Introduction to Machine Learning

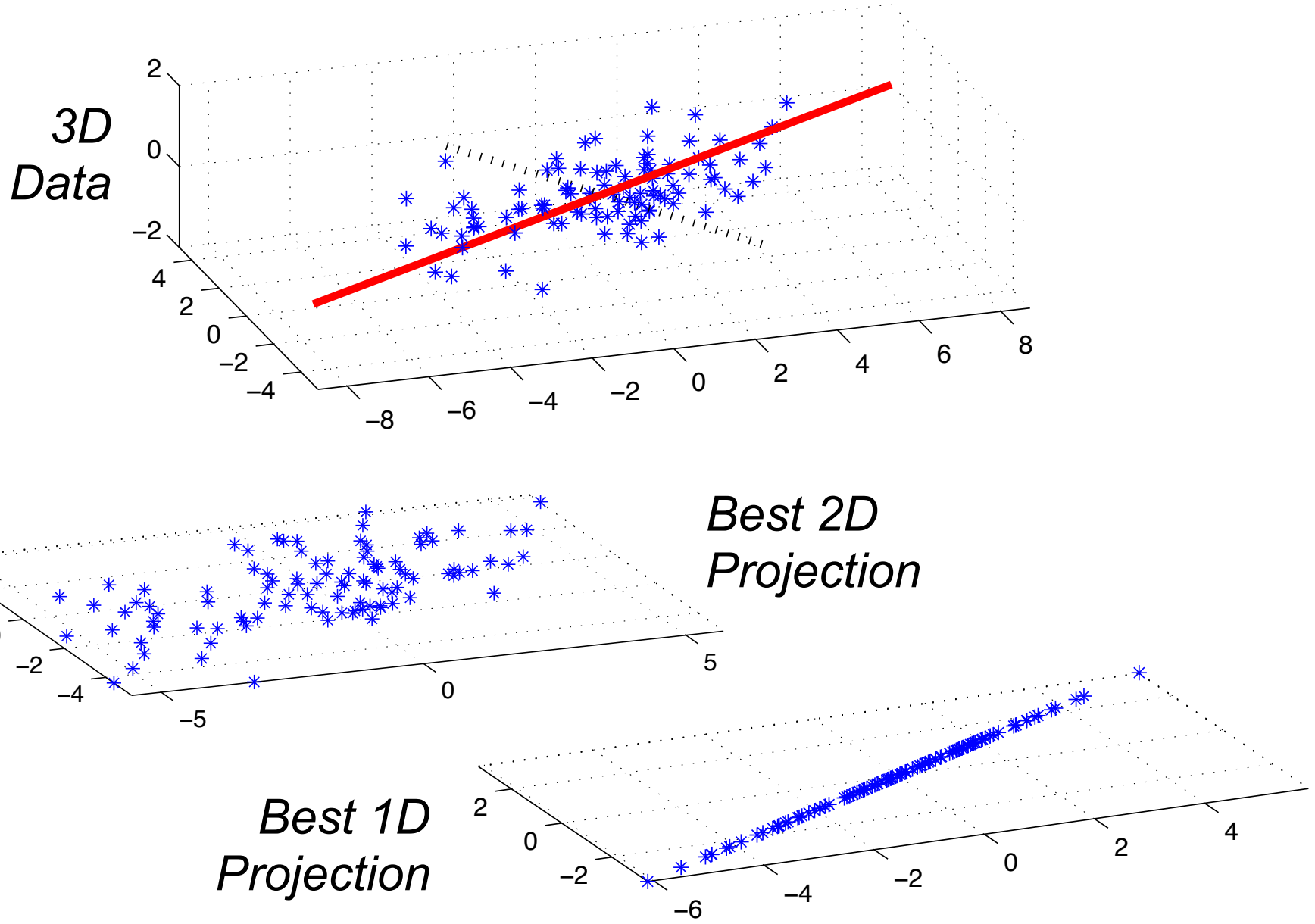
Brown University CSCI 1950-F, Spring 2012

Prof. Erik Sudderth

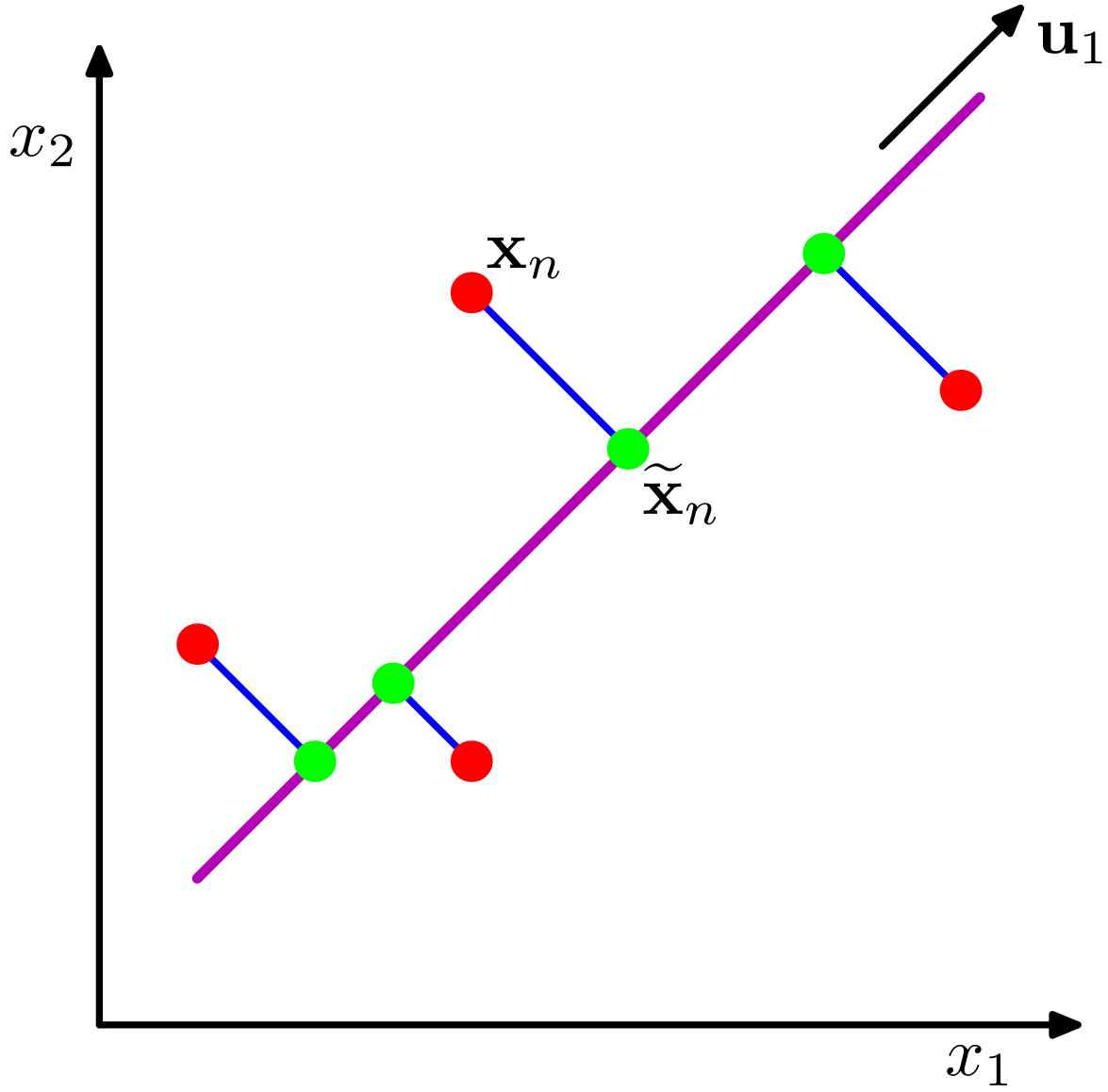
Lecture 22:
EM for Factor Analysis & PCA

