Computer Science

CSC535: Probabilistic Graphical Models

Parameter Learning

Prof. Jason Pacheco

Administrivia

- HW3 Correction: question1.m → question2.m
- See Piazza for notes on fuction-to-variable messages
- Numerically stable normalization of vector $f(x) \propto p(x)$

$$h(x) = \log f(x) - \log \max_x f(x)$$

$$p(x) = \exp(h(x)) \div \sum_x \exp(h(x))$$

Outline

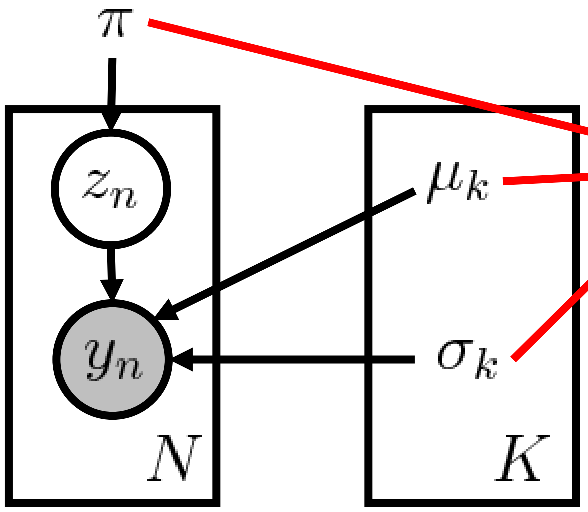
- Maximum Likelihood
- Maximum A Posteriori
- Expectation Maximization

Outline

- Maximum Likelihood
- Maximum A Posteriori
- Expectation Maximization

Example: Gaussian Mixture Model

Model is often specified in terms of *unknown parameters*



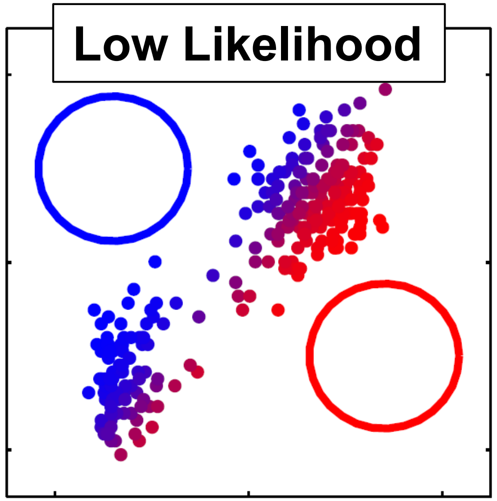
GMM

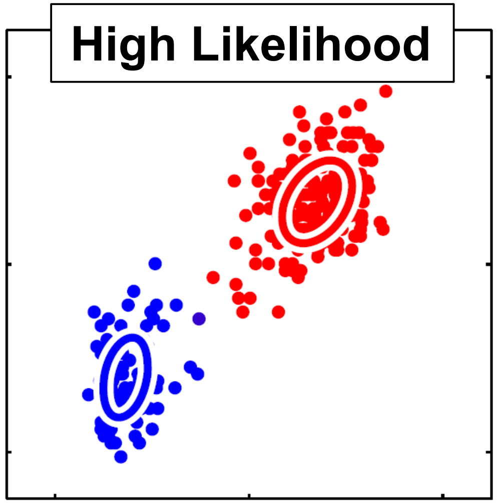
How *likely* are parameters for observed data?

$$\theta = \{\pi, \mu_1, \sigma_1, \ldots, \mu_K, \sigma_K\} \qquad \mathcal{Y} = \{y_1, \ldots, y_N\}$$

Marginal Likelihood (likelihood function):

$$p(\mathcal{Y} \mid \theta) = \sum_{z_1} \cdots \sum_{z_N} p(z_1, \ldots, z_N, \mathcal{Y} \mid \theta)$$





Intuition Learn / estimate parameters that assign highest probability (under the model) to data we've observed.

Source: Bishop, PRML

Maximum Likelihood Estimation

$$\theta^{\text{MLE}} = \arg\max_{\theta} p(\mathcal{Y} \mid \theta)$$

Consistency: Converges (in probability) to value being estimated

$$\theta^{\text{MLE}} \xrightarrow{P} \theta_0$$

True consistency *never happens* in practice since *all models are wrong* (but some are still useful)

Asymptotically Normal:

$$\sqrt{N}\left(\theta^{\text{MLE}} - \theta_0\right) \xrightarrow{D} \mathcal{N}(0, I^{-1})$$

→ **Fisher Information Matrix**

Efficiency: Achieves lowest possible variance of unbiased estimator (i.e. achieves Cramer-Rao lower bound)

Functional invariance, second-order efficiency, minimizes KL divergence, …

Maximum Likelihood Estimation

$$\theta^{\text{MLE}} = \arg\max_{\theta} p(\mathcal{Y} \mid \theta) = \arg\max_{\theta} \log p(\mathcal{Y} \mid \theta)$$

If concave then just solve for zero-gradient solution,

Log-Likelihood Function doesn't change argmax since log is monotonic

$$\mathcal{L}(\theta) \equiv \log p(\mathcal{Y} \mid \theta) \qquad \nabla_\theta \mathcal{L}(\theta^{\text{MLE}}) = 0$$

Logarithm serves a couple of practical purposes:

1) Simplifies derivatives for conditionally independent data

$$\nabla_\theta \mathcal{L}(\theta) = \sum_{i=1}^{N} \nabla_\theta \log p(y_i \mid \theta)$$

2) Avoids numerical under/overflow

MLE of Gaussian Mean

Assume data are i.i.d. univariate Gaussian,

Variance is known

$$p(\mathcal{Y} \mid \theta) = \prod_{i=1}^{N} \mathcal{N}(y_i \mid \theta, \sigma^2)$$

Log-likelihood function:

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} \log \left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}(y_i - \theta)^2 \sigma^{-2} \right) \right)$$

Constant doesn't depend on mean

$$= \text{const.} - \frac{1}{2} \sum_{i=1}^{N} \left((y_i - \theta)^2 \sigma^{-2} \right)$$

MLE doesn't change when we:
1) Drop constant terms (in θ)
2) Minimize negative log-likelihood

MLE estimate is *least squares estimator*:

$$\theta^{\text{MLE}} = -\frac{1}{2\sigma^2} \arg\max_{\theta} \sum_{i=1}^{N} (y_i - \theta)^2 = \arg\min_{\theta} \sum_{i=1}^{N} (y_i - \theta)^2$$

