Gaussian Models

ddebarr@uw.edu

2016-04-28

Agenda

- Introduction
- Gaussian Discriminant Analysis
- Inference
- Linear Gaussian Systems
- The Wishart Distribution
- Inferring Parameters

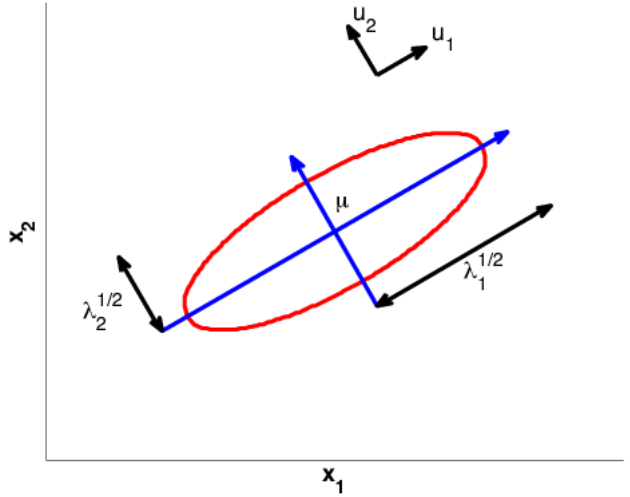
Gaussian Density Function

- MultiVariate Normal (MVN) Probability Density Function (PDF)

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \triangleq \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

Visualization of a 2D Gaussian Density

Based on spectral decomposition of the covariance matrix



Maximum Likelihood Estimate for Parameters

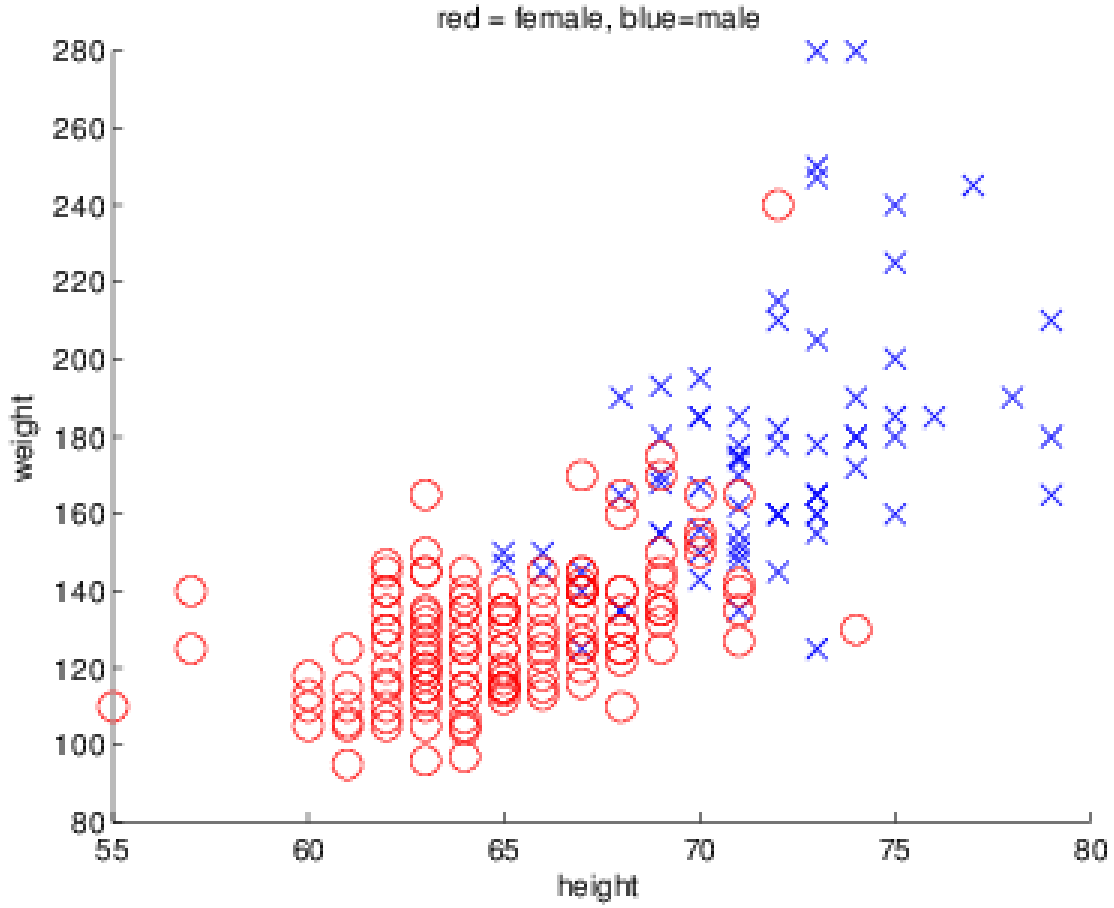
$$\hat{\boldsymbol{\mu}}_{mle} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \triangleq \overline{\mathbf{x}}$$

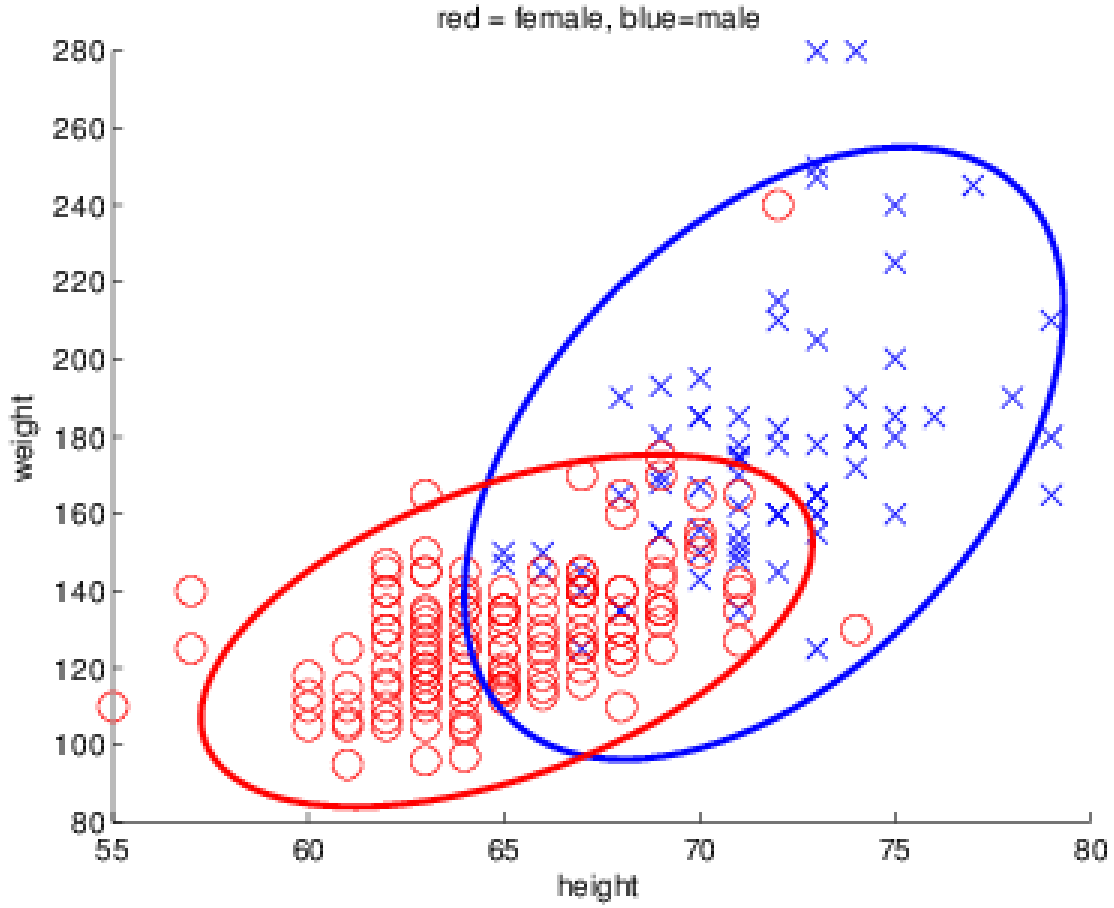
$$\hat{\boldsymbol{\Sigma}}_{mle} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T = \frac{1}{N}(\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^T) - \overline{\mathbf{x}}\,\overline{\mathbf{x}}^T$$

Reminder for reading:

To maximize a function f(), we solve f’(x) = 0 for the value of x

Gaussian Functions Fitted to Data





Maximum Entropy Interpretation of Gaussian

- The multivariate Gaussian is the distribution with maximum entropy, subject to the constraints that it has a specified mean and covariance
  - Maximum entropy means fewer assumptions

Gaussian Discriminant Analysis

- A discriminant is a function that can be used to distinguish between members of different classes
- Gaussian discriminant analysis uses the Gaussian distribution for the class conditional density function

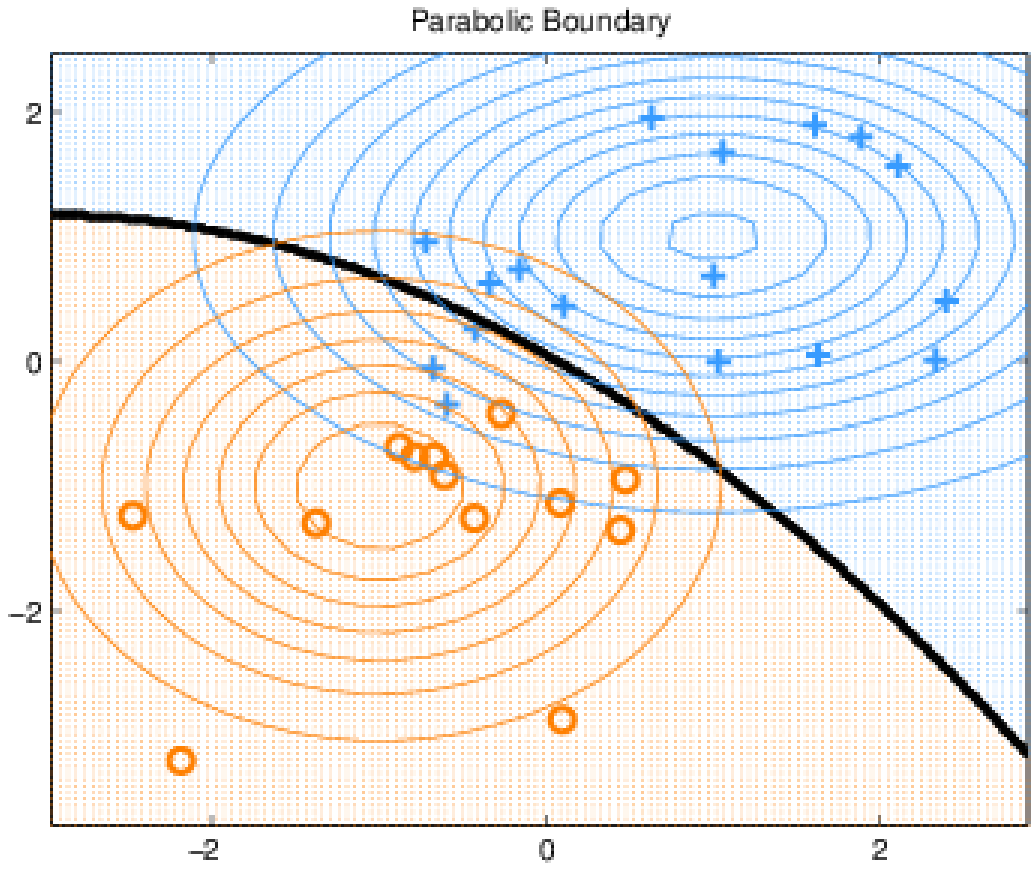
$$p(\mathbf{x}|y=c,\boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$$