Many figures courtesy Kevin Murphy's textbook,
Machine Learning: A Probabilistic Perspective

Principal Components Analysis (PCA)



Maximizes Variance & Minimizes Error



C. Bishop, Pattern Recognition & Machine Learning

PCA Optimal Solution

$$J(z, W, b \mid x, M) = \sum_{n=1}^{N} ||x_n - \tilde{x}_n||^2 = \sum_{n=1}^{N} ||x_n - W z_n - b||^2$$

$$b = \bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_n \qquad X = [x_1 - \bar{x}, x_2 - \bar{x}, \ldots, x_N - \bar{x}]$$

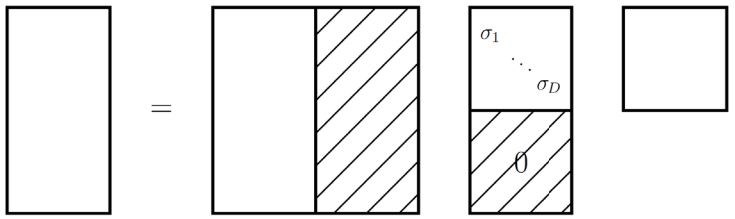
- Option A: Eigendecomposition of sample covariance matrix

$$\Sigma = \frac{1}{N} \sum_{n=1}^{N} (x_n - \bar{x})(x_n - \bar{x})^T = \frac{1}{N} X X^T = U \Lambda U^T$$

Construct W from eigenvectors with M largest eigenvalues

- Option B: Singular value decomposition (SVD) of centered data

$$X = U S V^T$$



Construct W from singular vectors with M largest singular values

Probabilistic PCA & Factor Analysis

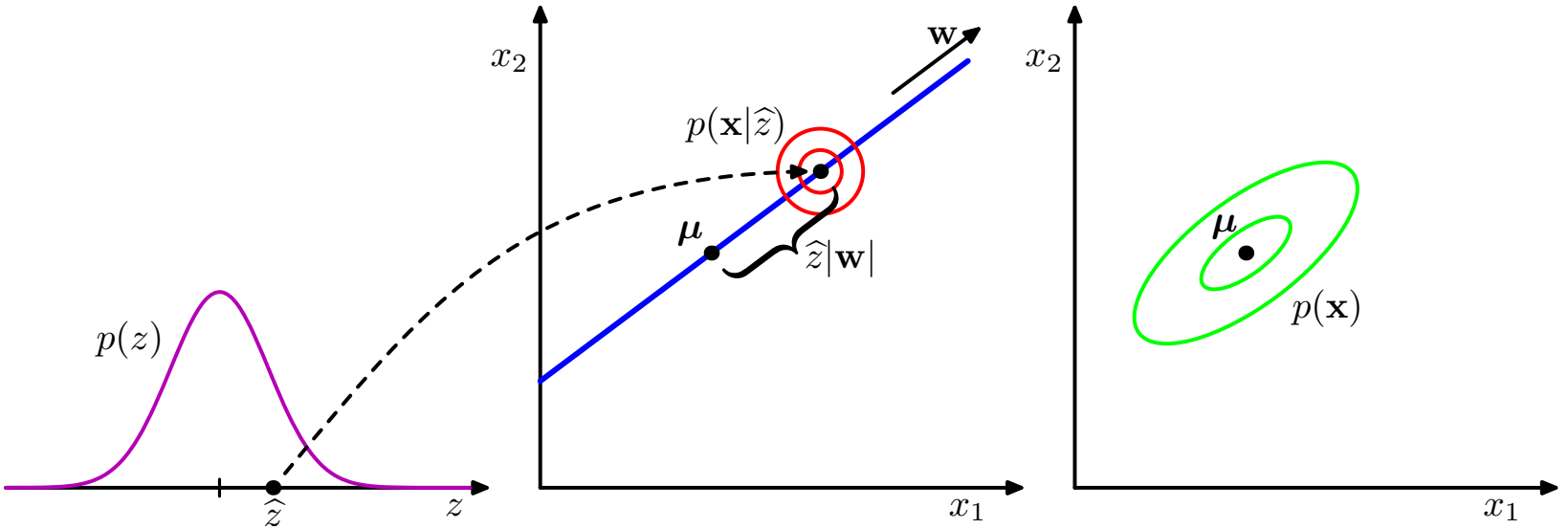
- **Both Models:** Data is a linear function of low-dimensional latent coordinates, plus Gaussian noise

$$p(x_i \mid z_i, \theta) = \mathcal{N}(x_i \mid W z_i + \mu, \Psi) \qquad p(z_i \mid \theta) = \mathcal{N}(z_i \mid 0, I)$$

$$p(x_i \mid \theta) = \mathcal{N}(x_i \mid \mu, W W^T + \Psi)$$

low rank covariance parameterization

- **Factor analysis:** Ψ is a general diagonal matrix
- **Probabilistic PCA:** $\Psi = \sigma^2 I$ is a multiple of identity matrix



C. Bishop, Pattern Recognition & Machine Learning

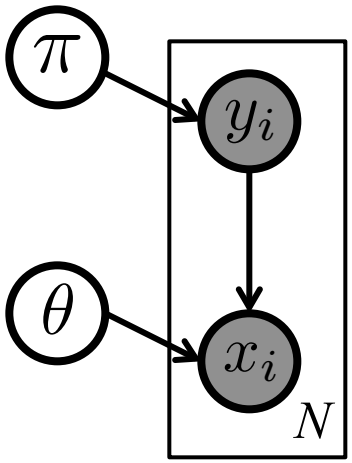
Dimensionality Reduction & Invariance

$$p(x_i \mid z_i, \theta) = \mathcal{N}(x_i \mid Wz_i + \mu, \Psi) \qquad p(z_i \mid \theta) = \mathcal{N}(z_i \mid 0, I)$$

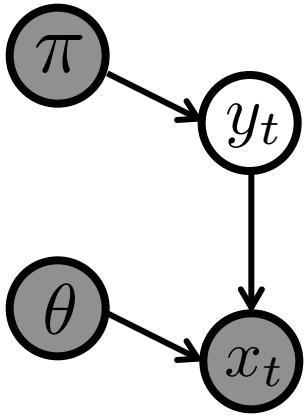
$$p(x_i \mid \theta) = \mathcal{N}(x_i \mid \mu, WW^T + \Psi)$$

- Global translation of high-dimensional data: $x_i \leftarrow x_i + t$
  Modeled by shift of model mean parameter: $\mu \leftarrow \mu + t$
- Rescaling of embedding: $z_i \leftarrow \alpha z_i, W \leftarrow \alpha^{-1} W$
  Gaussian prior chooses canonical latent scale: $\mathrm{Cov}[Z_i] \approx I$
- Rotation or reflection of embedding: $W \leftarrow WR, z_i \leftarrow R^T z_i$
  Determined arbitrarily by learning initialization $RR^T = I$
- Rescaling of high-dimensional data: $x_{ij} \leftarrow \beta_j x_{ij}$
  Affects PPCA, but Factor Analysis is invariant: $\Psi_{jj} \leftarrow \beta_j^2 \Psi_{jj}$
- Rotation of high-dimensional data: $x_i \leftarrow Qx_i, QQ^T = I$
  Affects Factor Analysis, but PPCA is invariant: $\Psi = \sigma^2 I$