MLE of Gaussian Mean

Sum of squares objective is convex,

$$\theta^{\text{MLE}} = \arg\min_{\theta} \sum_{i=1}^{N} (y_i - \theta)^2$$

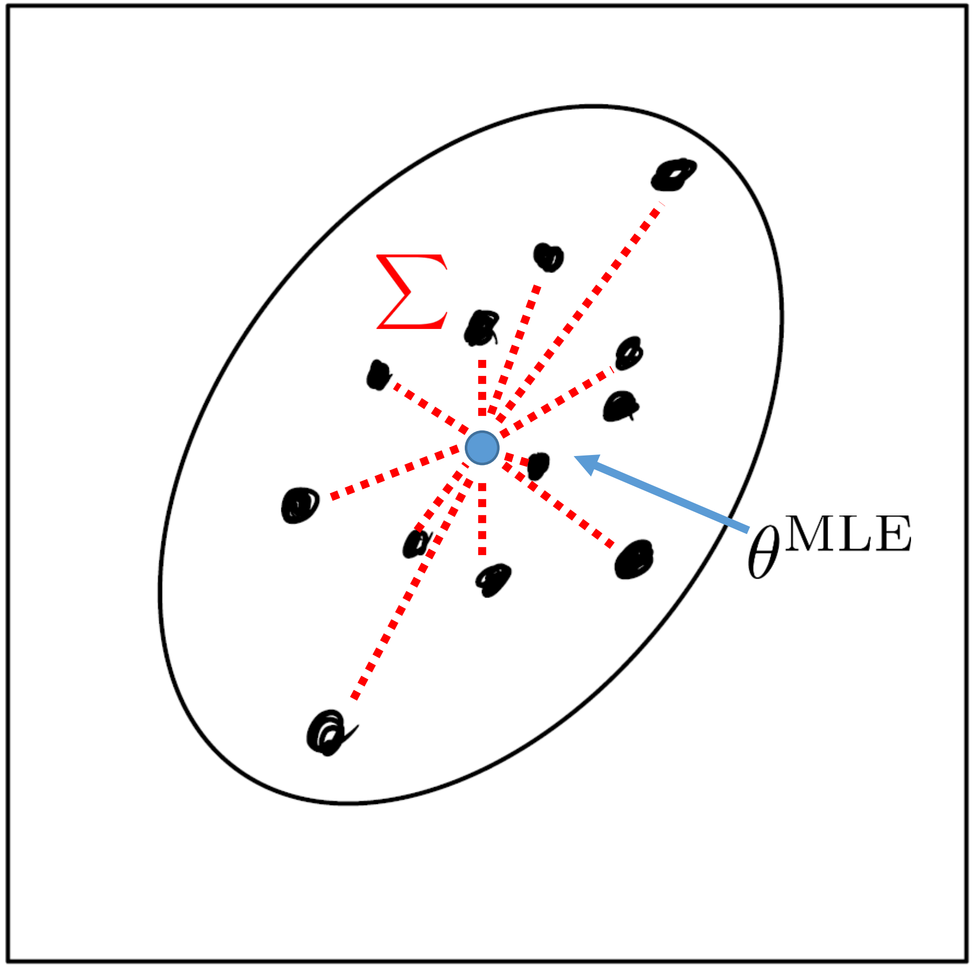
Set derivative to zero and solve,

$$\sum_{i=1}^{N} \frac{d}{d\theta}(y_i - \theta)^2 = -2\sum_{i=1}^{N}(y_i - \theta) = 0$$

$$\left(\sum_{i=1}^{N} y_i\right) - N\theta = 0$$

MLE is empirical mean of data,

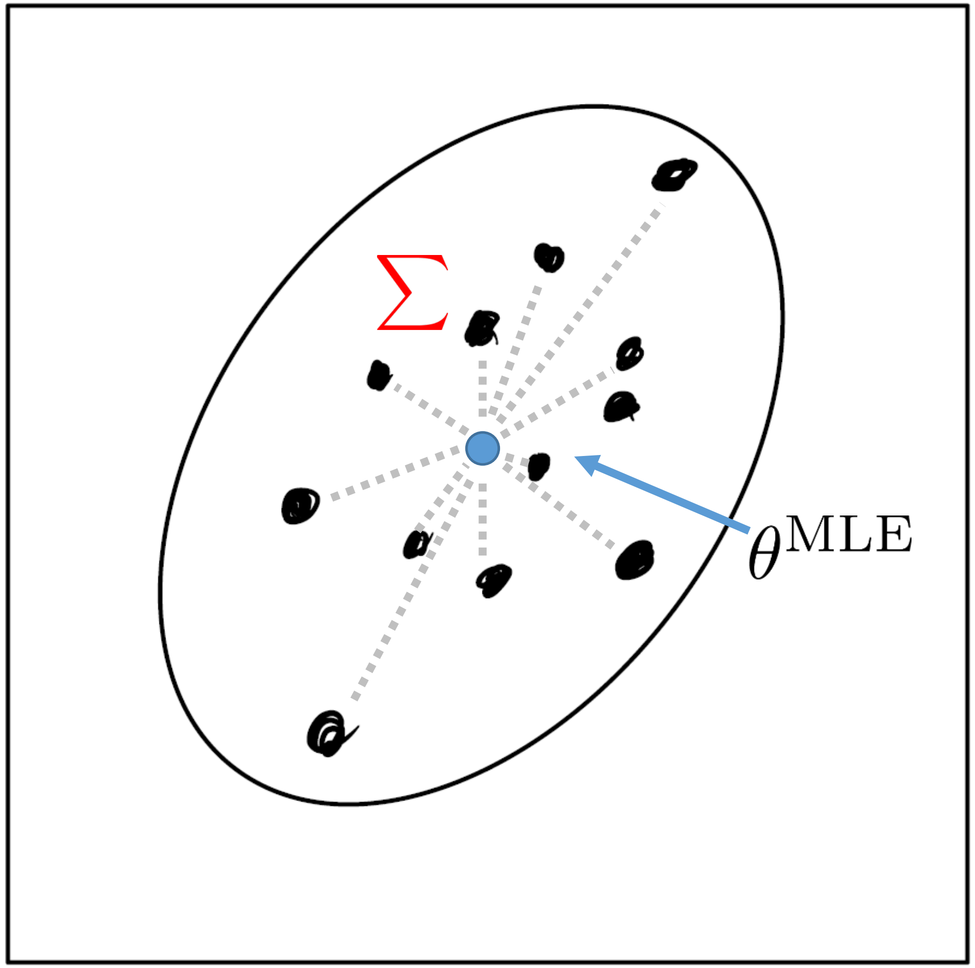
$$\theta^{\text{MLE}} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