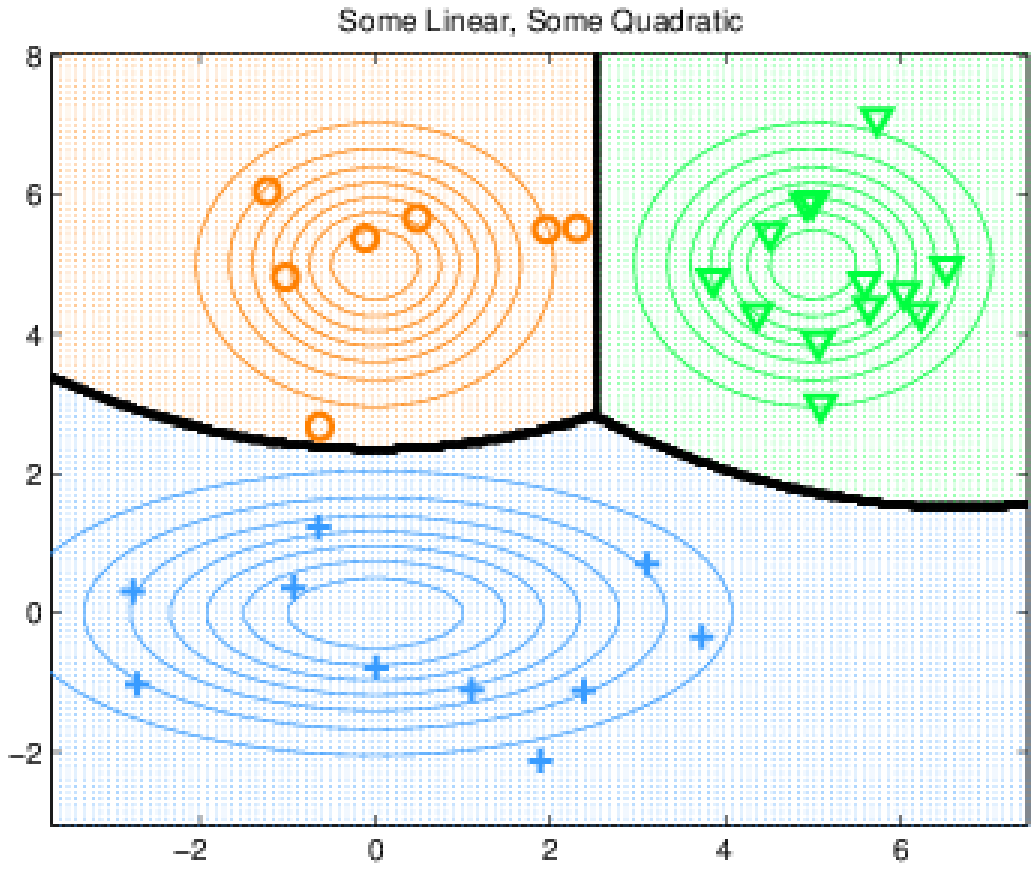
Quadratic Discriminant Analysis (QDA)

- Called quadratic because it uses squared terms
- The decision boundary may not be linear [may not be a line in 2-dimensional space (or a hyperplane in n-dimensional space)]

$$p(y=c|\mathbf{x},\boldsymbol{\theta}) \quad = \quad \frac{\pi_c|2\pi\boldsymbol{\Sigma}_c|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^T\boldsymbol{\Sigma}_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c)\right]}{\sum_{c'}\pi_{c'}|2\pi\boldsymbol{\Sigma}_{c'}|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{c'})^T\boldsymbol{\Sigma}_{c'}^{-1}(\mathbf{x}-\boldsymbol{\mu}_{c'})\right]}$$

Examples of QDA Decision Boundaries





Linear Discriminant Analysis (LDA)

- The same model as QDA, except all classes use the same covariance matrix
- The decision boundary is linear: the quadratic term cancels out because it becomes independent of the class
- Note: there are two separate expansions for the LDA acronym
  - Linear Discriminant Analysis is used for classification
  - Latent Dirichlet Allocation is used for topic modeling (for text)

Linear Discriminant Analysis

$$p(y=c|\mathbf{x},\boldsymbol{\theta}) \propto \pi_c \exp\left[\boldsymbol{\mu}_c^T\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}\mathbf{x}^T\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_c^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c\right]$$

$$= \exp\left[\boldsymbol{\mu}_c^T\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_c^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c + \log\pi_c\right]\exp[-\frac{1}{2}\mathbf{x}^T\boldsymbol{\Sigma}^{-1}\mathbf{x}]$$

$$\boldsymbol{\beta}_c = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c$$

$$\gamma_c = -\frac{1}{2}\boldsymbol{\mu}_c^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c + \log\pi_c$$

$$p(y=c|\mathbf{x},\boldsymbol{\theta}) = \frac{e^{\boldsymbol{\beta}_c^T\mathbf{x}+\gamma_c}}{\sum_{c'} e^{\boldsymbol{\beta}_{c'}^T\mathbf{x}+\gamma_{c'}}} = \mathcal{S}(\boldsymbol{\eta})_c = \frac{e^{\eta_c}}{\sum_{c'=1}^{C} e^{\eta_{c'}}}$$