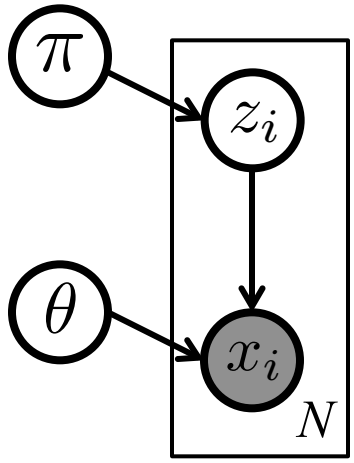
Expectation Maximization (EM)



Supervised Training



Supervised Testing



Unsupervised Learning

$\pi, \theta \longrightarrow$ *parameters (define low-dimensional manifold)*

$z_1, \ldots, z_N \longrightarrow$ *hidden data (locate observations on manifold)*

- **Initialization:** Randomly select starting parameters
- **E-Step:** Given parameters, find posterior of hidden data
  - Equivalent to test inference of full posterior distribution
- **M-Step:** Given posterior distributions, find likely parameters
  - Similar to supervised ML/MAP training
- **Iteration:** Alternate E-step & M-step until convergence

EM as Lower Bound Maximization

$$\ln p(x \mid \theta) = \ln \left(\int_z p(x, z \mid \theta)\, dz \right)$$

$$\ln p(x \mid \theta) \geq \int_z q(z) \ln p(x, z \mid \theta)\, dz - \int_z q(z) \ln q(z)\, dz \triangleq \mathcal{L}(q, \theta)$$

- **Initialization:** Randomly select starting parameters $\theta^{(0)}$
- **E-Step:** Given parameters, find posterior of hidden data

$$q^{(t)} = \arg\max_q \mathcal{L}(q, \theta^{(t-1)})$$

- **M-Step:** Given posterior distributions, find likely parameters

$$\theta^{(t)} = \arg\max_\theta \mathcal{L}(q^{(t)}, \theta)$$

- **Iteration:** Alternate E-step & M-step until convergence

EM: Expectation Step

$$\ln p(x \mid \theta) \geq \int_z q(z) \ln p(x, z \mid \theta)\, dz - \int_z q(z) \ln q(z)\, dz \triangleq \mathcal{L}(q, \theta)$$

$$q^{(t)} = \arg\max_q \mathcal{L}(q, \theta^{(t-1)})$$

- General solution, for any probabilistic model:

$$q^{(t)}(z) = p(z \mid x, \theta^{(t-1)})$$

posterior distribution given current parameters

- For factor analysis and probabilistic PCA these are Gaussian:

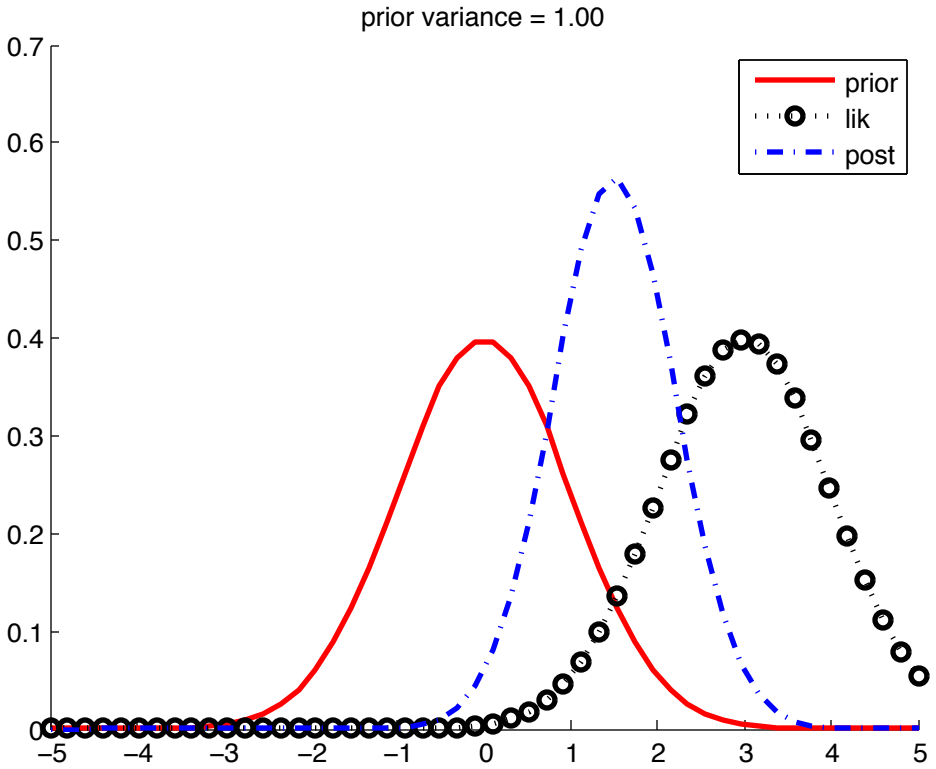
$$p(z \mid x, \theta) = \prod_{i=1}^{N} p(z_i \mid x_i, \theta) \qquad \theta = \{W, \mu, \Psi\}$$

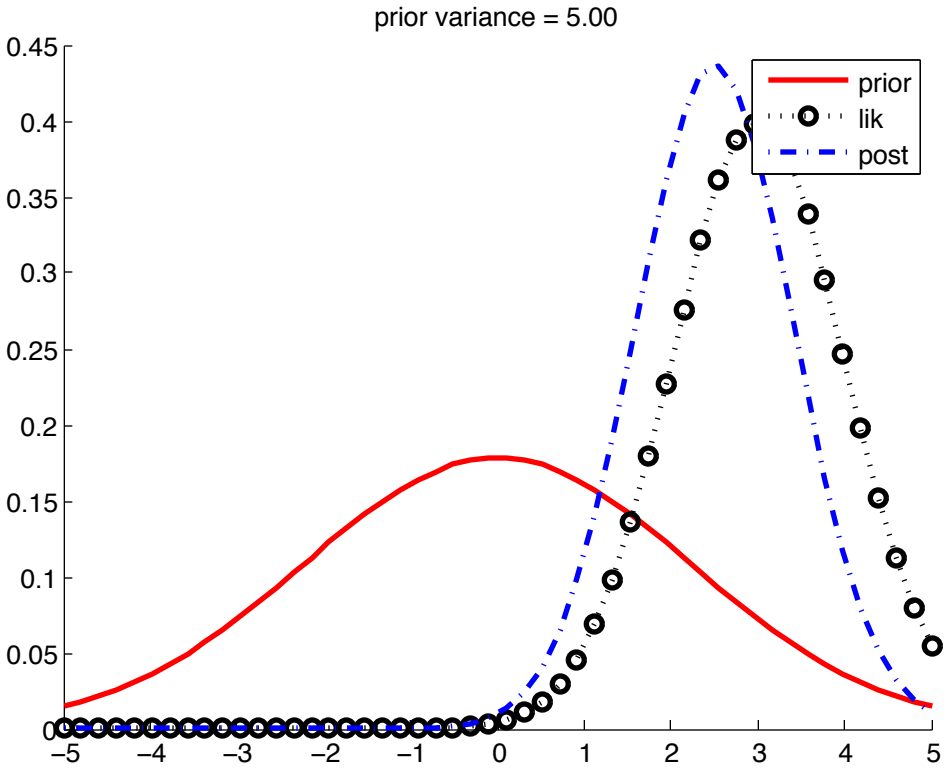
$$p(z_i \mid x_i, W, \mu, \Psi) = \mathcal{N}(z_i \mid \Sigma_i W^T \Psi^{-1}(x_i - \mu), \Sigma_i)$$

$$\Sigma_i^{-1} = I + W^T \Psi^{-1} W$$

Gaussians & Prior Covariance

$$p(x) = \mathcal{N}(x|\mu_0, \lambda_0^{-1}) \qquad p(y_i|x) = \mathcal{N}(y_i|x, \lambda_y^{-1})$$