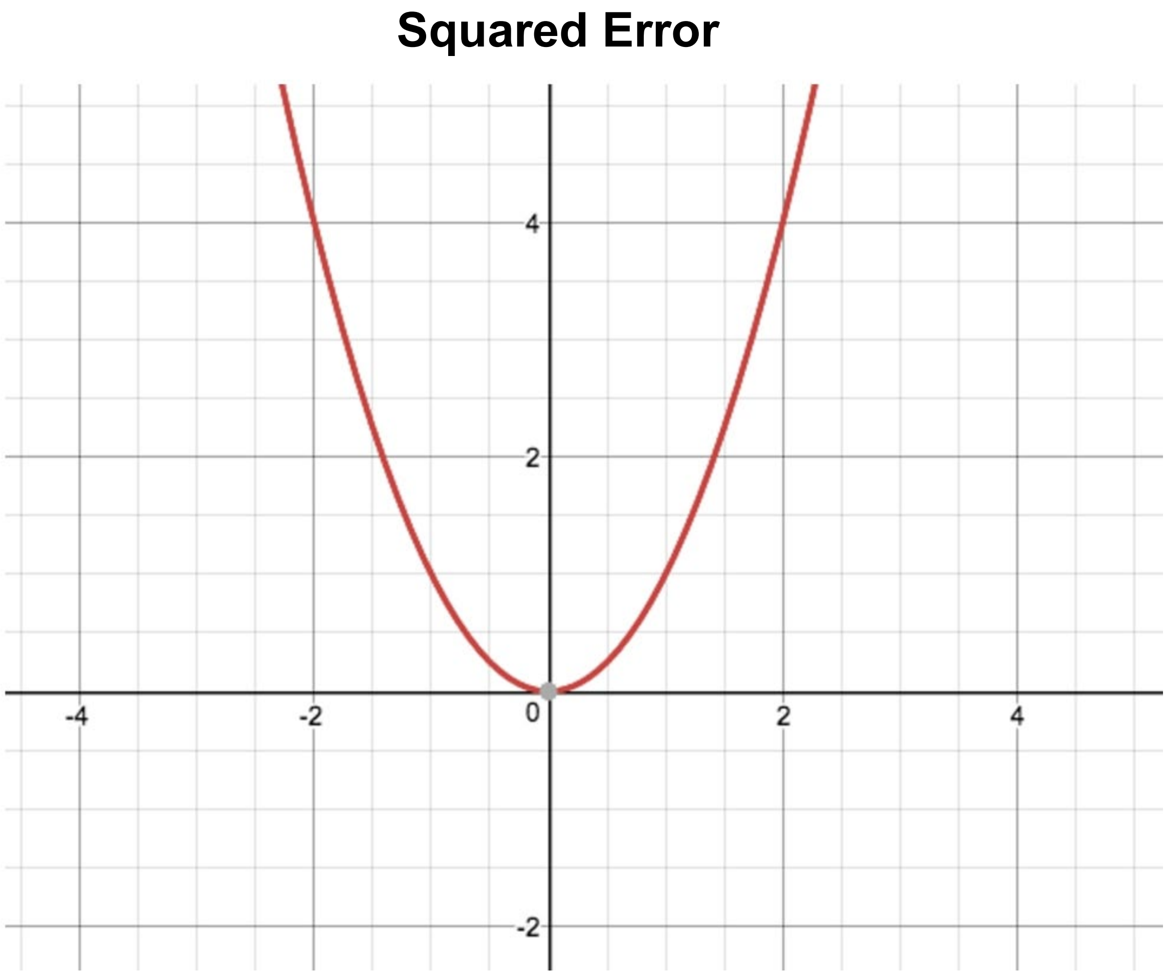


Minimize squared distance from mean

Outliers

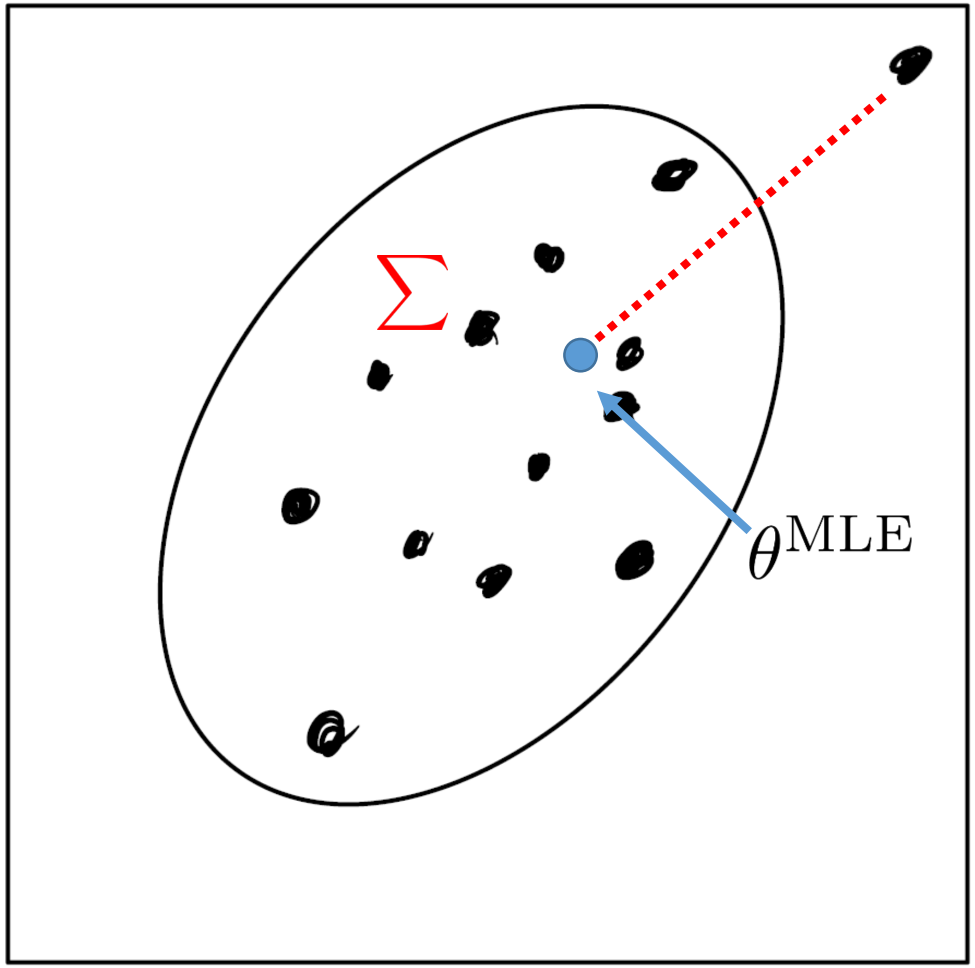
How does an outlier affect the estimator?

