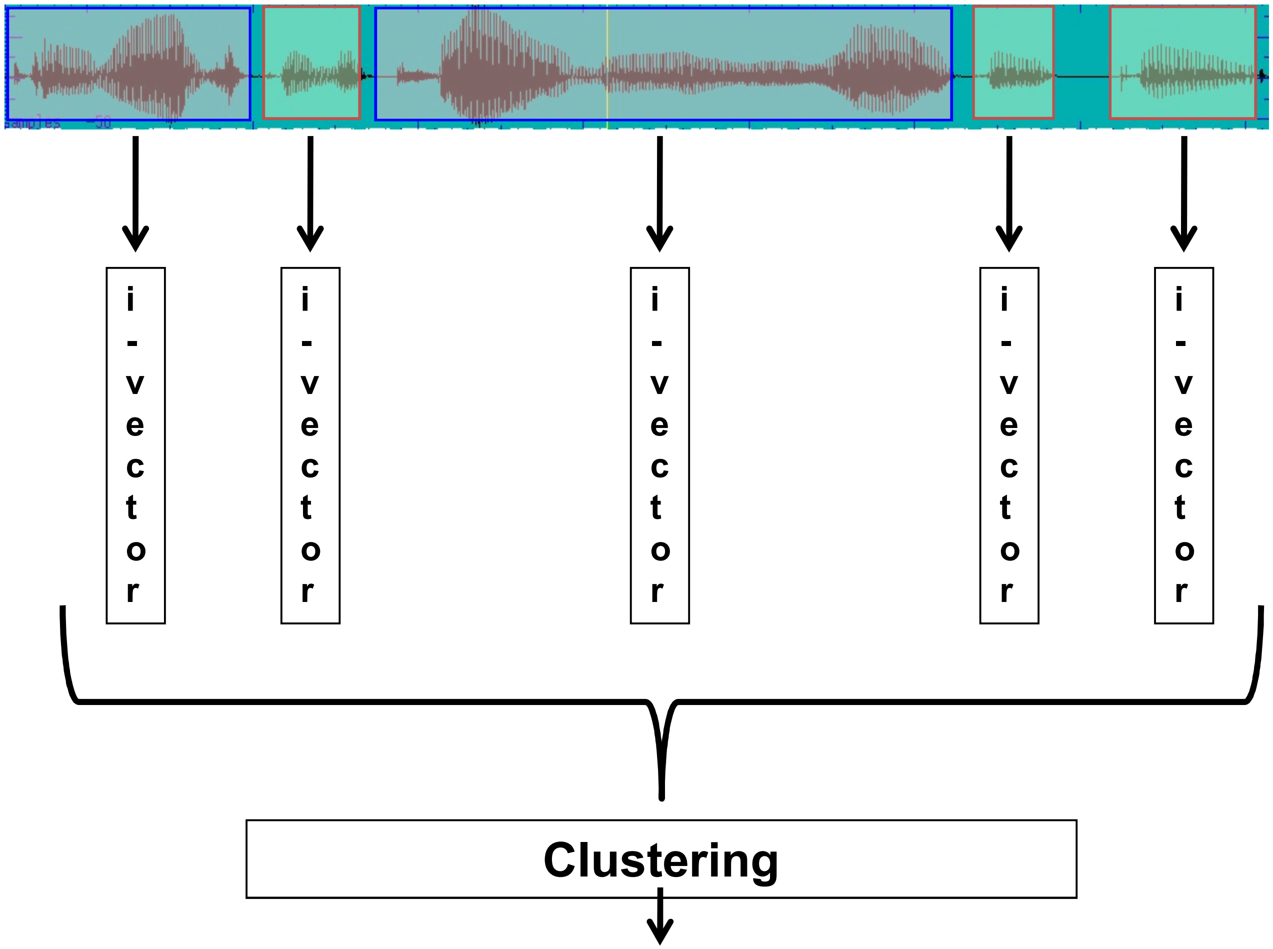

Males data with no intersession compensation
Cell phone
Landline
215573qqn
215573now
Mic_CH08
Mic_CH04
Mic_CH12
Mic_CH13
Mic_CH02
Mic_CH07
Mic_CH05
▲= high VE
■= low VE
●= normal VE
♦=room LDC
* =room HIVE
* =room HIVE
♦=room LDC
MIC