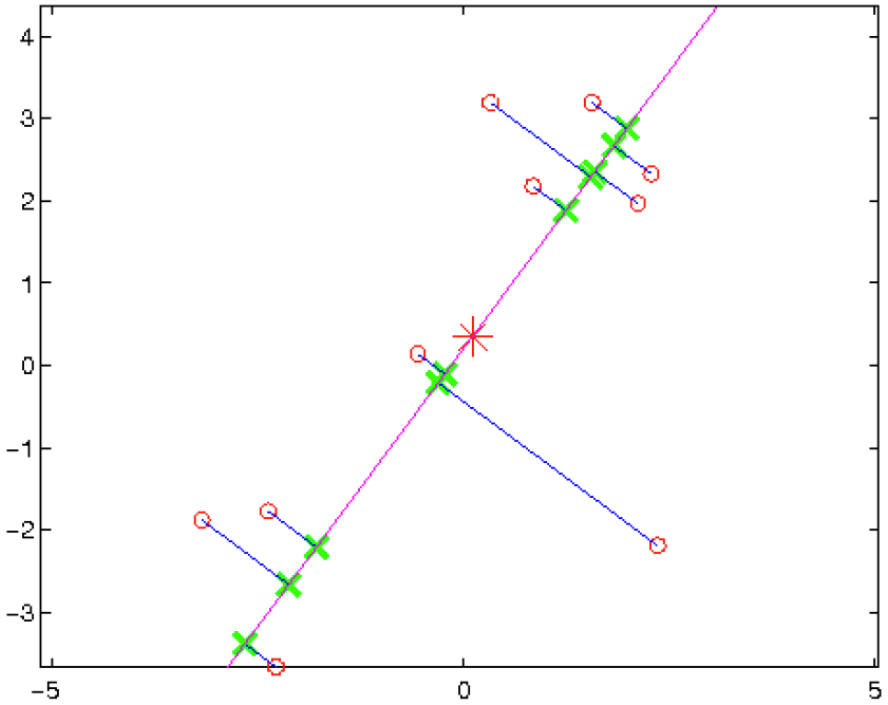
$$p(x|\mathbf{y}) = \mathcal{N}(x|\mu_N, \lambda_N^{-1})$$

$$\lambda_N = \lambda_0 + N\lambda_y$$

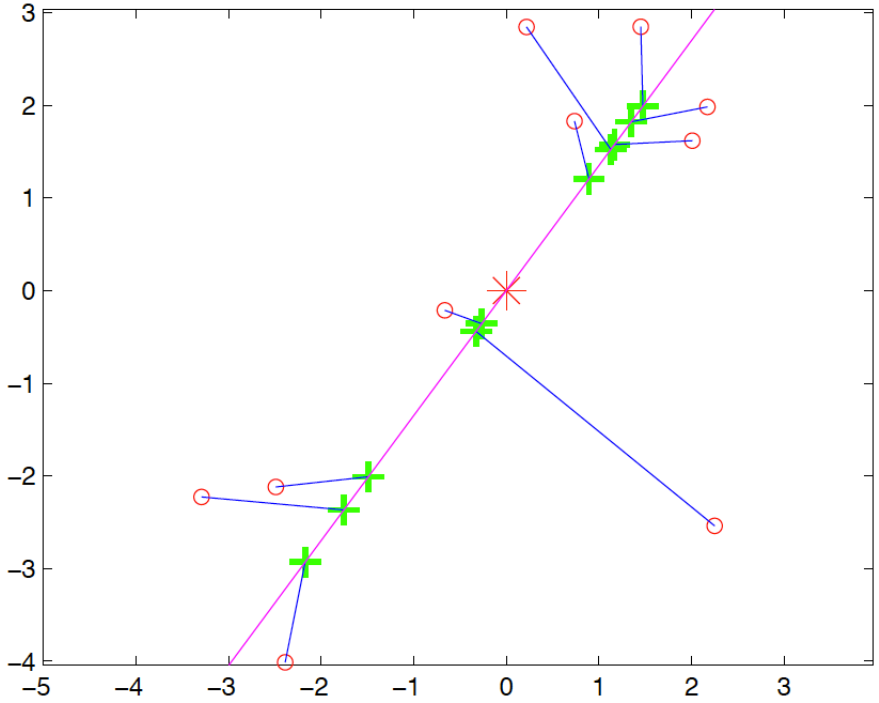
$$\mu_N = \frac{N\lambda_y \overline{y} + \lambda_0 \mu_0}{\lambda_N} = \frac{N\lambda_y}{N\lambda_y + \lambda_0}\overline{y} + \frac{\lambda_0}{N\lambda_y + \lambda_0}\mu_0$$

PCA versus Probabilistic PCA

$$p(z_i \mid x_i, W, \mu, \Psi) = \mathcal{N}(z_i \mid \Sigma_i W^T \Psi^{-1}(x_i - \mu), \Sigma_i) \qquad \Sigma_i^{-1} = I + W^T \Psi^{-1} W$$



Standard PCA
(orthogonal projection)



Probabilistic PCA
(shrunk towards mean)

- Maximum likelihood estimates of probabilistic PCA parameters are equal to the classic PCA eigenvector solution
- For classical PCA, optimal embedding is orthogonal projection
- For PPCA, latent coordinates are biased towards mean (zero)

EM: Maximization Step

$$\ln p(x \mid \theta) \geq \int_z q(z) \ln p(x, z \mid \theta)\, dz - \int_z q(z) \ln q(z)\, dz \triangleq \mathcal{L}(q, \theta)$$

$$\theta^{(t)} = \arg\max_\theta \mathcal{L}(q^{(t)}, \theta) = \arg\max_\theta \int_z q(z) \ln p(x, z \mid \theta)\, dz$$