"Softmax" function

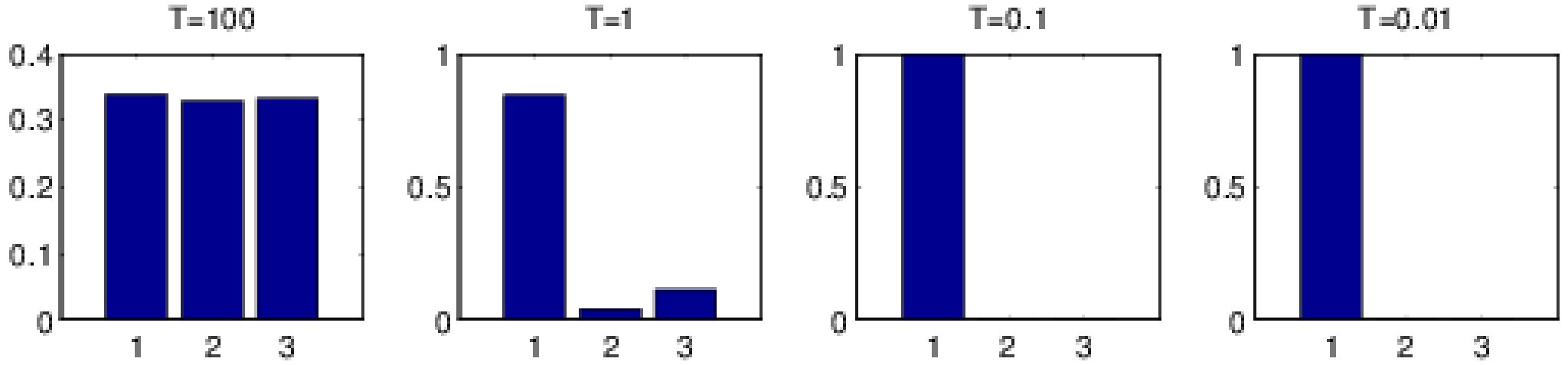
Softmax Distribution Note

When normalized by “temperature”, a “winner” emerges as temperature is reduced

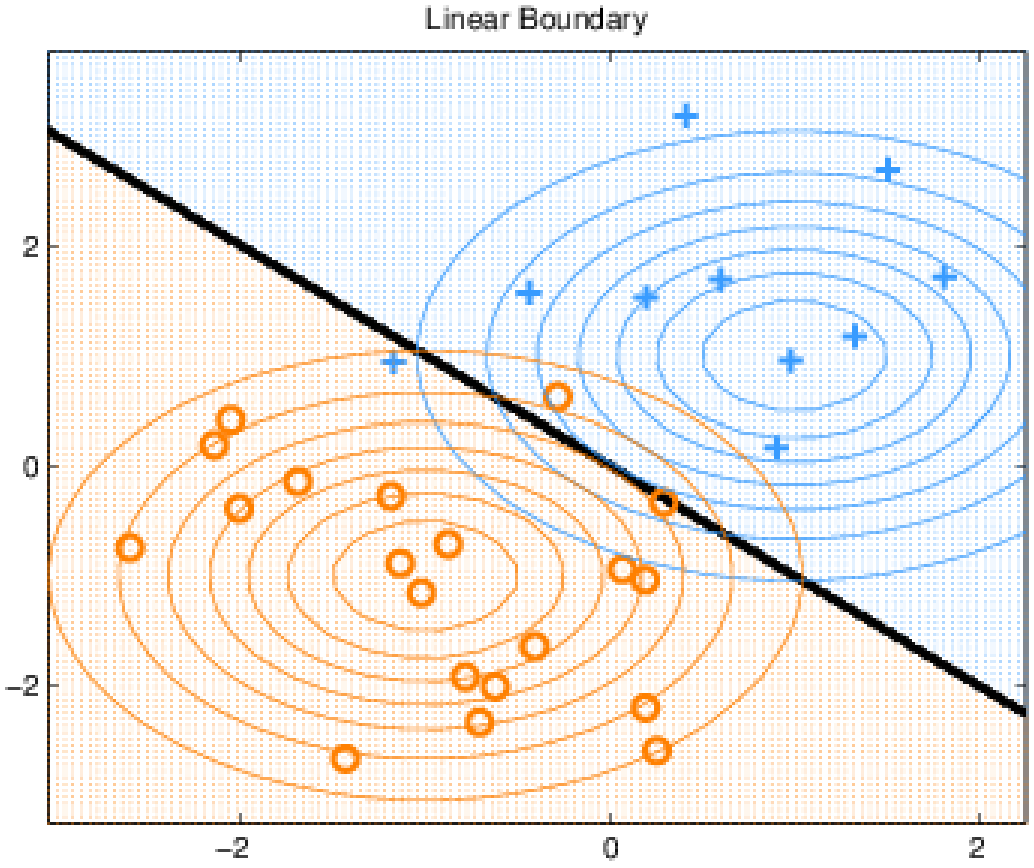
$$s\left(\vec{\eta}/_{T}\right)$$

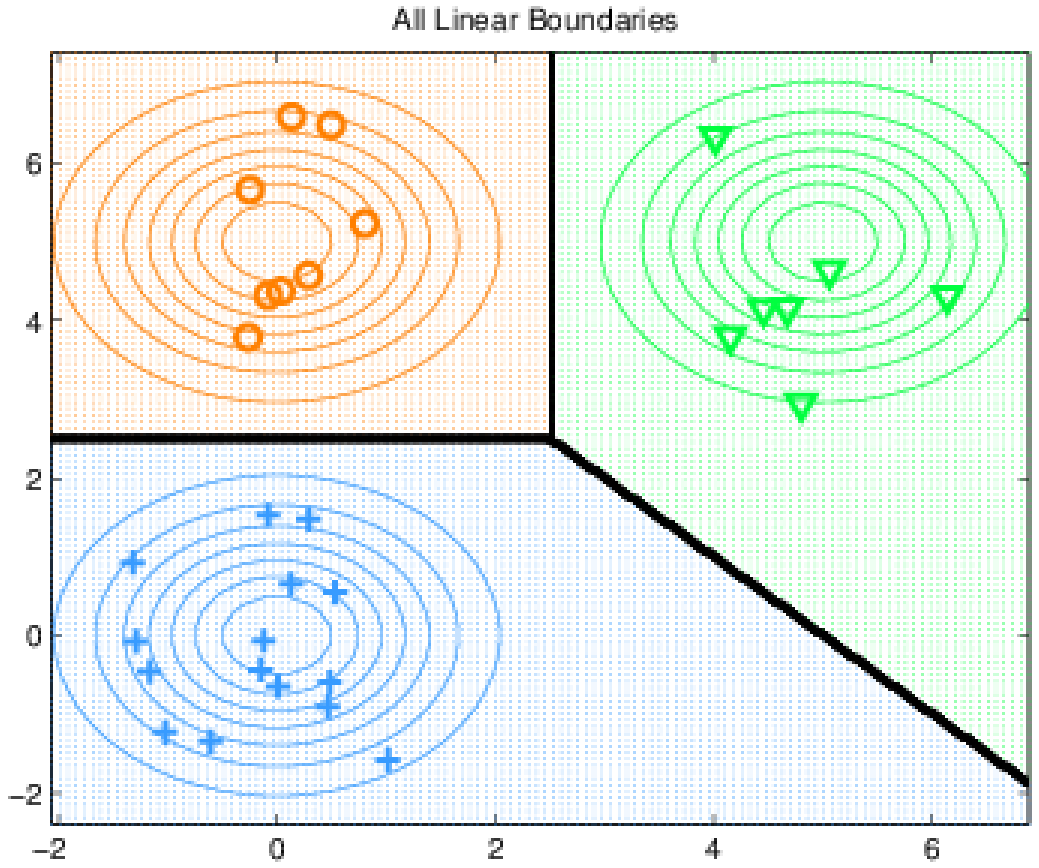
T is a constant called the temperature

$\vec{\eta} = [3, 0, 1]$

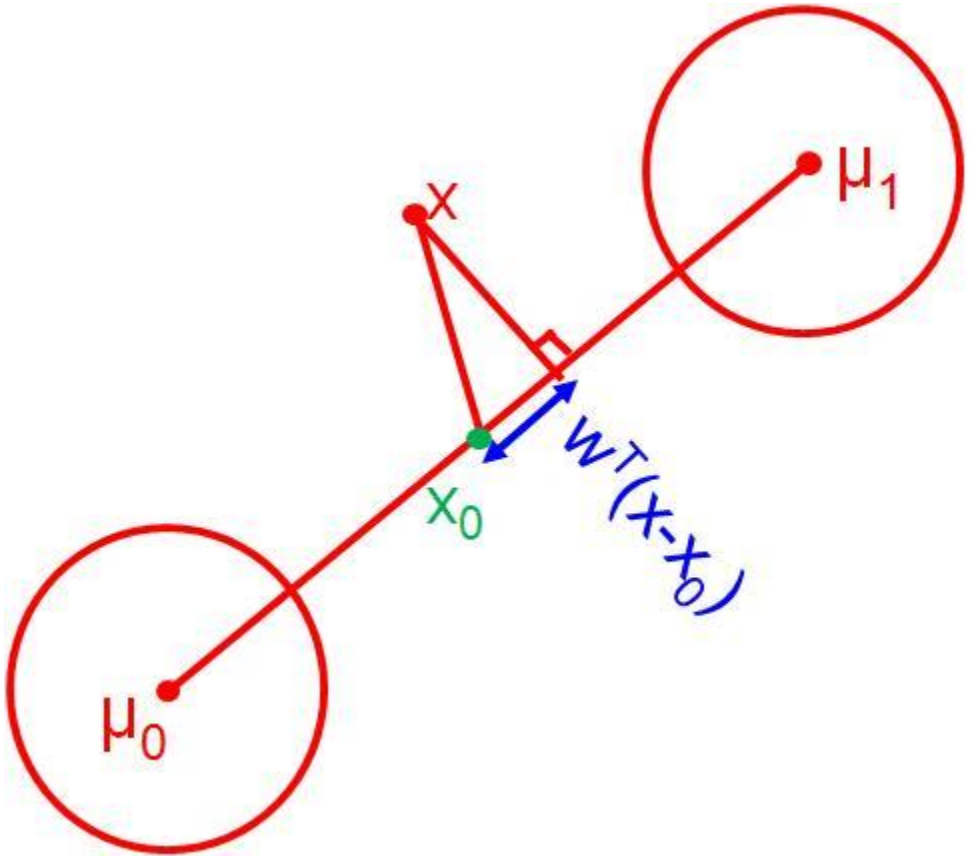


Examples of LDA Decision Boundaries





The Geometry of 2 Class LDA



Strategies to Avoid Overfitting

- Use a diagonal covariance matrix for each class [naïve Bayes]
- Use the same covariance matrix for all classes [LDA]
- Use a full covariance matrix, but impose a prior and then integrate it out [analogous to Bayesian naïve Bayes]
- Fit the covariance matrix using MAP estimation
- Project the data to a lower-dimension subspace and fit the Gaussians there

Regularized LDA

- MAP estimation of the covariance matrix

$$\hat{\boldsymbol{\Sigma}} = \lambda \text{diag}(\hat{\boldsymbol{\Sigma}}_{mle}) + (1 - \lambda)\hat{\boldsymbol{\Sigma}}_{mle}$$

- A larger value of lambda means reduced covariance (off diagonal entries shifting towards zero)

Diagonal LDA

A diagonal covariance matrix is used with pooled empirical variance

$$s_j^2 = \frac{\sum_{c=1}^{C} \sum_{i:y_i=c} (x_{ij} - \overline{x}_{cj})^2}{N - C}$$

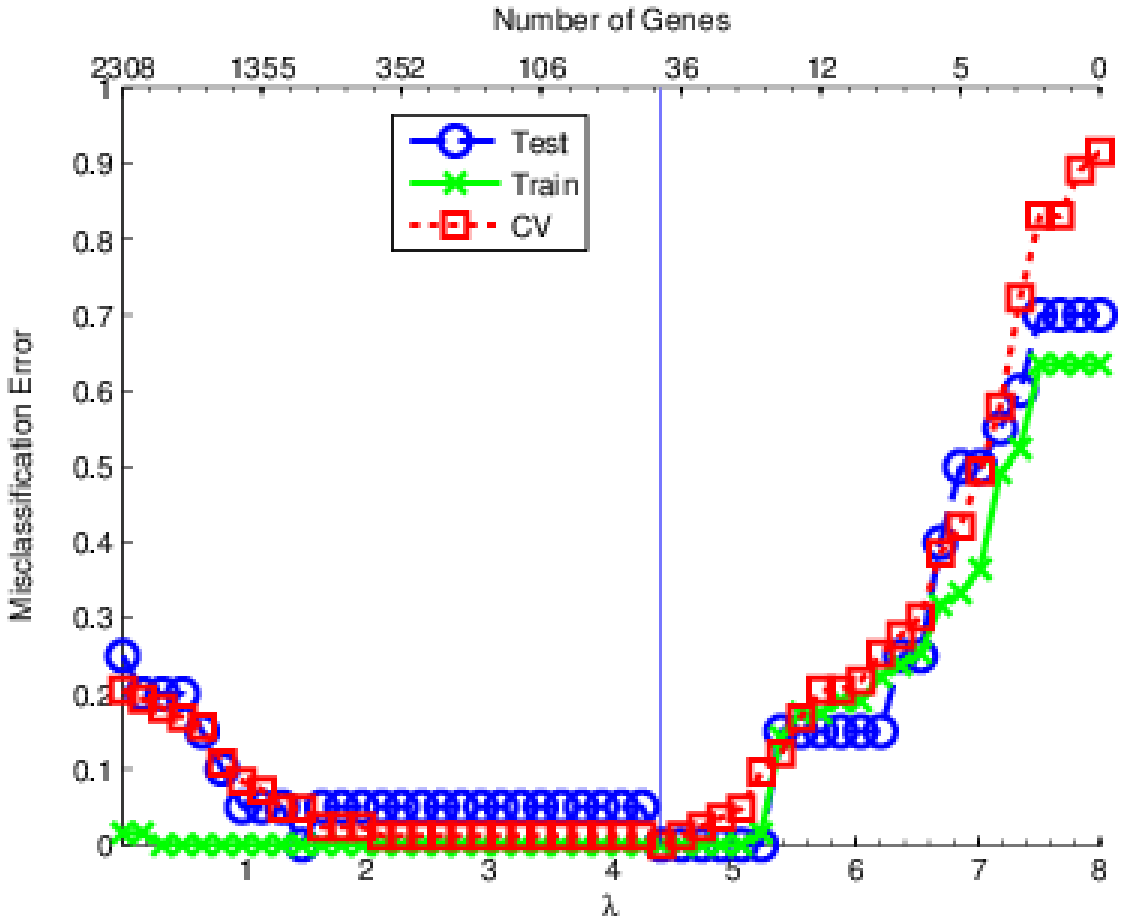
Nearest Neighbor Shrunken Centroids Classifier

$$\mu_{cj} = m_j + \Delta_{cj}$$

For feature ‘j’: ClassSpecificMean = GlobalMean + ClassSpecificOffset
Ignore features where ClassSpecificOffset is always zero

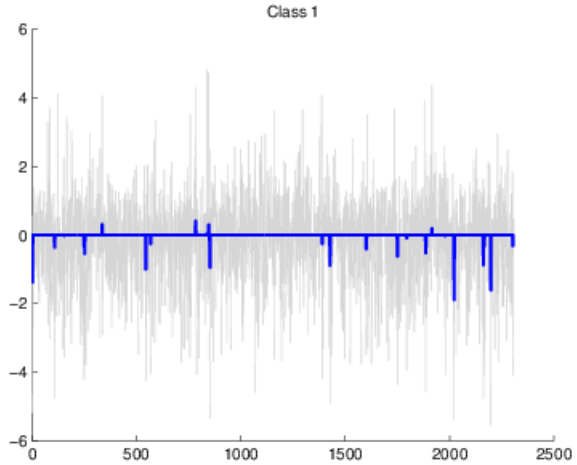
SBRCT: Small Blue Round Cell Tumor data

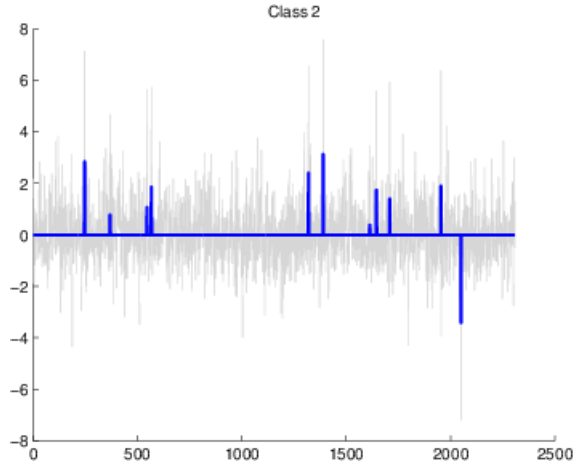
- 2,308 gene expression values
- 4 classes
- 63 training examples
- 20 testing examples