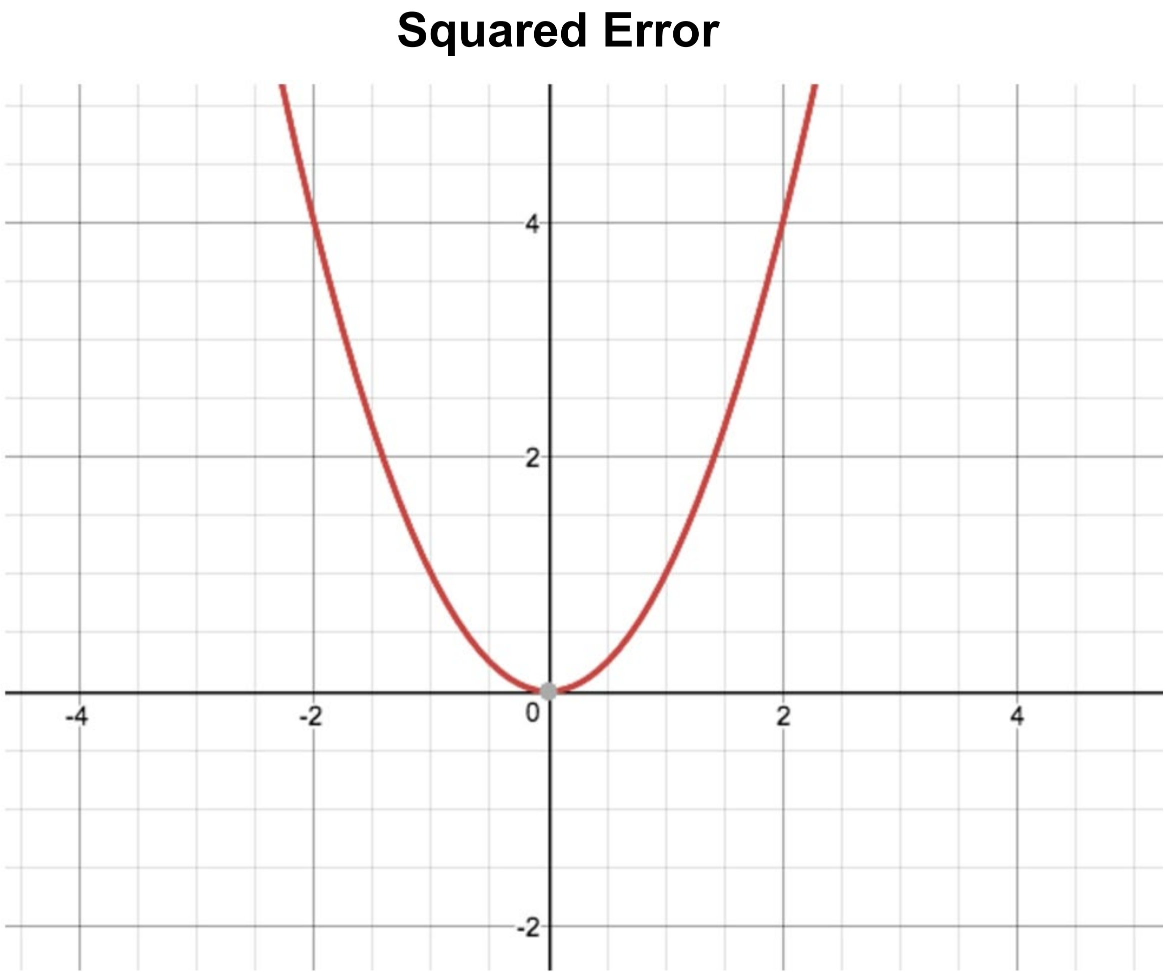




Outliers

How does an outlier affect the estimator?





Regularized Maximum Likelihood

Penalty term R minimizes effect of outliers on estimator,

$$\theta^{\text{MLE}} = \arg\max_{\theta} \mathcal{L}(\theta) - \lambda R(\theta)$$

Regularization weight ← λ | $R(\theta)$ → Regularizer

Example L2-regularized Least-Squares,

$$\theta^{\text{MLE}} = \arg\min_{\theta} \sum_{i=1}^{N} (y_i - \theta)^2 + \frac{\lambda}{2}\theta^2$$

In regression setting these have various names: ridge regression, LASSO

Example L1-regularized Least-Squares,

L1 is not differentiable, and so care must be taken in optimizer

$$\theta^{\text{MLE}} = \arg\min_{\theta} \sum_{i=1}^{N} (y_i - \theta)^2 + \lambda|\theta|$$

Regularized Maximum Likelihood

Penalty term R minimizes effect of outliers on estimator,

$$\hat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta) - \lambda R(\theta)$$

Regularization weight (λ) — Regularizer ($R(\theta)$)

Example L2-regularized Least-Squares,

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} (y_i - \theta)^2 + \frac{\lambda}{2}\theta^2$$

In regression setting known as ridge regression

$$\frac{1}{2} \sum_{i=1}^{N} \frac{d}{d\theta} (y_i - \theta)^2 + \frac{d}{d\theta} \frac{\lambda}{2}\theta^2 \quad = -\left(\sum_{i=1}^{N} y_i \right) + N\theta + \lambda\theta = 0$$

$$\hat{\theta} = \frac{1}{N + \lambda} \sum_{i} y_i$$

λ acts as *pseudocount*

Linear Regression - Ordinary Least Squares (OLS)

Linear function of inputs X,

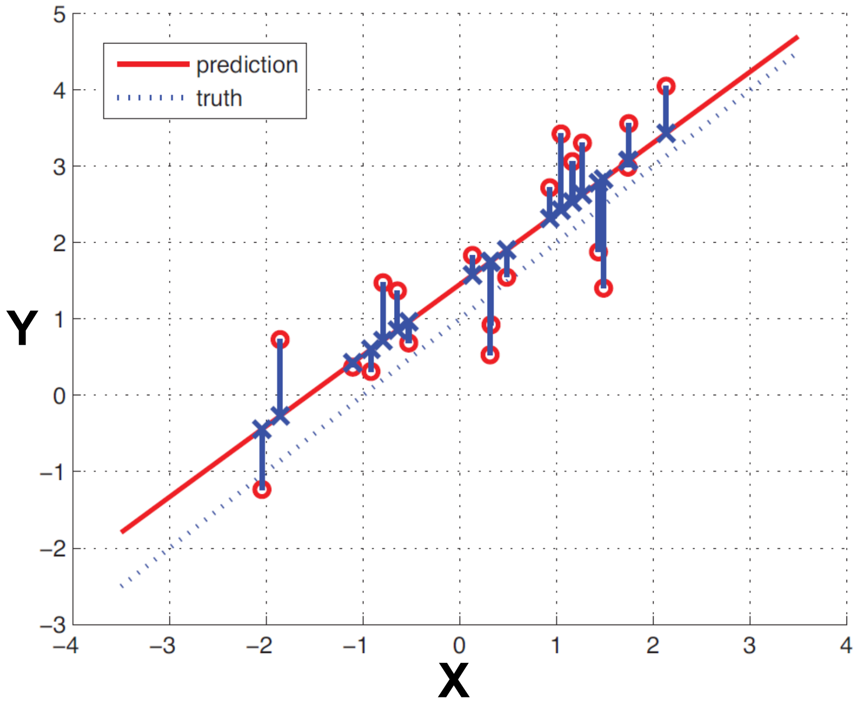
$$y = \theta_0 + \theta_1 x_1 + \ldots + \theta_d x_d + \epsilon$$

With $\epsilon \sim \mathcal{N}(0, \sigma^2)$ and MLE,

Shorthand:
$x^i = (1, x_1^i, \ldots, x_d^i)^T$

$$\theta^{\text{MLE}} = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} (y^i - \theta^T x^i)^2$$

Source: Kevin Murphy's Textbook



Solving for zero-gradient:

$$0 = \frac{1}{2} \sum_{i=1}^{N} \nabla_\theta (y^i - \theta^T x^i)^2$$

$$0 = \sum_{i=1}^{N} (y^i - \theta^T x^i)(x^i)^T = \sum_{i=1}^{N} y^i (x^i)^T - \theta^T \sum_{i=1}^{N} x^i (x^i)^T$$