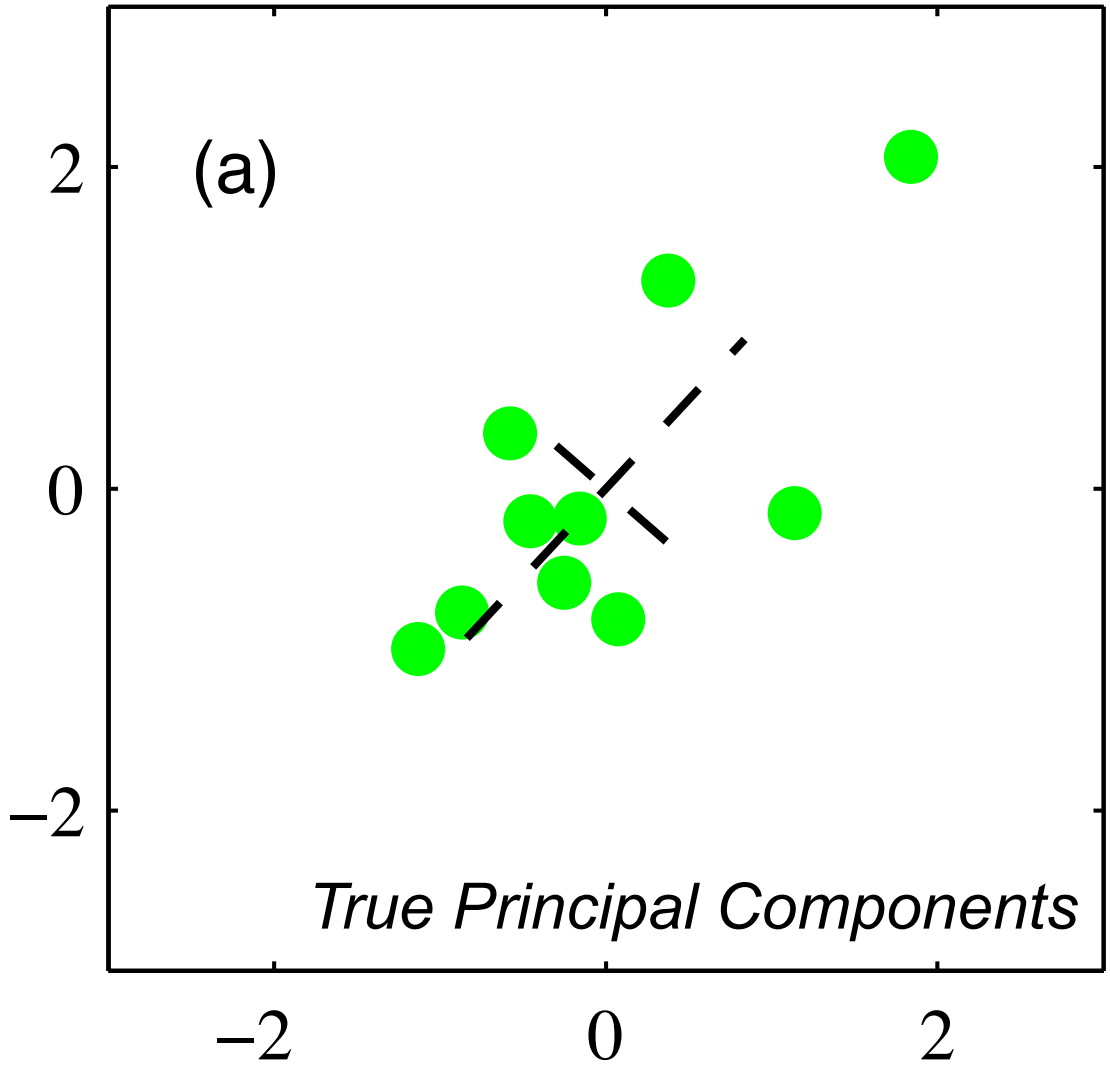
- Unlike E-step, no simplified general solution
- For factor analysis and probabilistic PCA, these reduce to *weighted linear regression* problems

$$-\ln p(x, z \mid \theta) = C + \frac{1}{2} \sum_{i=1}^{N} \left[||z_i||^2 + D \log \sigma^2 + \sigma^{-2} ||x_i - W z_i - \mu||^2 \right]$$

$$\Psi = \sigma^2 I$$

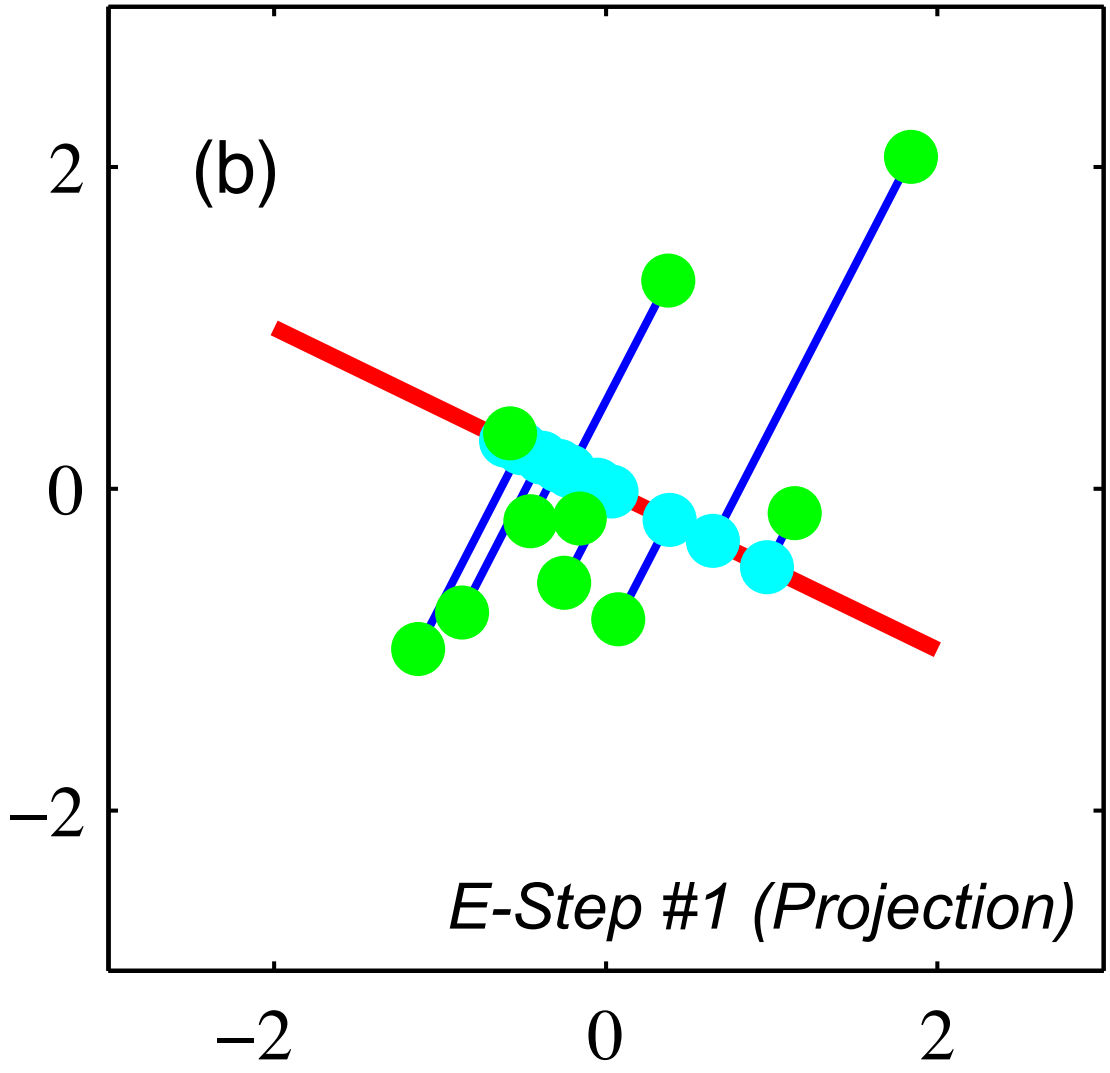
$$\min_{W, \mu, \Psi} \frac{1}{2} \sum_{i=1}^{N} \left[D \log \sigma^2 + \sigma^{-2} \mathbb{E}_q [||x_i - W z_i - \mu||^2] \right]$$