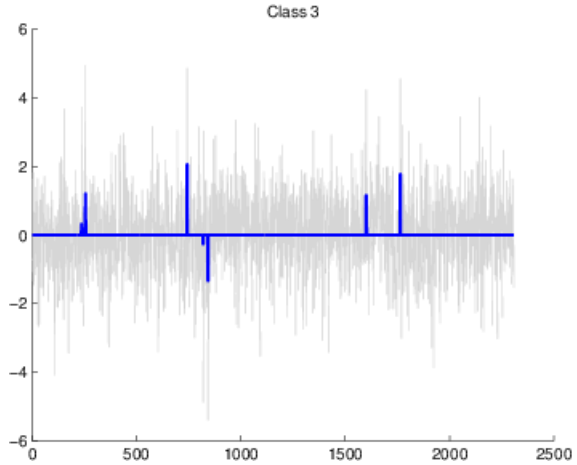


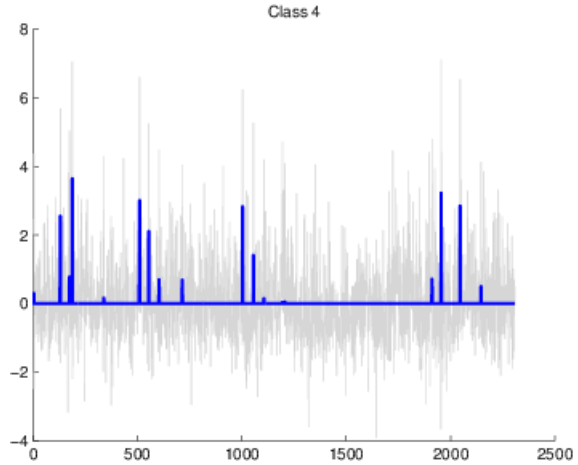
SRBCT Centroids [Gene Expression Values]

Gray: ClassSpecificMean
Blue: ClassSpecificOffset









Marginals and Posterior Conditionals

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}, \quad \boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{pmatrix} \boldsymbol{\Lambda}_{11} & \boldsymbol{\Lambda}_{12} \\ \boldsymbol{\Lambda}_{21} & \boldsymbol{\Lambda}_{22} \end{pmatrix}$$

$$p(\mathbf{x}_1) = \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11})$$

$$p(\mathbf{x}_2) = \mathcal{N}(\mathbf{x}_2 | \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22})$$

$$p(\mathbf{x}_1 | \mathbf{x}_2) = \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2})$$

$$\begin{aligned} \boldsymbol{\mu}_{1|2} &= \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= \boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{11}^{-1}\boldsymbol{\Lambda}_{12}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= \boldsymbol{\Sigma}_{1|2}\left(\boldsymbol{\Lambda}_{11}\boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{12}(\mathbf{x}_2 - \boldsymbol{\mu}_2)\right) \end{aligned}$$

$$\boldsymbol{\Sigma}_{1|2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21} = \boldsymbol{\Lambda}_{11}^{-1}$$

Reminder:
The variables on this page are vectors and matrices.

Example:
For posterior conditional, we may use observed variables to estimate the density for unobserved variables.

2D Example: Marginals and Posterior Conditionals

$$p(x_1) = \mathcal{N}(x_1|\mu_1, \sigma_1^2)$$

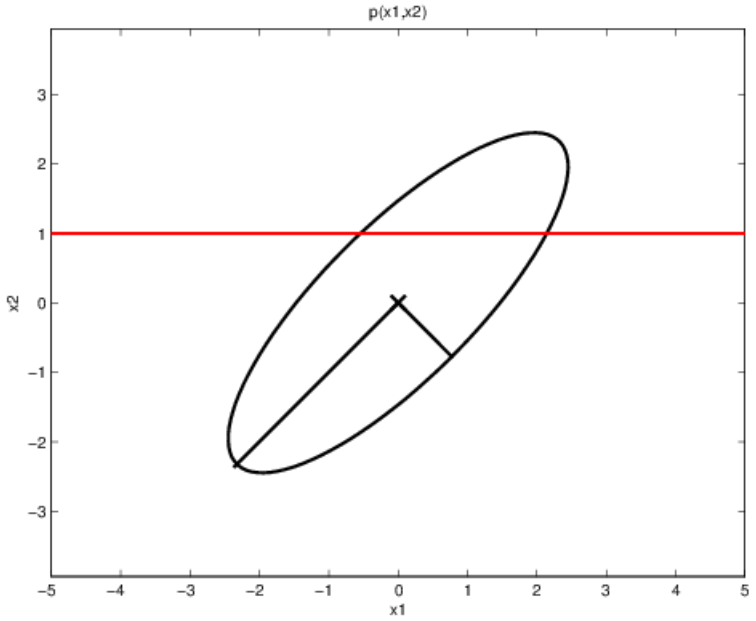
$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

$$p(x_1|x_2) = \mathcal{N}\left(x_1|\mu_1 + \frac{\rho\sigma_1\sigma_2}{\sigma_2^2}(x_2 - \mu_2),\ \sigma_1^2 - \frac{(\rho\sigma_1\sigma_2)^2}{\sigma_2^2}\right)$$

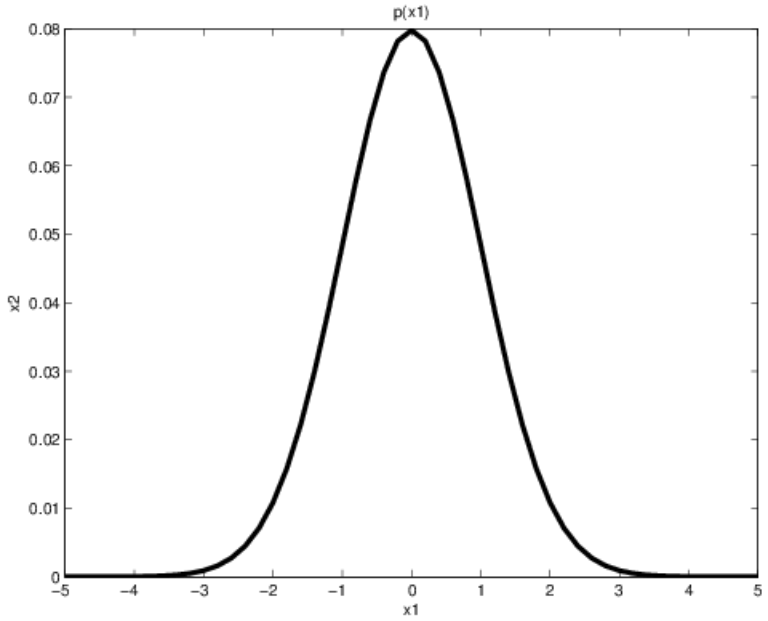
If $\sigma_1 = \sigma_2 = \sigma$, we get

$$p(x_1|x_2) = \mathcal{N}\left(x_1|\mu_1 + \rho(x_2 - \mu_2),\ \sigma^2(1 - \rho^2)\right)$$

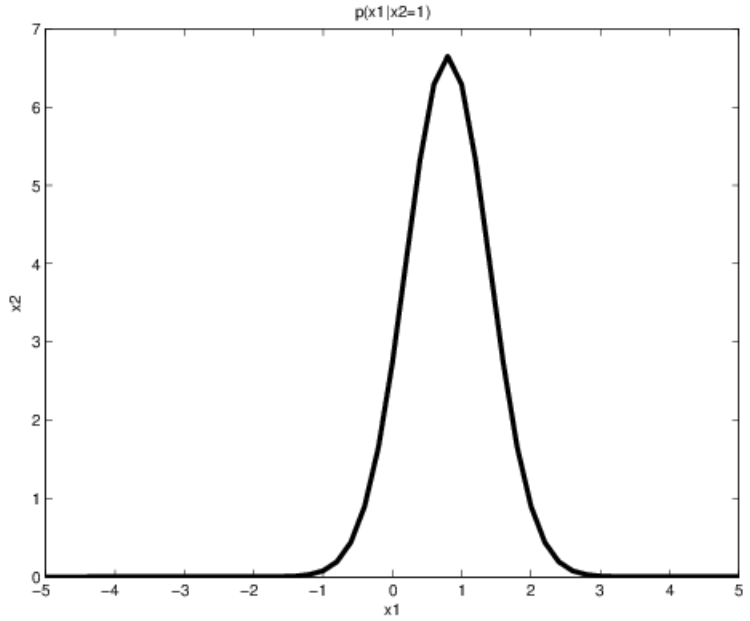
2D Example Visualized



Centered at (0, 0)
Correlation coefficient is 0.8



Marginal for x_1



Posterior for x_1 conditioned on x_2

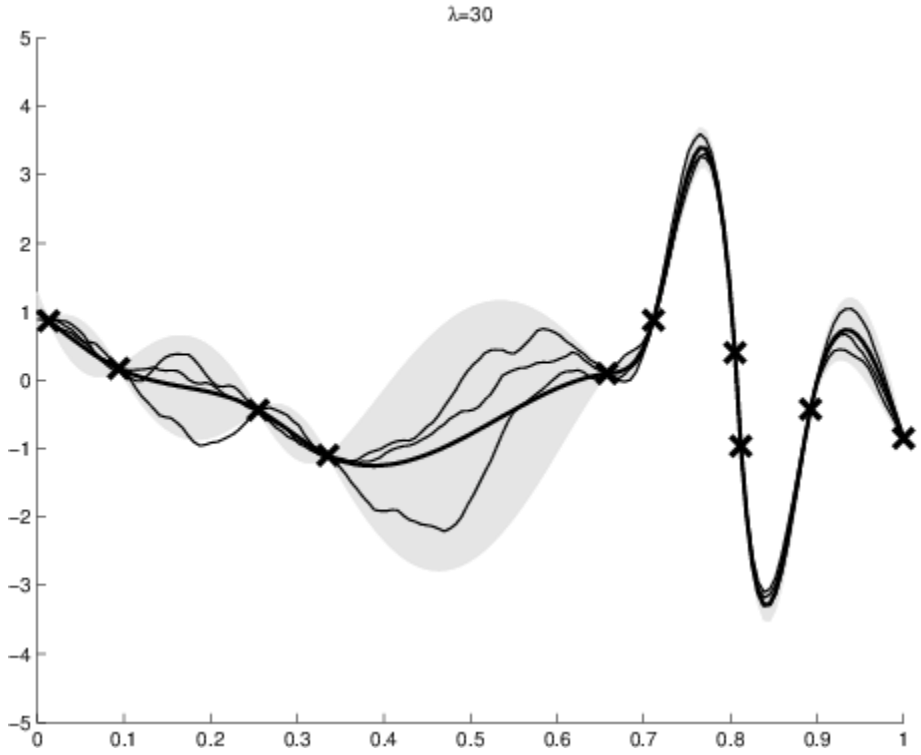
Inference: Interpolating Noise-Free Data

Smoothed estimates based on second order differencing.