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_1 & \ldots & x_1 \\ \vdots & \vdots & \vdots & \vdots \\ x_d & x_d & \ldots & x_d \end{pmatrix}$$

$$\theta^{\text{MLE}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

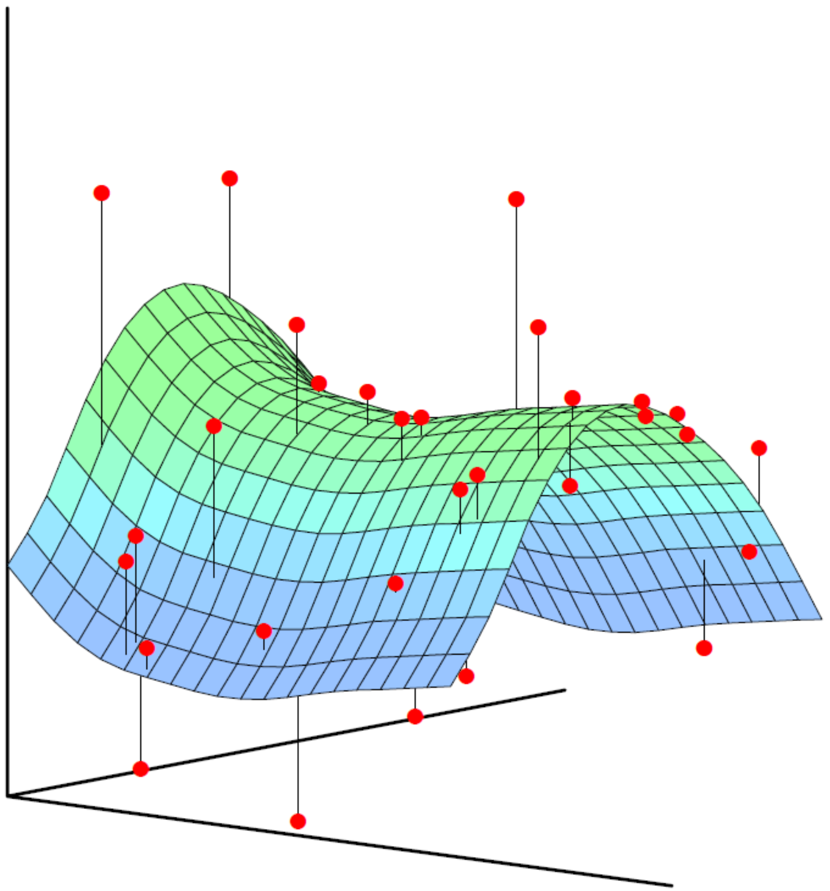
where

$$\mathbf{y} = (y^1, \ldots, y^N)^T$$

Linear Regression – Basis Functions

Predicted functions may be nonlinear in X

Source: Elements of Stat. Learning



Define a set of *basis functions* or *features*:

$$f_\theta(x) = \sum_{k=1}^{K} h_k(x)\theta_k,$$

Output is linear Gaussian (in basis func's):

$$p(y \mid \theta, h(x)) = \mathcal{N}(f_\theta(x), \sigma^2)$$

Least squares solution takes same form:

$$\theta^{\text{MLE}} = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{y}$$

F is a matrix of feature evaluations at each input in training set

L2 Regularized Linear Regression – Ridge Regression

Source: Kevin Murphy's Textbook

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} (y^i - \theta^T x^i)^2 + \frac{\lambda}{2} \theta^T \theta$$

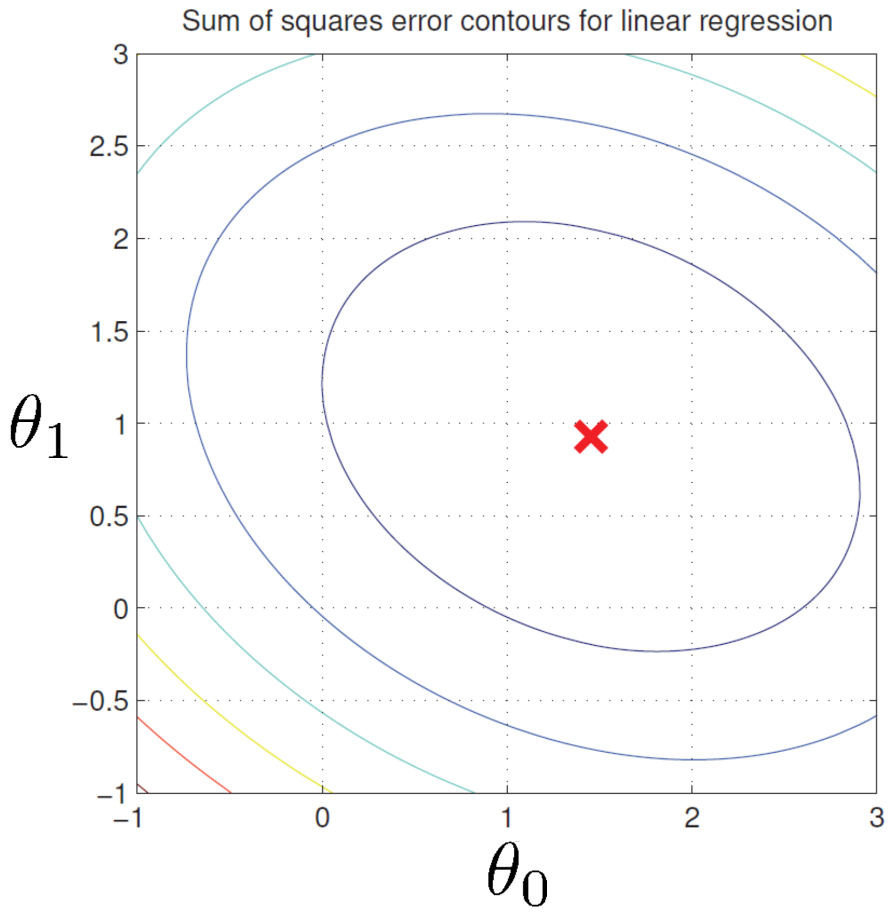
After some algebra…

$$\hat{\theta} = (\lambda I + \mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

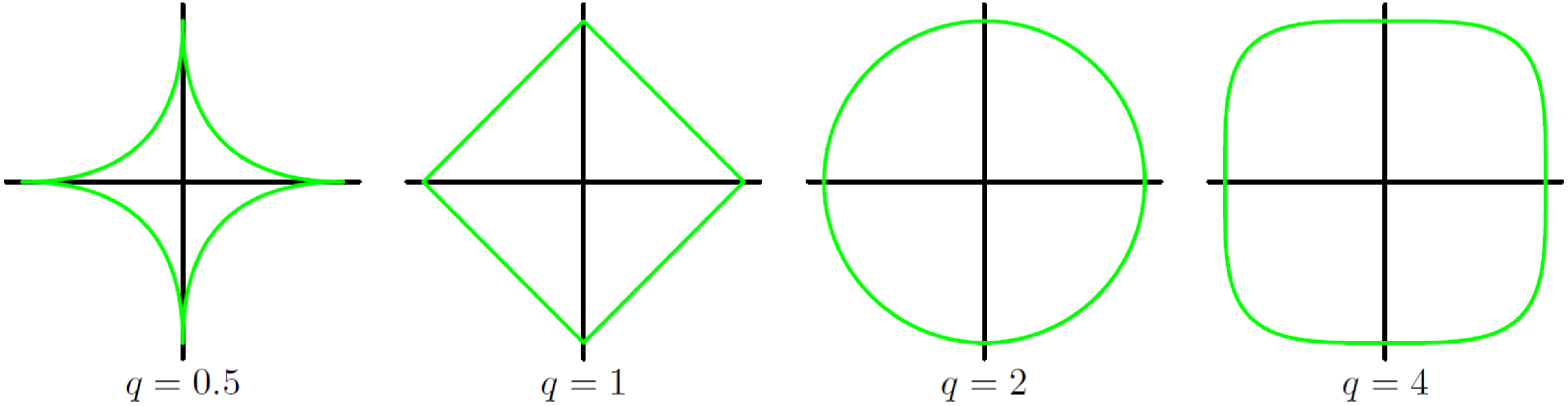
Compare to unregularized solution:

$$\theta^{\mathrm{MLE}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Regularized least-squares includes pseudocount in weighting similar to Gaussian mean estimator