EM Algorithm for Probabilistic PCA



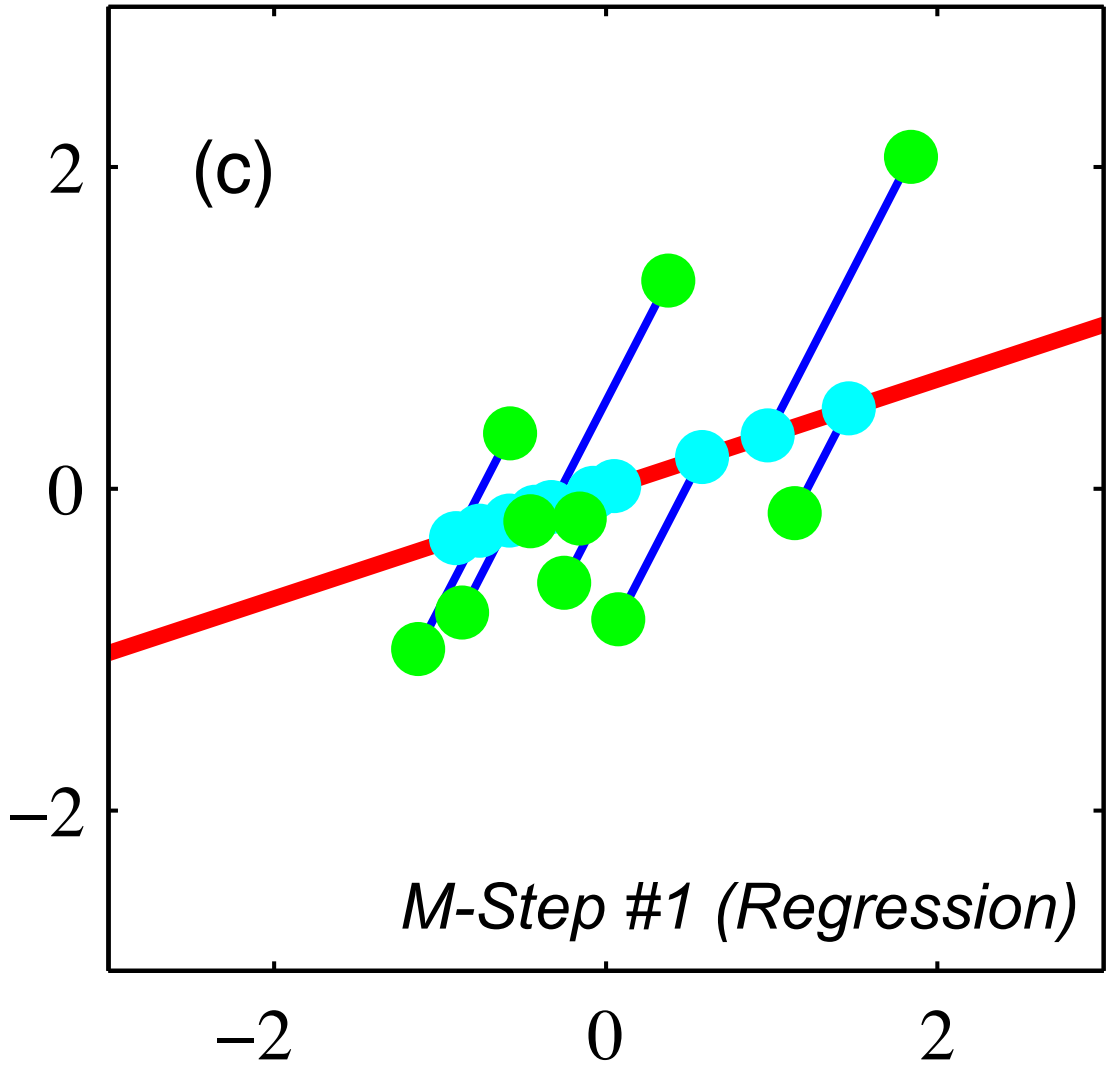
C. Bishop, Pattern Recognition & Machine Learning

EM Algorithm for Probabilistic PCA



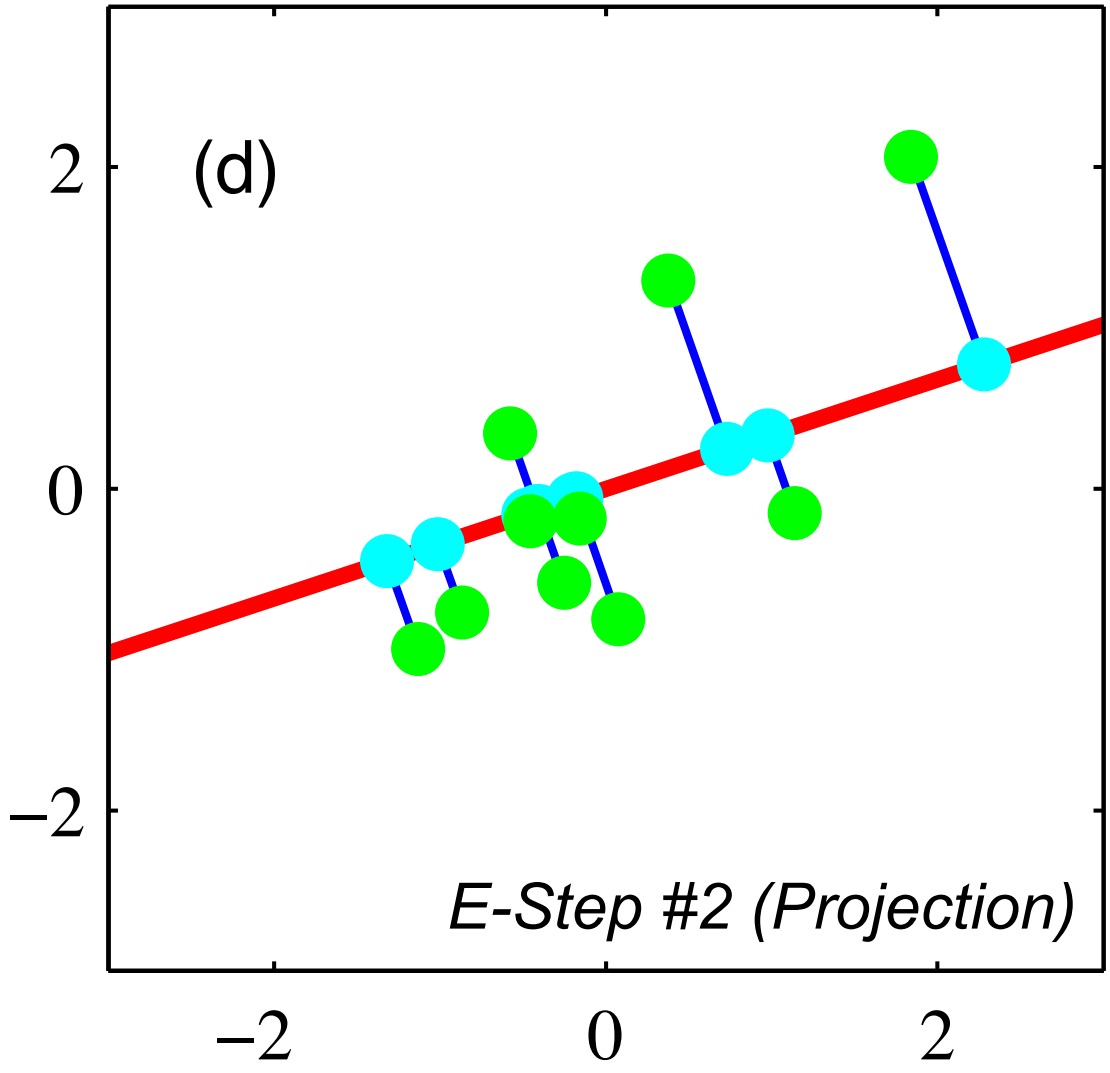
C. Bishop, Pattern Recognition & Machine Learning