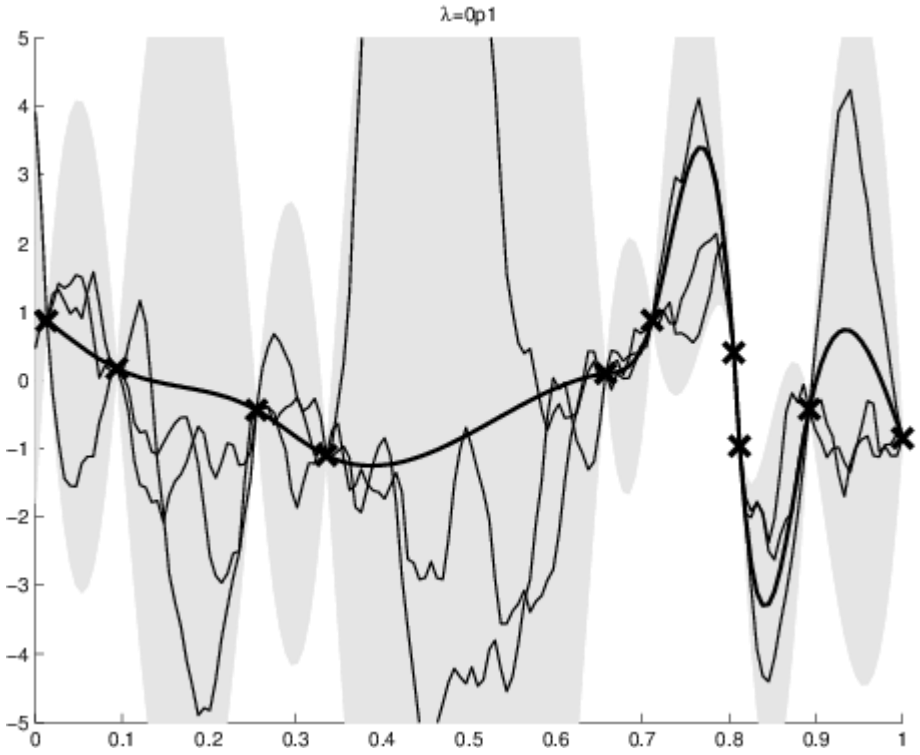
$$x_j = \frac{1}{2}(x_{j-1} + x_{j+1}) + \epsilon_j, \quad 2 \leq j \leq D-1$$

$$\mathbf{Lx} = \boldsymbol{\epsilon} \qquad \mathbf{L} = \frac{1}{2}\begin{pmatrix} -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & \ddots & & \\ & & & -1 & 2 & -1 \end{pmatrix}$$

Example: Interpolating Noise Free Data



Larger emphasis on prior

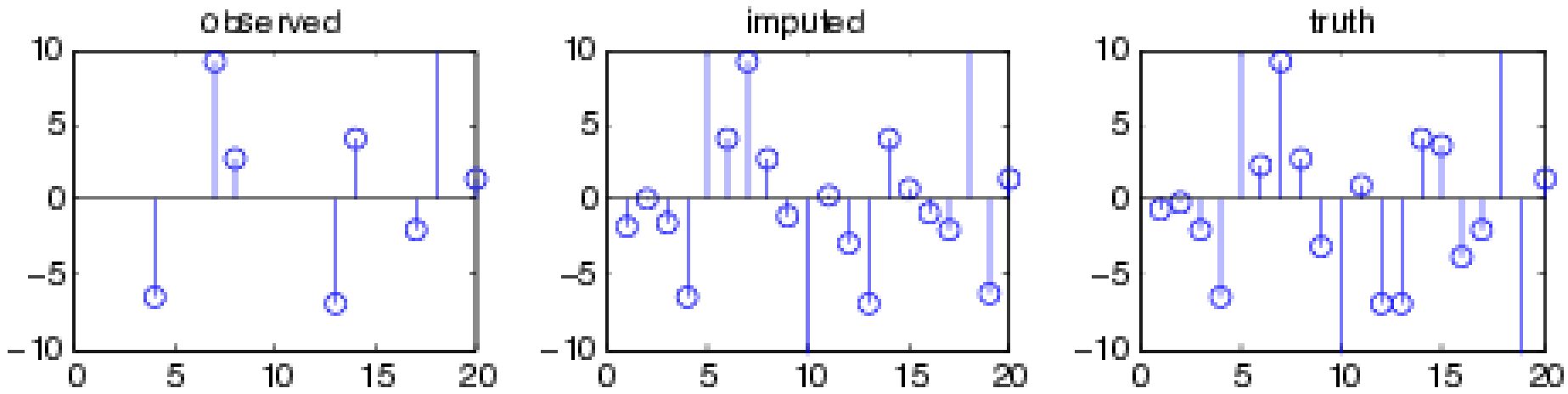


Smaller emphasis on likelihood

Noise free: we believe the sensors are accurate, so the line goes through the observed values.
Picture shows interval estimates for x_2 given a value for x_1 (larger as we move away from observed values).

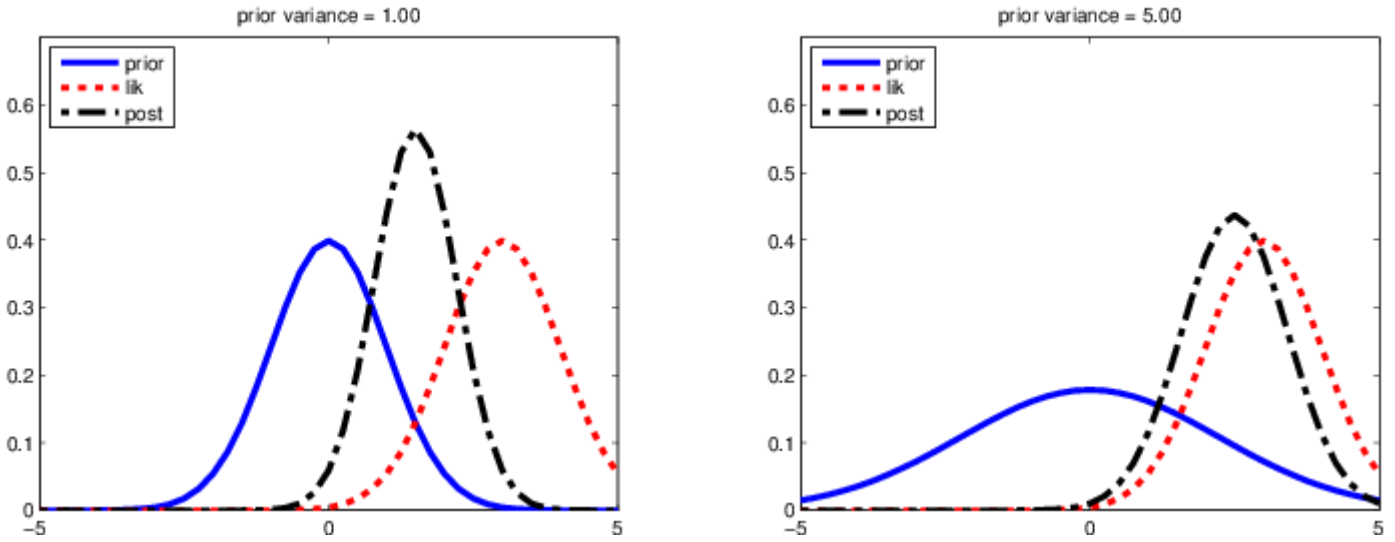
Inference: Data Imputation

Imputing missing values, based on parameter estimates for observed data



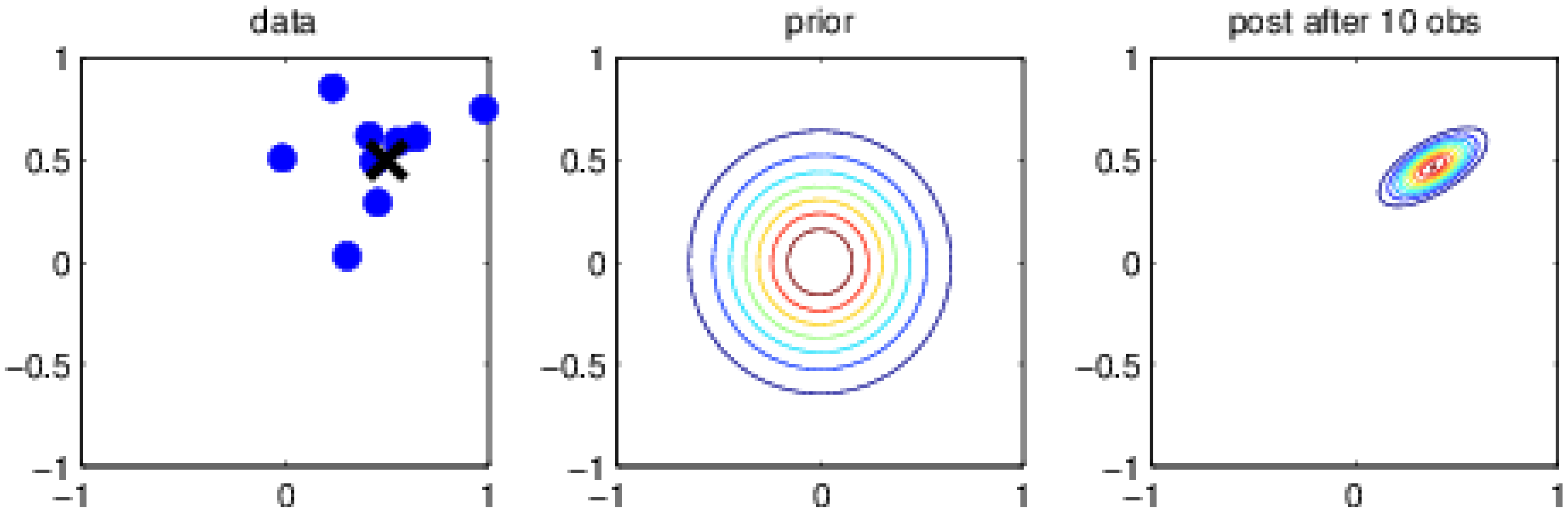
Another example where we're making use of a conditional density;
i.e. the density for the missing values given the observed values.
[20 dimensions!]

Inference About A Noisy 1D Observation



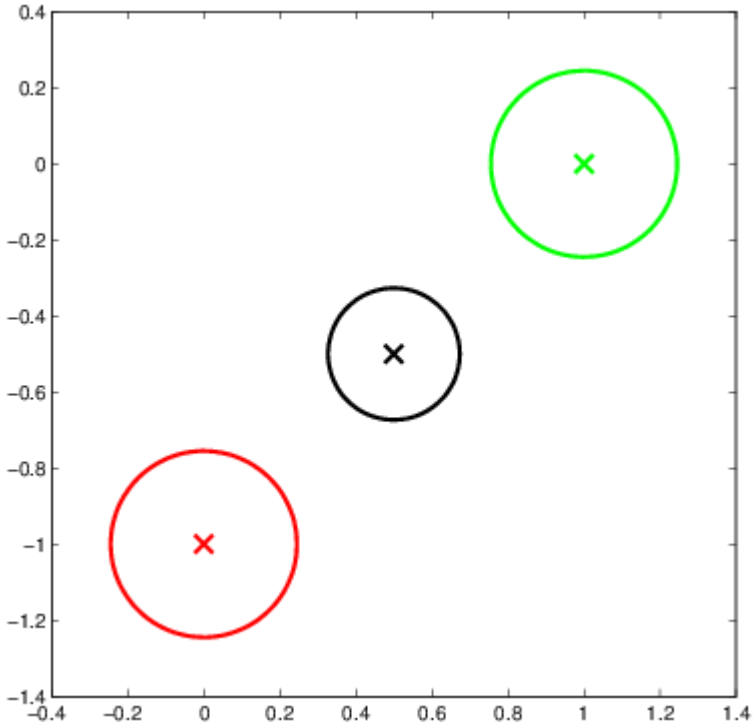
We say that a distribution with fatter tails is less precise.
On the left, we see a more precise prior pulls the posterior toward it.
On the right, we see a less precise prior has a reduced effect on the posterior.