Other Regularization Terms



q<1 is not a norm, and thus not convex

L1 is non-differentiable

L2 Regularization

A more general regularization penalty,

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} (y_i - \theta)^2 + \frac{\lambda}{2} |\theta|^q$$

MLE More Generally

MLE has a closed-form in Gaussian models because they are convex:

$$\theta^{\mathrm{MLE}} = \arg\max_{\theta} \underbrace{\log p(\mathcal{Y} \mid \theta) \equiv \mathcal{L}(\theta)}_{\text{Quadratic in Gaussian MLE}}$$

Log-likelihood is typically non-convex, so we use numerical methods such as Gradient descent:

$$\theta^{k+1} = \theta^k + \beta \nabla_\theta \mathcal{L}(\theta^k)$$

In this setting we cannot generally guarantee optimal MLE estimators

Administrivia

- HW2 grades by end of week
- Midterm: Monday 10/26 (take-home)
- Clarification of parallel sum-product for factor graphs

MLE Summary

- Recall the trick of maximizing the p.d.f. by minimizing the negative log
- The Gaussian form for the likelihood led to a least-squares problem
- Least-squares solutions are tightly connected to assuming Gaussian distribution for the random effects (noise)
- If the random part is not Gaussian, then squared error may not make sense
- Squared error and Gaussian assumptions are mathematically very convenient but they are **very sensitive to outliers** (this motivates *robust estimators*)
- The least-squares solution leads to the average as being the “best” way to characterize a group of independent numbers, but there are other answers:
  - Minimum absolute value for error
  - Median
  - Minimum risk / maximal gain

Outline

- Maximum Likelihood
- Maximum A Posteriori
- Expectation Maximization

Maximum A Posteriori (MAP) Estimation

Recall the MAP estimator maximizes posterior probability,

$$\theta^{\mathrm{MAP}} = \arg\max_\theta p(\theta \mid \mathcal{Y})$$

$$= \arg\max_\theta p(\theta, \mathcal{Y})$$ **(Bayes' rule)**

$$= \arg\max_\theta p(\mathcal{Y} \mid \theta) p(\theta)$$ **(Probability Chain Rule)**

$$= \arg\max_\theta \log p(\mathcal{Y} \mid \theta) + \log p(\theta)$$ **(Monotonicity of Logarithm)**

Prior serves as regularizer in regularized MLE:

$$\theta^{\mathrm{MLE}} = \arg\max_\theta \mathcal{L}(\theta) - \lambda R(\theta)$$

So conceptually, defining a regularizer in MLE imposes prior beliefs

MAP of Gaussian Mean

Gaussian prior on θ with i.i.d. Gaussian observations:

Variance is known

$$p(\mathcal{Y}, \theta) = \mathcal{N}(\theta \mid 0, \lambda^{-1}) \prod_{i=1}^{N} \mathcal{N}(y_i \mid \theta, \sigma^2)$$

Log-joint probability:

$$J(\theta) = \log\left(\sqrt{\frac{\lambda}{2\pi}} \exp\left(-\frac{1}{2}\theta^2\lambda\right)\right) + \sum_{i=1}^{N} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}(y_i - \theta)^2\sigma^{-2}\right)\right)$$

$$= \text{const.} - \frac{\lambda}{2}\theta^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \theta)^2$$

Minimize negative log-joint (+ rearrange terms):

MAP estimate equivalent to regularized least squares estimator

Note Likelihood variance can be incorporated into prior variance

$$\theta^{\text{MAP}} = \arg\min_{\theta} \frac{1}{2}\sum_{i=1}^{N}(y_i - \theta)^2 + \frac{\lambda}{2}\theta^2$$

Bayesian Linear Regression

Gaussian prior on regression weights,

$$p(\theta) = \mathcal{N}(\theta \mid m_0, S_0) \qquad p(y \mid \theta, x) = \mathcal{N}(y \mid \theta^T x, \sigma^2)$$

Posterior over N observations is Gaussian (yay for Gaussians!),

$$p(\theta \mid \mathcal{Y}, \mathcal{X}) = \mathcal{N}(\theta \mid m_N, S_N)$$

$$m_N = S_N \left(S_0^{-1} m_0 + \sigma^{-2} \mathbf{X}^T \mathbf{y} \right) \qquad S_N^{-1} = S_0^{-1} + \sigma^{-2} \mathbf{X}^T \mathbf{X}$$