EM Algorithm for Probabilistic PCA



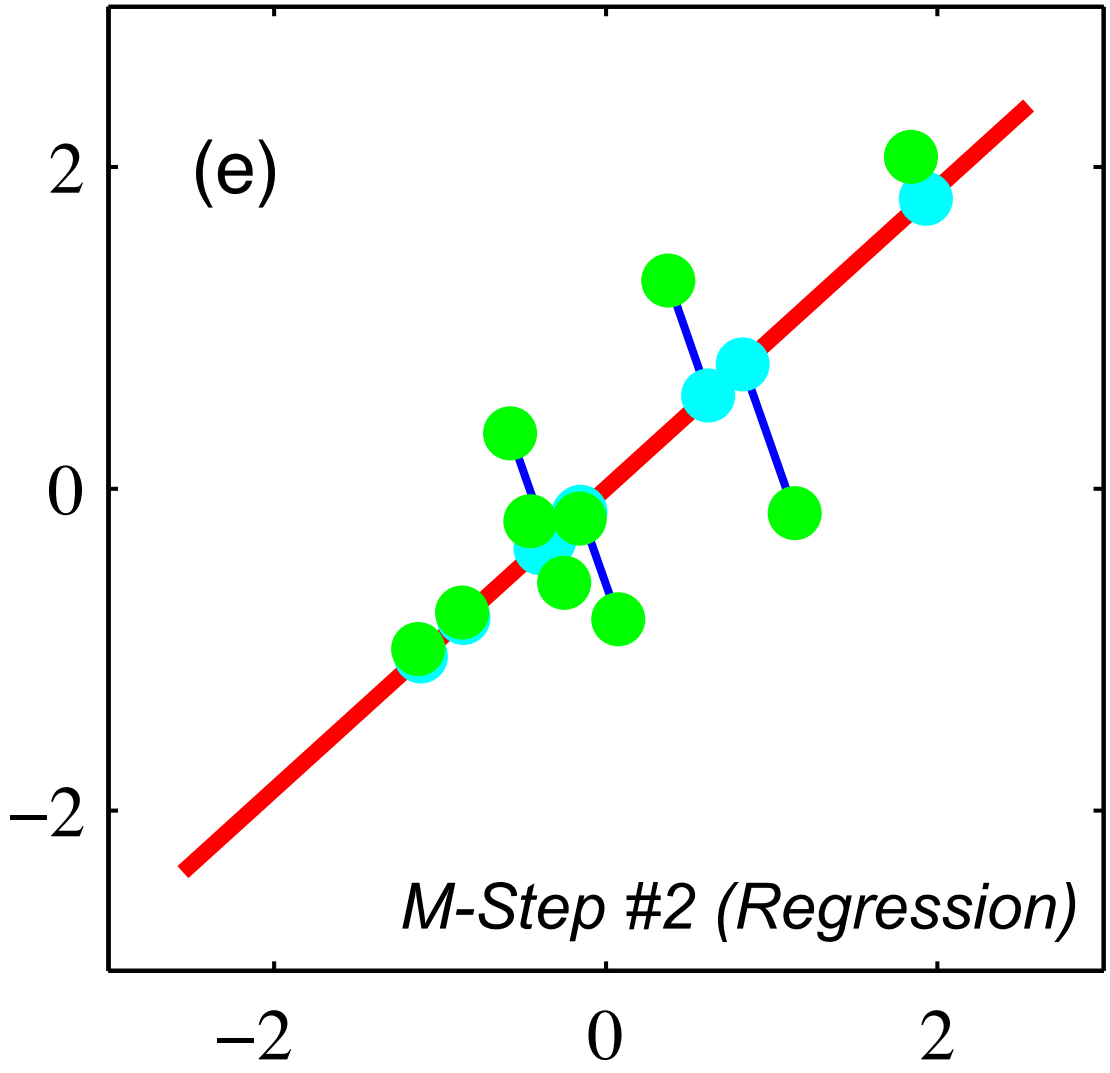
C. Bishop, Pattern Recognition & Machine Learning

EM Algorithm for Probabilistic PCA



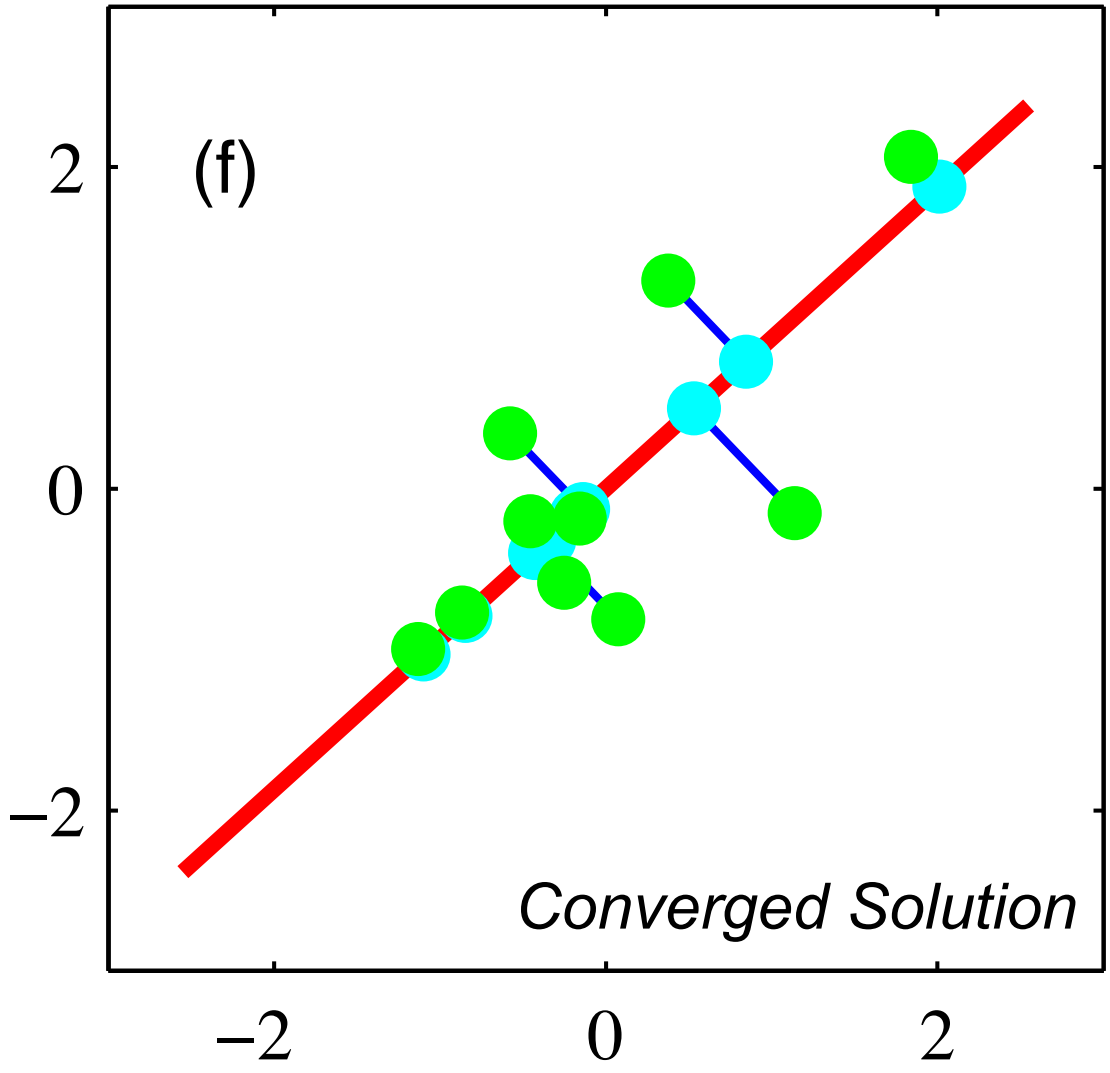
C. Bishop, Pattern Recognition & Machine Learning