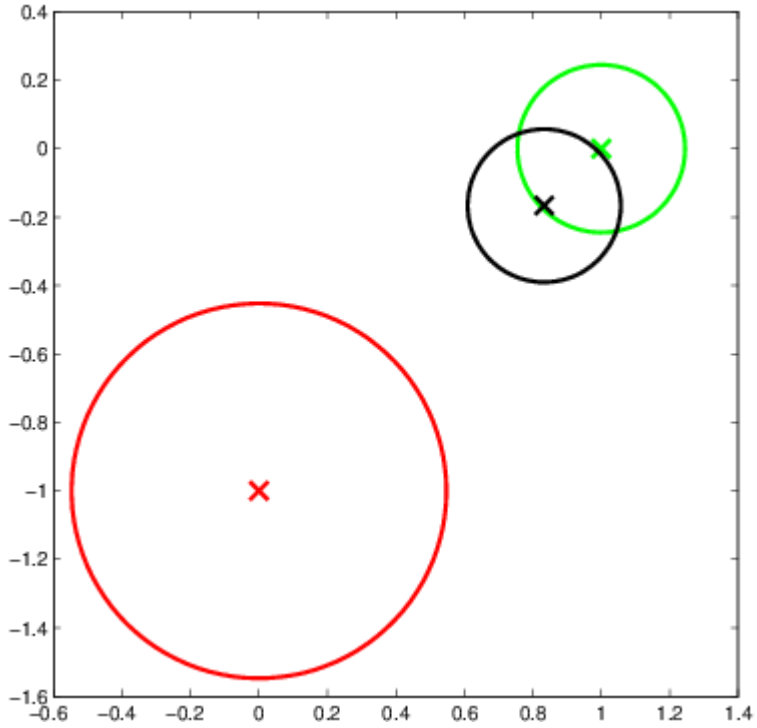
Inference About Noisy 2D Observations

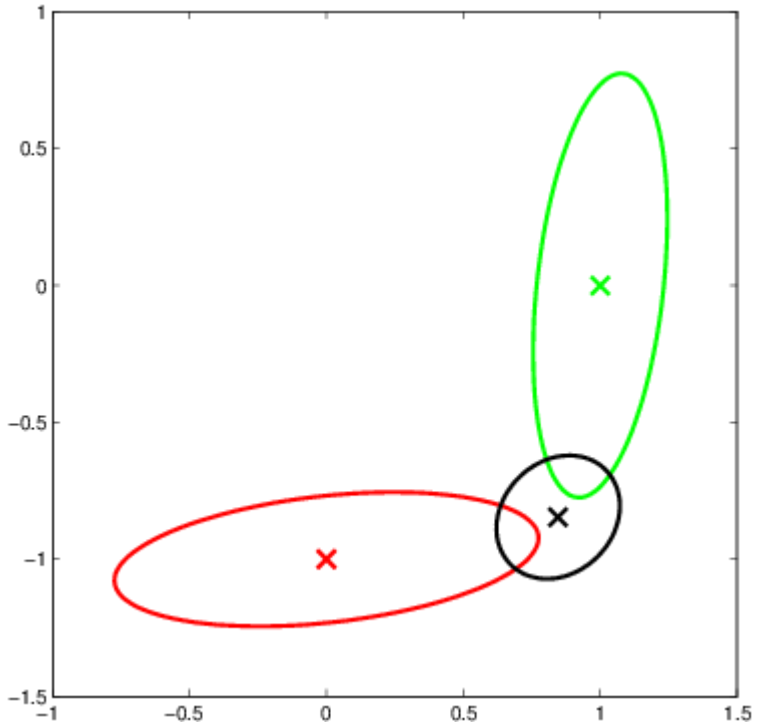


The prior (in the center) has a mean at (0, 0) and a variance of 0.1 for each dimension.
After only 10 observations, the posterior is more closely aligned with the data.
The “true” location is located at (0.5, 0.5).

Sensor Fusion

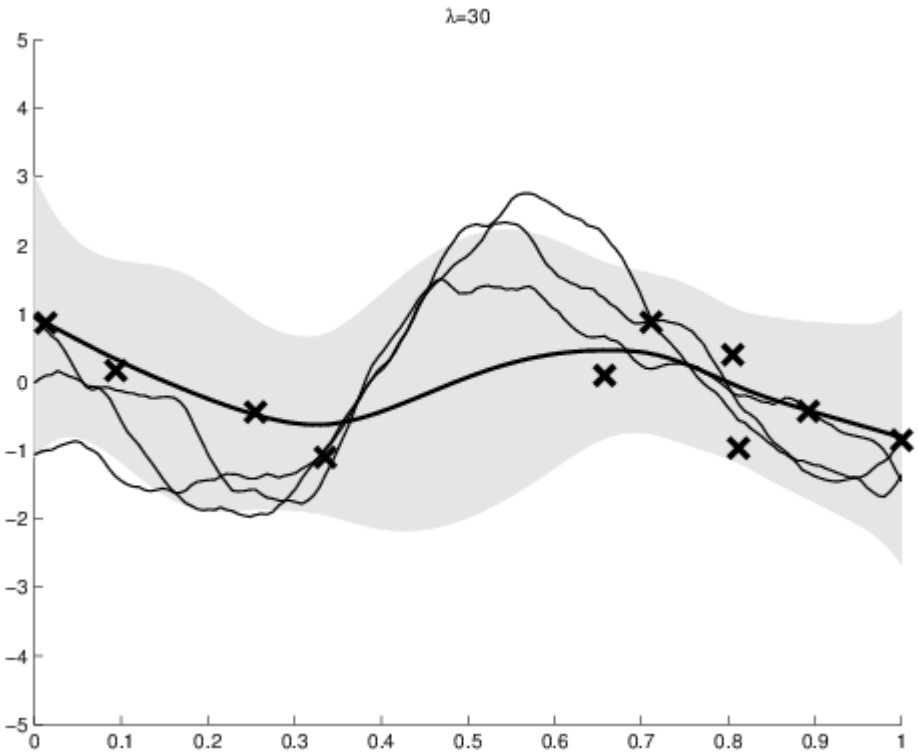


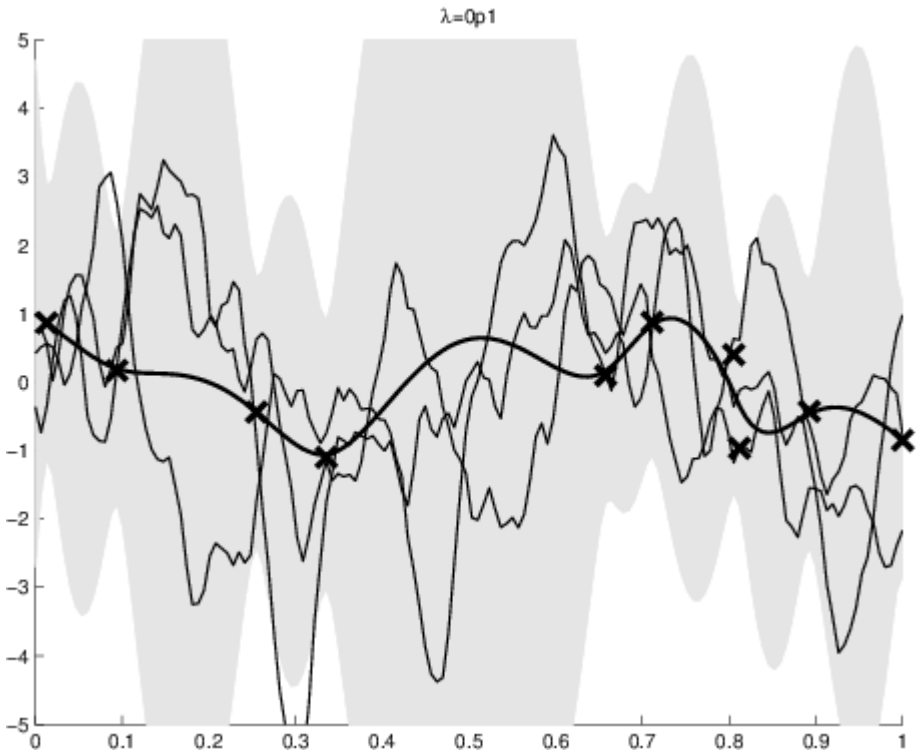




The contours reflect our uncertainty.
On the left, we are equally uncertain about the green and red sensors, so our posterior (black) is equidistant.
In the middle, we have more uncertainty about the red sensor, so our posterior is closer to the green.
On the right, we have more uncertainty about the first dimension for the red sensor and the second dimension for the green.

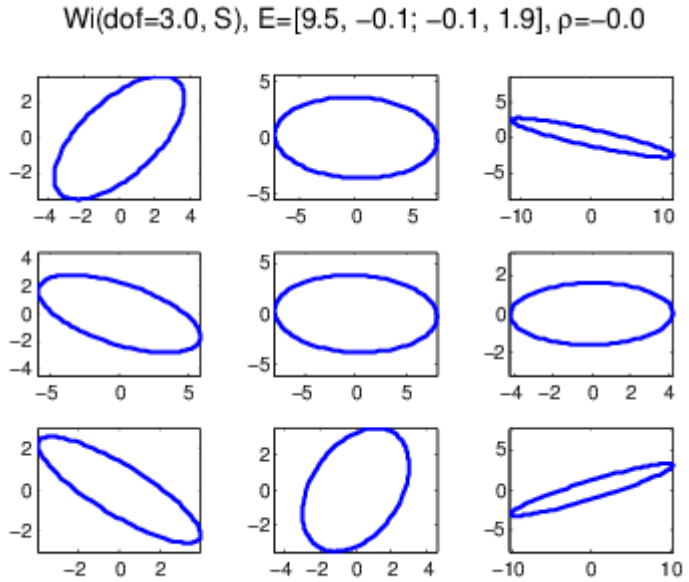
Interpolation with Noisy Data

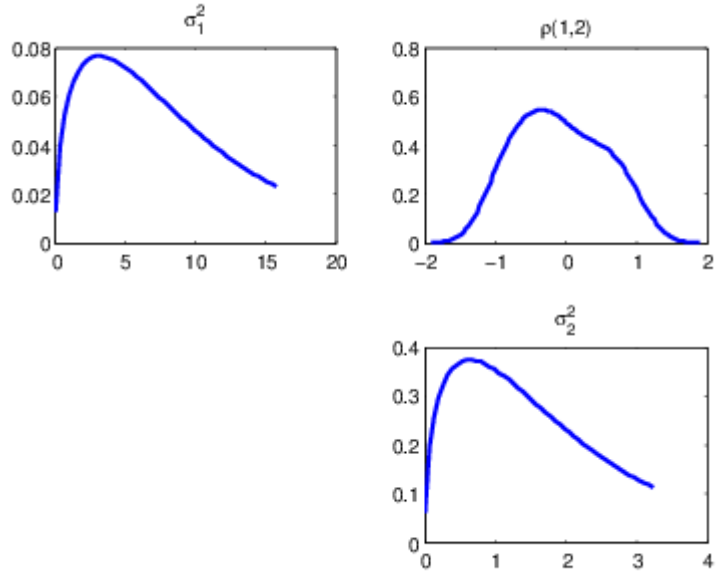




Again, the stronger prior appears on the left.
The line no longer passes through all observations,
reflecting our uncertainty about the observations.