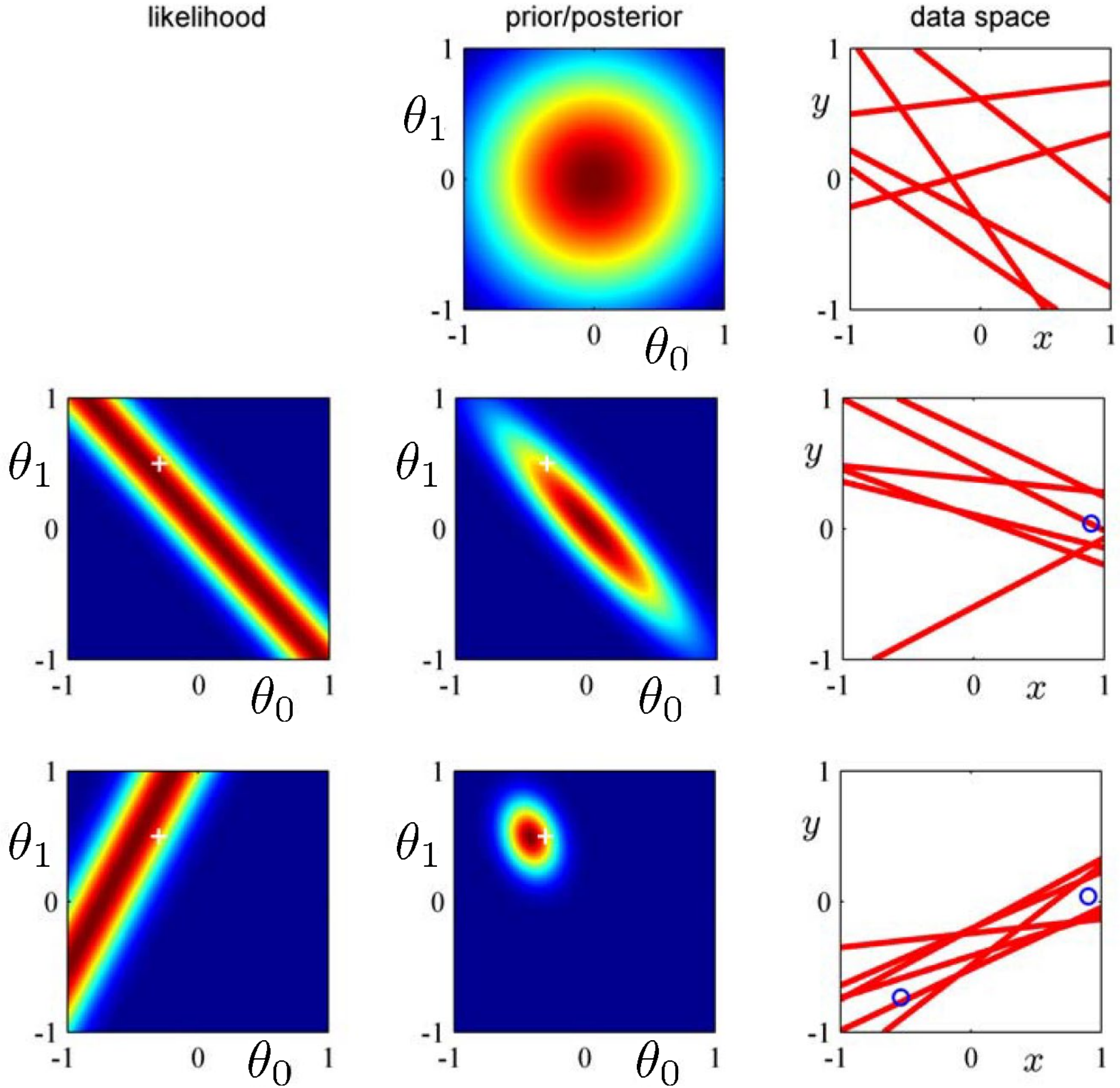
MAP is posterior mean,

$$\theta^{\text{MAP}} = m_N$$

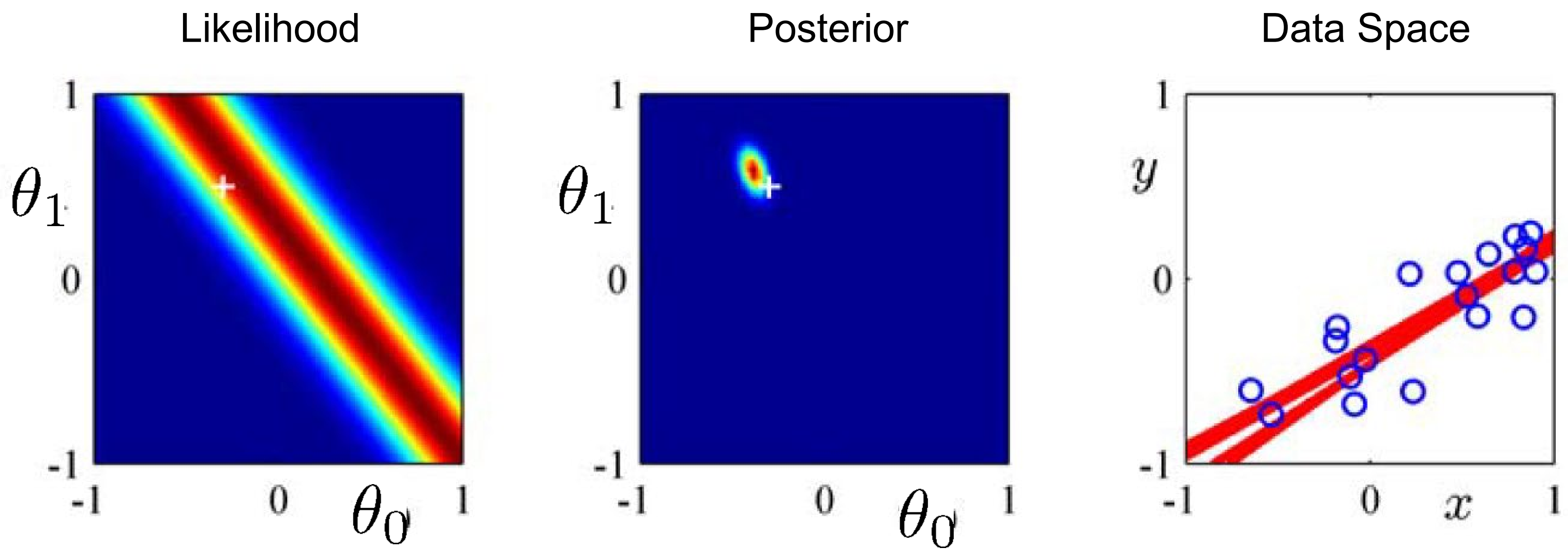
Again equivalent to regularized least squares (ridge regression)

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_1 & \dots & x_1 \\ \vdots & \vdots & \vdots & \vdots \\ x_d & x_d & \dots & x_d \end{pmatrix}$$

$$\mathbf{y} = (y^1, \dots, y^N)^T$$



Source: Chris Bishop, PRML



Posterior concentrates on true weights as more data observed

Likelihood outweighs prior in the limit (converges to MLE)

Source: Chris Bishop, PRML

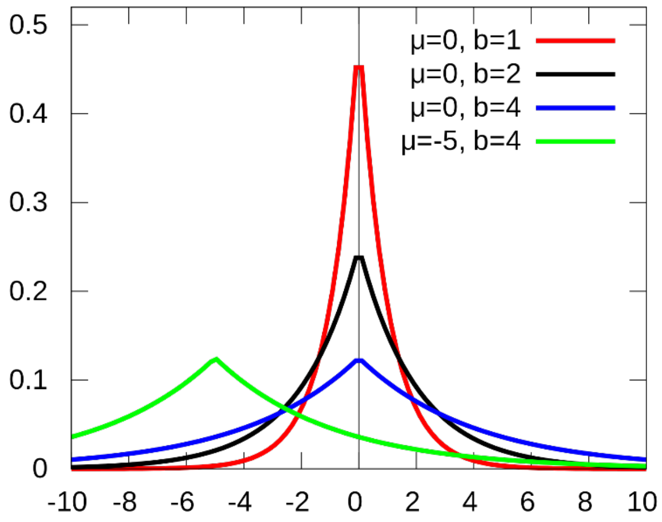
Sparse Prior on Regression Weights

Laplace distribution

$$\text{Laplace}(\theta \mid \mu, b) = \frac{1}{2b} \exp\left(-\frac{|\theta - \mu|}{b}\right)$$

Mean μ and scale $b > 0$.

Compared to Gaussian: Higher probability at zero, larger tails



Regression Joint Probability

$$p(\theta, \mathcal{Y} \mid \mathcal{X}) = \text{Laplace}(\theta \mid 0, \lambda^{-1}) \prod_{i=1}^{N} \mathcal{N}(y^i, \theta^T x^i, \sigma^2)$$

Does not have closed-form.
Convex, but non-differentiable.
Solve via iterative methods.