Males data with no intersession compensation

Cell phone
Landline
215573qqn
215573now
Mic_CH08
Mic_CH04
Mic_CH12
Mic_CH13
Mic_CH02
Mic_CH07
Mic_CH05
▲= high VE
■= low VE
●= normal VE
♦=room LDC
* =room HIVE

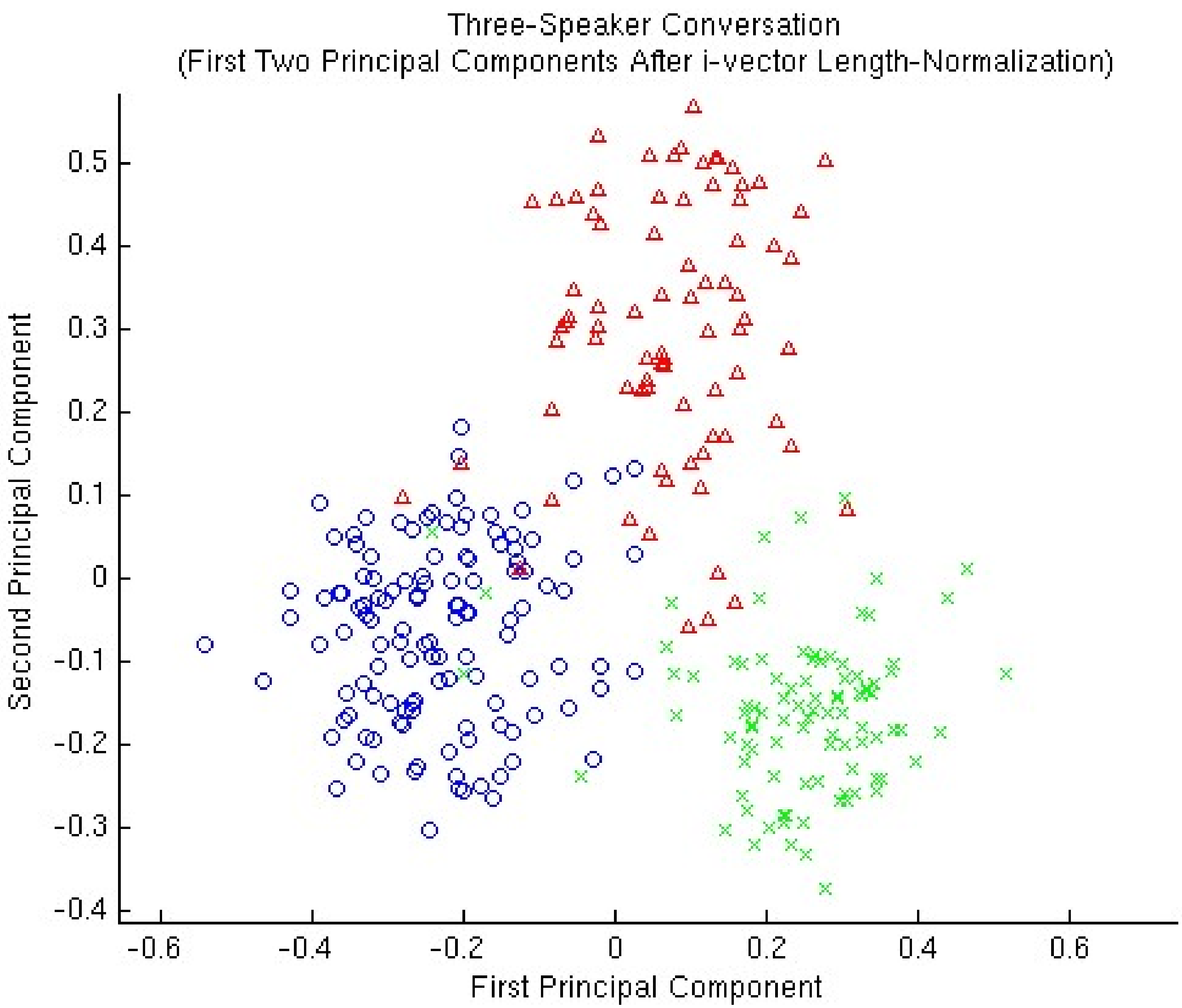


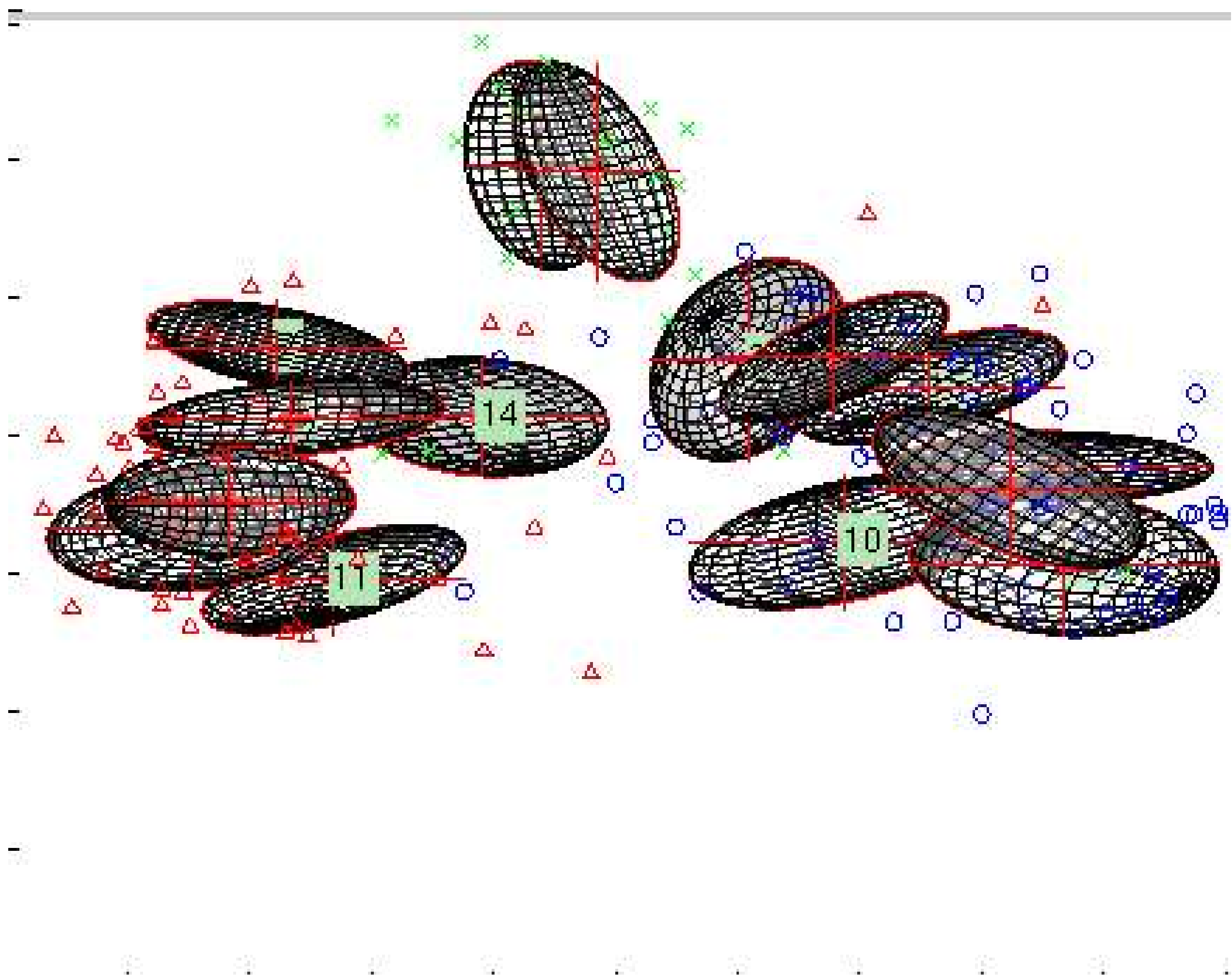
Speaker representation



Speaker clustering

PCA Visualization




14
11
10