Wishart Distribution for the Covariance Matrix

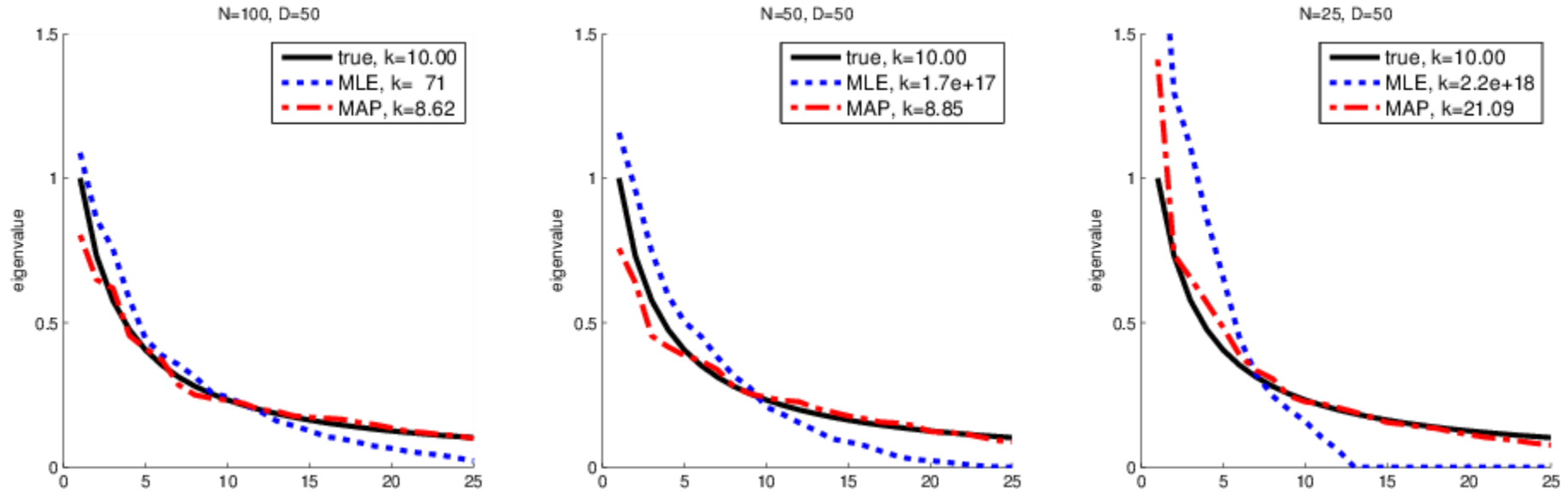




$$\mathbf{S} = [3.1653, -0.0262; -0.0262, 0.6477]$$

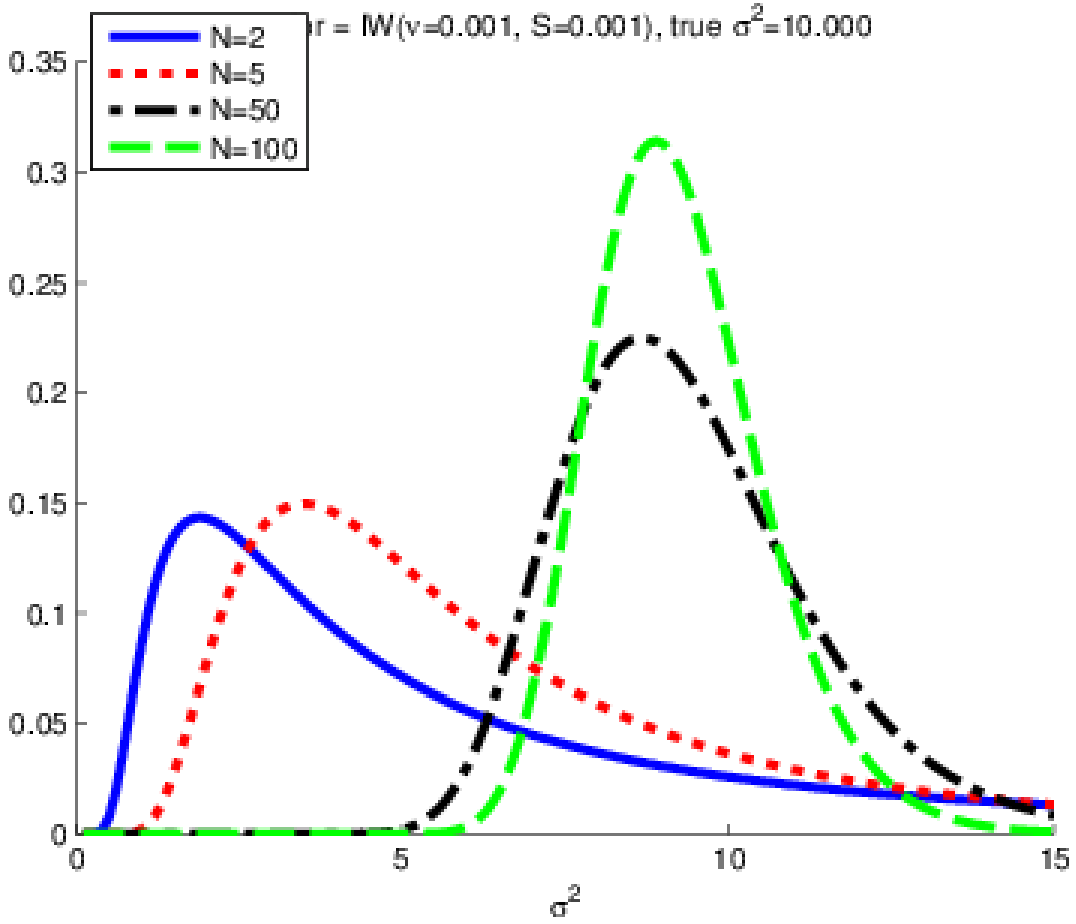
The Wishart distribution is a generalization of the gamma distribution.
It's used to model uncertainty about the covariance matrix parameter.

Covariance Matrix Estimation



Recall that the eigenvalue measures variation.
On the right, with fewer observations, we see the MAP estimate is better aligned to the true values than the MLE.
On the left, with more observations, we see the MLE is moving closer to the true values.

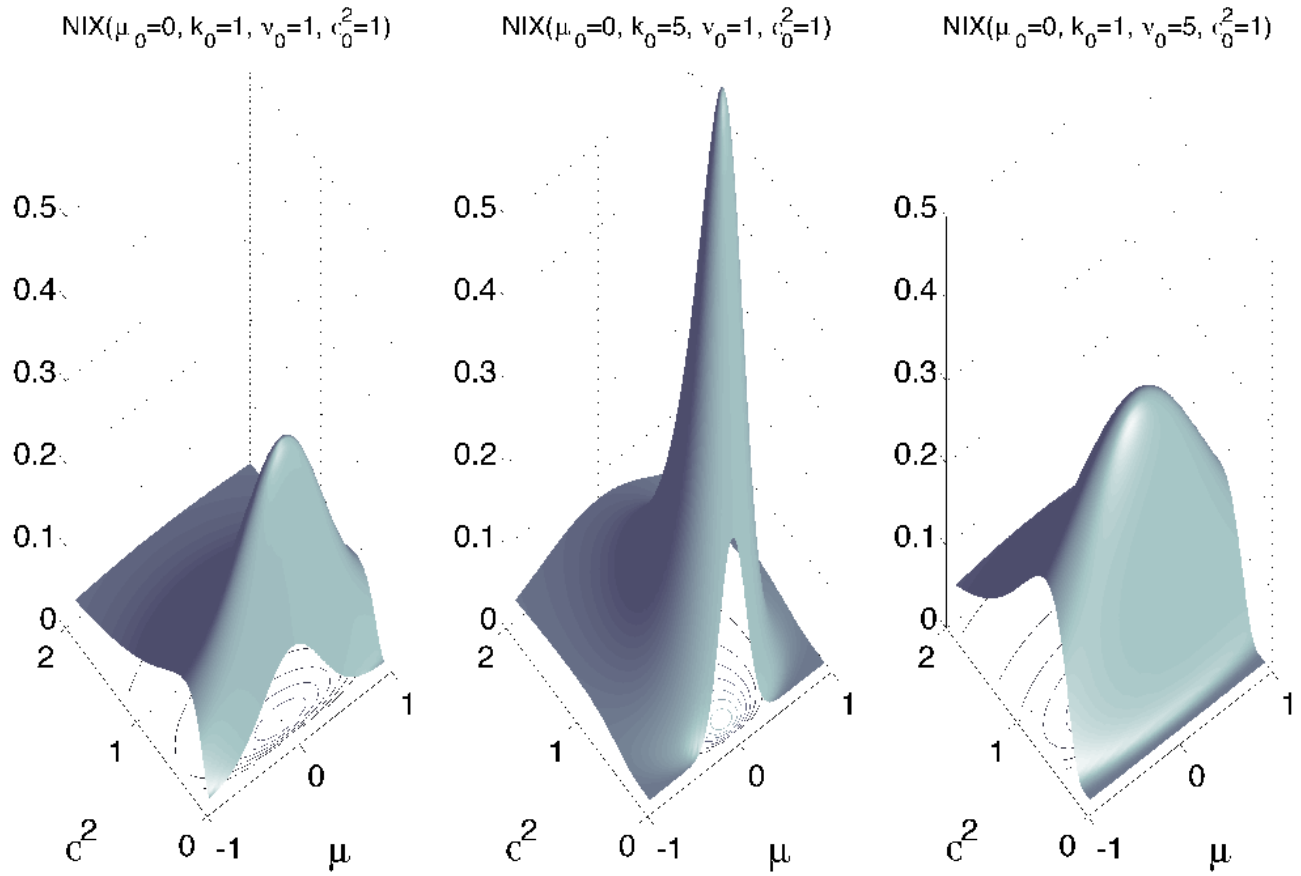
Sequential Updating of the Posterior for Variance