EM Algorithm for Probabilistic PCA



C. Bishop, Pattern Recognition & Machine Learning

EM Algorithm for Probabilistic PCA



C. Bishop, Pattern Recognition & Machine Learning

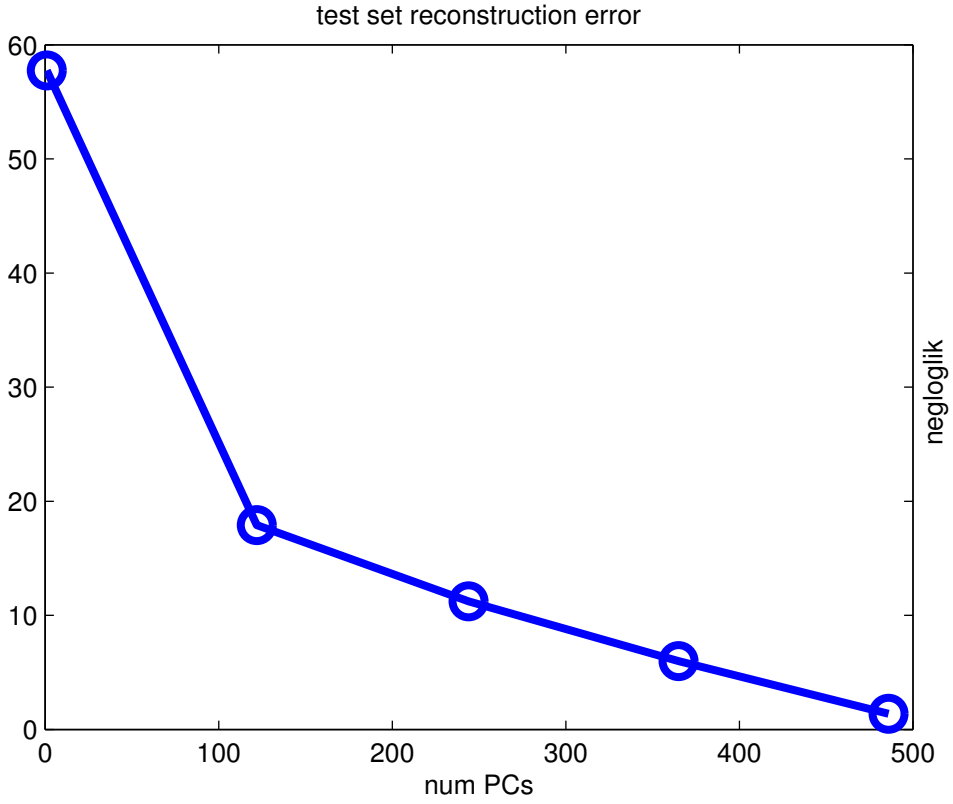
Why use the EM Algorithm for PCA?

- For large datasets, can be more computationally efficient than an eigendecomposition or SVD
- Regularization: can put priors on model parameters, do Bayesian model order selection, etc.
- Cleanly handles cases where some entries of the data matrix are unobserved or missing (e.g., movie ratings)
- Generalizes to other models where there is no closed form for the maximum likelihood estimates (e.g., factor analysis)

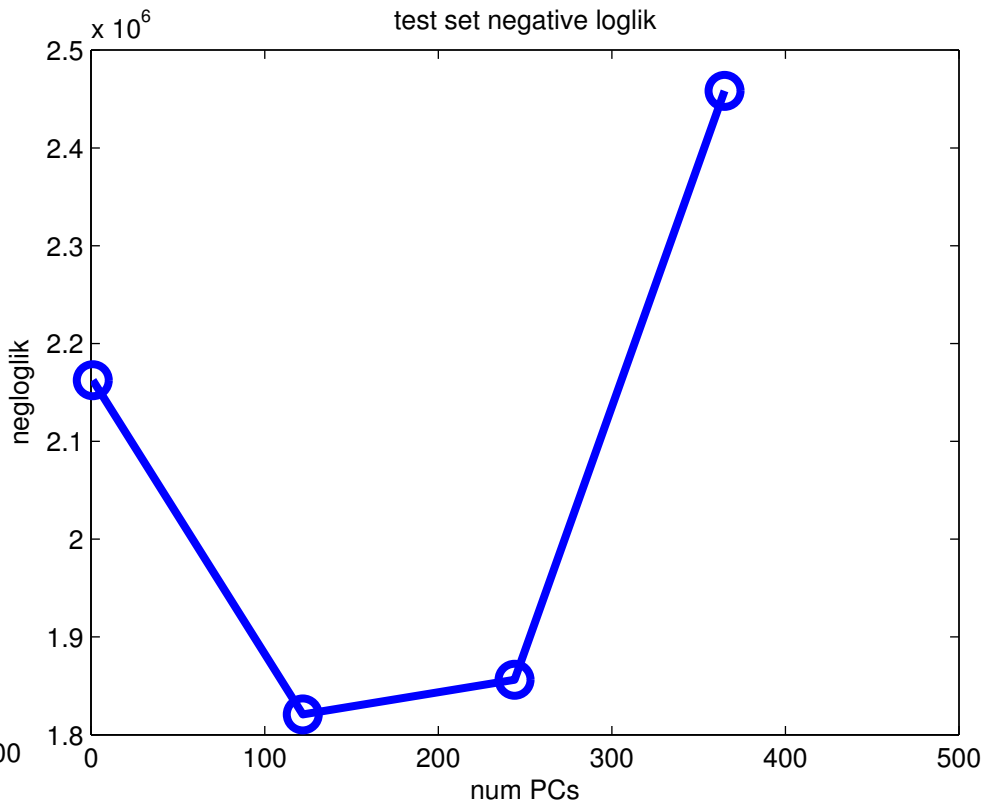
Probabilistic PCA or Factor Analysis?

- Probabilistic PCA models all rotations of the input data equally well (are basis vectors meaningful?)
- Factor analysis models all element-wise rescalings of the input data equally well (better when varying units)

Prediction of Validation Data



Standard PCA
(reconstruction error)



Probabilistic PCA
(negative log likelihood)