As we accumulate more evidence, our estimate moves closer to the true variance

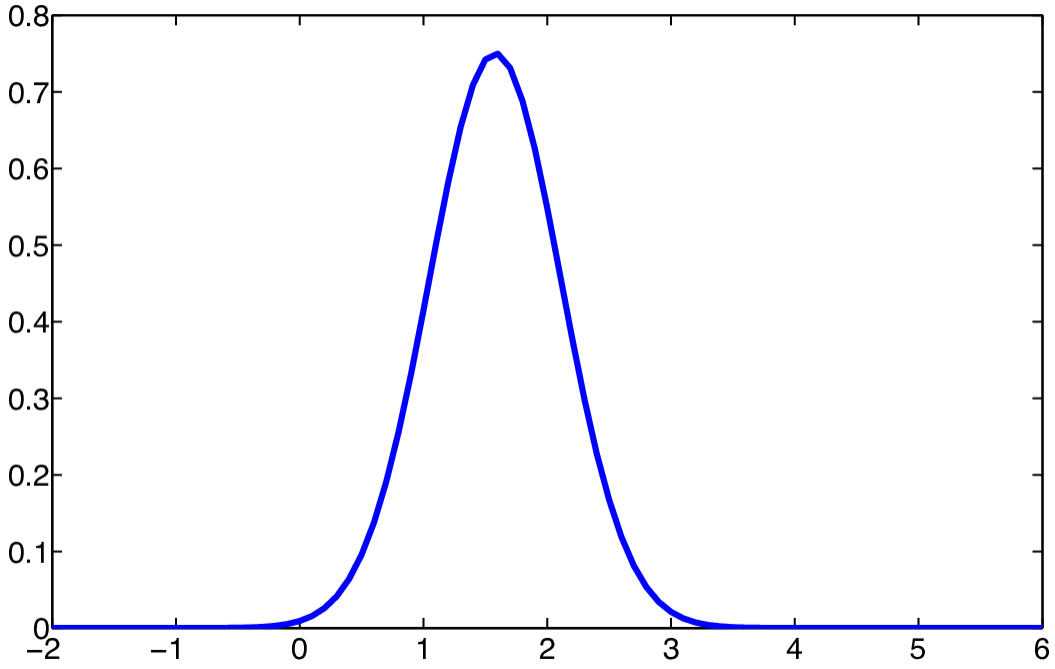
Effect of Prior on Parameter Estimates

Based on NIX: Normal Inverse Chi-Squared Distribution

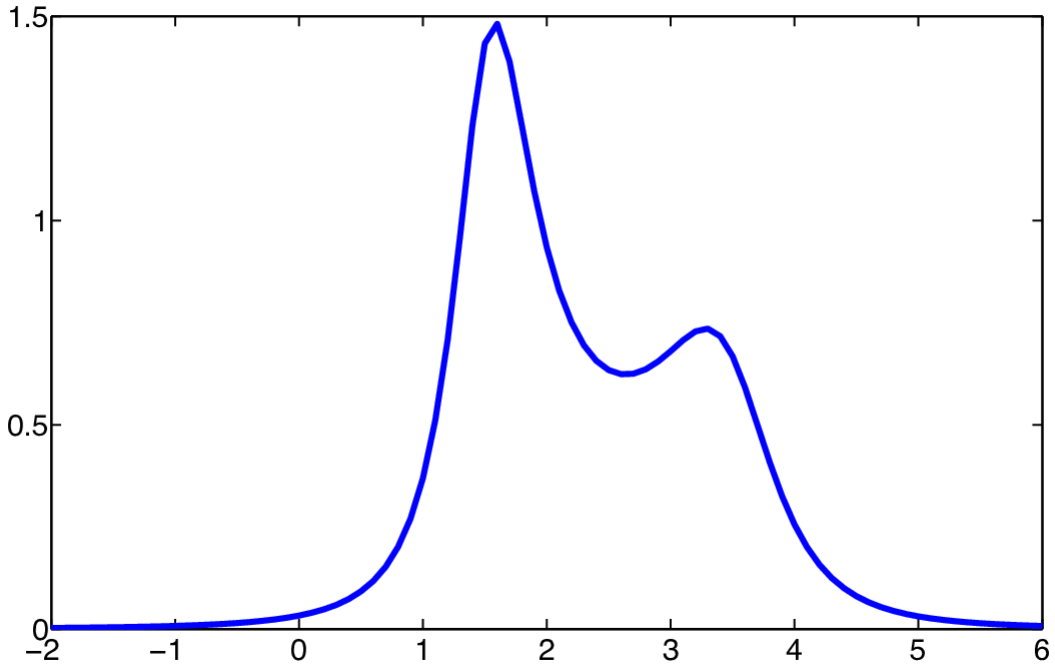


We're estimating the mean and variance for a one-dimensional Gaussian. On the left, our priors have less emphasis. In the middle, we're emphasizing our prior for the mean. On the right, we're emphasizing our prior for the variance.

Multi-Sensor Fusion Example



Plug-in Approximation:
Emphasizing the more reliable sensor



Exact Posterior:
Reflecting uncertainty about which sensor is correct

Natural Language Processing

Natural Language Processing

- Representation
- Latent Dirichlet Allocation

Applications

- Speech recognition
- Language translation
- Information retrieval
- Text classification (e.g. sentiment analysis)
- … anything where human language is an input …

Preprocessing Steps for Text

- Options
  - Convert text to lower-case
  - Remove stop words (e.g. “a”, “an”, “the”, …)
  - Stem tokens (e.g. remove suffixes like “ing”)
- Break up text into tokens (e.g. based on whitespace and punctuation)
- Derive ngrams (e.g. bigrams are pairs of adjacent words)
- Derive inverse document frequency values for training corpus

$$\text{idf}_t = \log \frac{N}{\text{df}_t}$$

idf produces larger weights for terms which appear in fewer documents and smaller weights for terms which appear in more documents

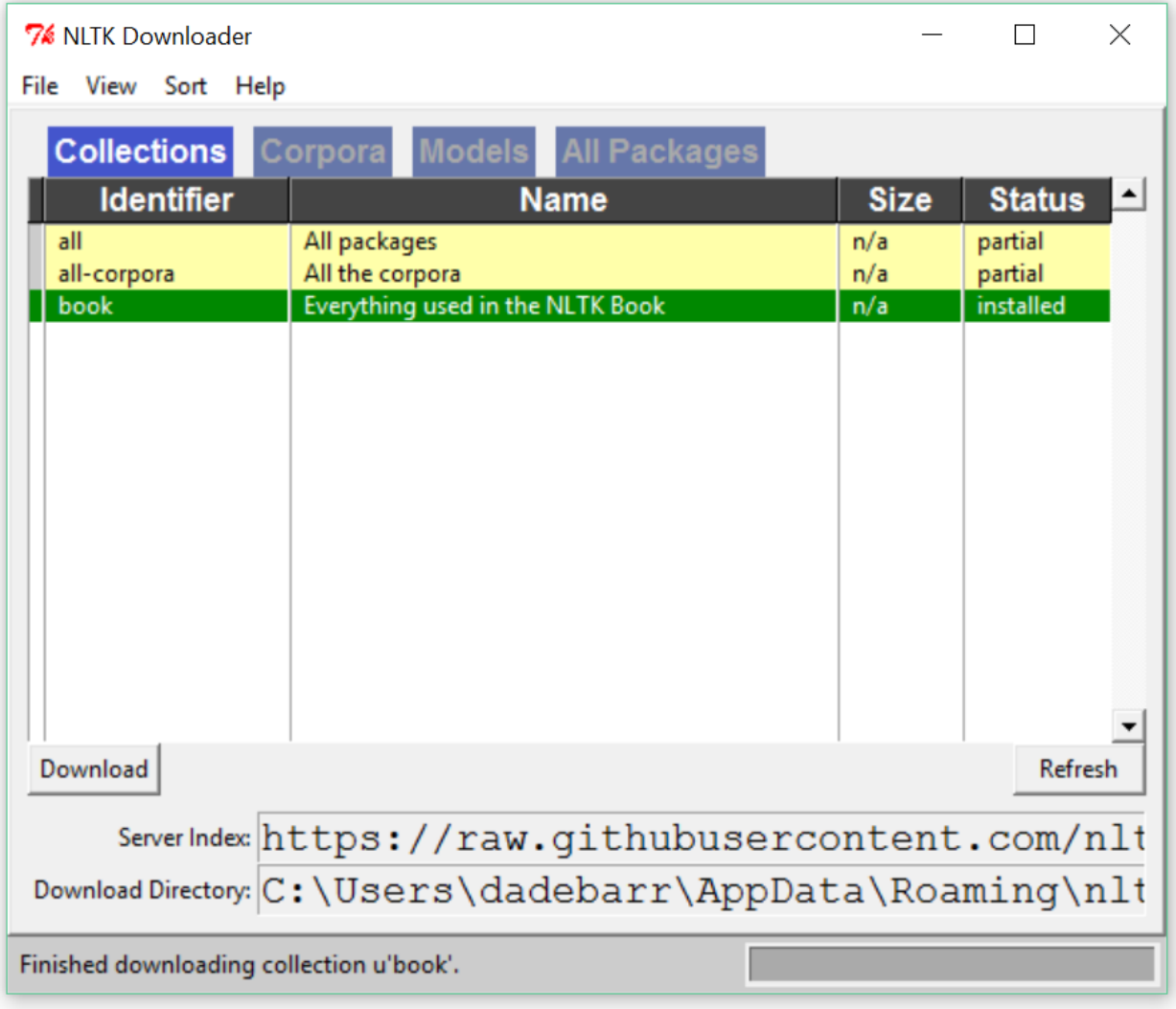
- Derive term-frequency inverse document frequency vectors

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t$$

Downloading the Natural Language ToolKit

```
import nltk
nltk.download()
```

[click “book” then “Download”]



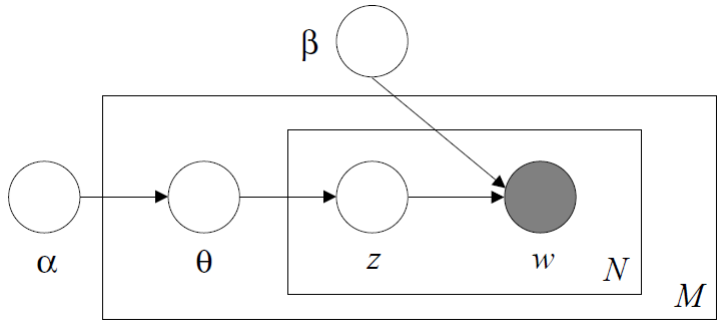
Text Classification Example

- A corpus (set) of Reuters news articles is used to produce a text classification model
- The vectors are in sparse format, with one line per article
  - Each line begins with a class label: “1” for positive class observations and “-1” for negative class observations
  - The remaining entries consist of non-zero index:TFIDF value pairs for terms found in the news article

Topic Modeling: Latent Dirichlet Allocation

LDA assumes the following generative process for each document $\mathbf{w}$ in a corpus D:

1. Choose $N \sim \text{Poisson}(\xi)$. [the number of words for a document]
2. Choose $\theta \sim \text{Dir}(\alpha)$. [the topic probabilities for a document]
3. For each of the N words w_n:
   (a) Choose a topic $z_n \sim \text{Multinomial}(\theta)$.
   (b) Choose a word w_n from $p(w_n \mid z_n, \beta)$, a multinomial probability conditioned on the topic z_n.



LDA: Probability of Observing Corpus

LDA is an unsupervised learning method.

Our goal is to learn about groups of related terms.

http://jmlr.org/papers/v3/blei03a.html

The likelihood of the corpus can be estimated using our model:

$$p(D \mid \alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn} \mid \theta_d) p(w_{dn} \mid z_{dn}, \beta) \right) d\theta_d$$

Pseudocode for LDA Training With Gibbs Sampling

Input: words $\mathbf{w} \in$ documents $\mathbf{d}$
Output: topic assignments $\mathbf{z}$ and counts $n_{d,k}, n_{k,w}$, and n_k
begin
  randomly initialize $\mathbf{z}$ and increment counters
  foreach *iteration* **do**
    for $i = 0 \rightarrow N - 1$ **do**
      $word \leftarrow w[i]$
      $topic \leftarrow z[i]$
      $n_{d,topic}$-=1; $n_{word,topic}$-=1; n_{topic}-=1
      for $k = 0 \rightarrow K - 1$ **do**
        $p(z = k|\cdot) = (n_{d,k} + \alpha_k)\frac{n_{k,w}+\beta_w}{n_k+\beta \times W}$
      end
      $topic \leftarrow$ sample from $p(z|\cdot)$
      $z[i] \leftarrow topic$
      $n_{d,topic}$+=1; $n_{word,topic}$+=1; n_{topic}+=1
    end
  end
  return $\mathbf{z}$, $n_{d,k}, n_{k,w}, n_k$
end

Installing LDA

pip install lda –user

See the python notebook for the examples covered in class