MAP Estimate

$$\theta^{\text{MAP}} = \arg\min_{\theta} - \log p(\theta, \mathcal{Y} \mid \mathcal{X}) = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} (y^i - \theta^T x^i)^2 - \lambda|\theta|$$

Equivalent to L1-regularized least squares MLE (LASSO)

Summary

Bayesian approach allows for different perspective of MLE

- MAP = MLE for particular regularizer/prior
- MLE Regularizer implicitly imposes prior belief
- MAP estimate can be sequentially updated with additional data
- Inference = optimization (can avoid calculus in Gaussian case)

Administrivia

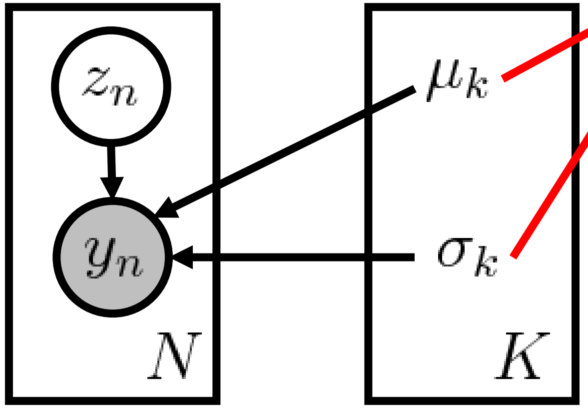
- HW3 due later today
- HW2 graded and solutions posted
- Review readings this week (no assignments)
- “Take-home” midterm Monday
  - Everything up-to-and-including parameter learning material
- We will have a midterm review lecture Monday

Outline

- Maximum Likelihood
- Maximum A Posteriori
- Expectation Maximization

Marginal Likelihood Calculation

Recall the Gaussian Mixture Model…

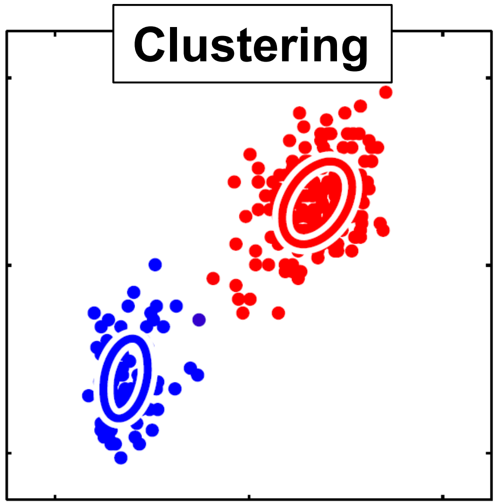


$$\theta = \{\mu_1, \sigma_1, \dots, \mu_K, \sigma_K\}$$

Marginal Likelihood (likelihood function):

$$p(\mathcal{Y} \mid \theta) = \underbrace{\sum_{z_1} \cdots \sum_{z_N}} p(z_1, \dots, z_N, \mathcal{Y} \mid \theta)$$

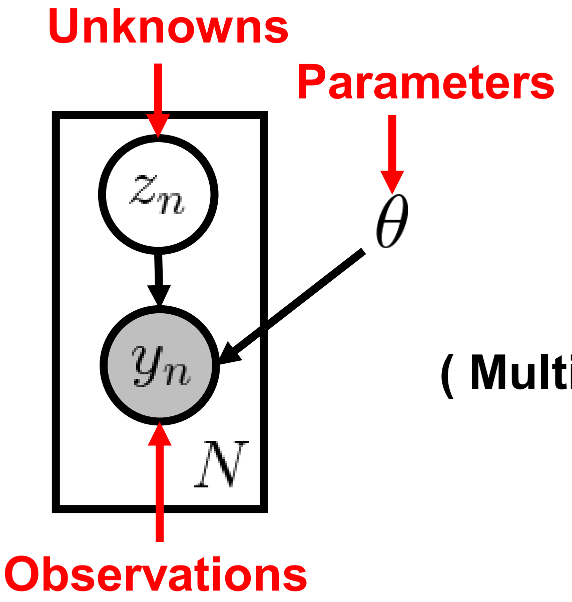
Sum over all possible K^N assignments,
which we cannot compute



Motivation Approximate MLE / MAP when we cannot compute the marginal likelihood in closed-form

Lower Bounding Marginal Likelihood

Conditionally-independent model with partial observations…



$$\log p(\mathcal{Y} \mid \theta) = \log \sum_{z_1} \dots \sum_{z_N} p(z_1, \dots, z_N, \mathcal{Y} \mid \theta)$$

(Multiply by q(z)/q(z)=1) $$= \log \sum_{z} p(z, \mathcal{Y} \mid \theta) \left(\frac{q(z)}{q(z)} \right)$$

Shorthand
$z = z_1, \dots, z_N$

(Definition of Expected Value) $$= \log \mathbf{E}_q \left[\frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right]$$

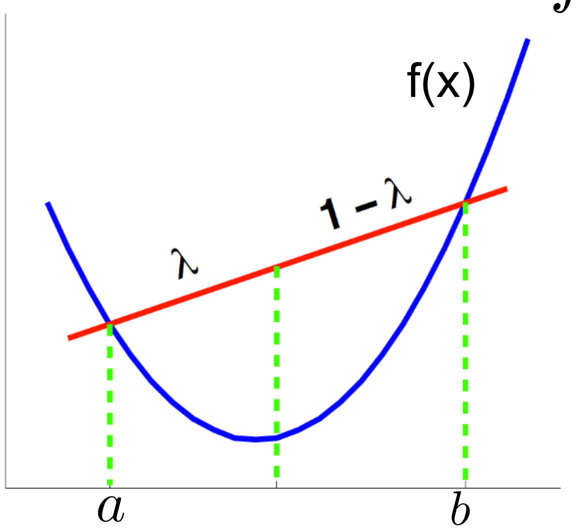
q(z) is *any* distribution with support over Z

(Jensen's Inequality) $$\geq \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right]$$

Jensen's Inequality

Definition A function f(x) is convex iff for any points a,b and $0 \leq \lambda \leq 1$

$$f(\lambda a + (1-\lambda)b) \leq \lambda f(a) + (1-\lambda) f(b)$$



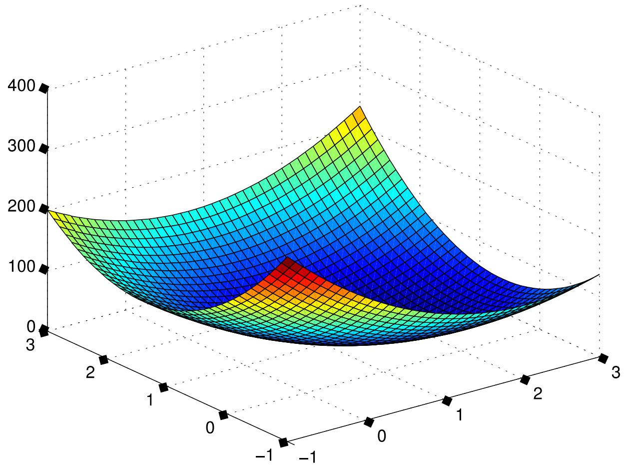
Jensen's Inequality holds for any convex f(x),

$$f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$$

Proof (sketch) is by induction on m points,

$$f\left(\sum_{i=1}^{m} \lambda_i x_i\right) \leq \sum_{i=1}^{m} \lambda_i f(x_i)$$

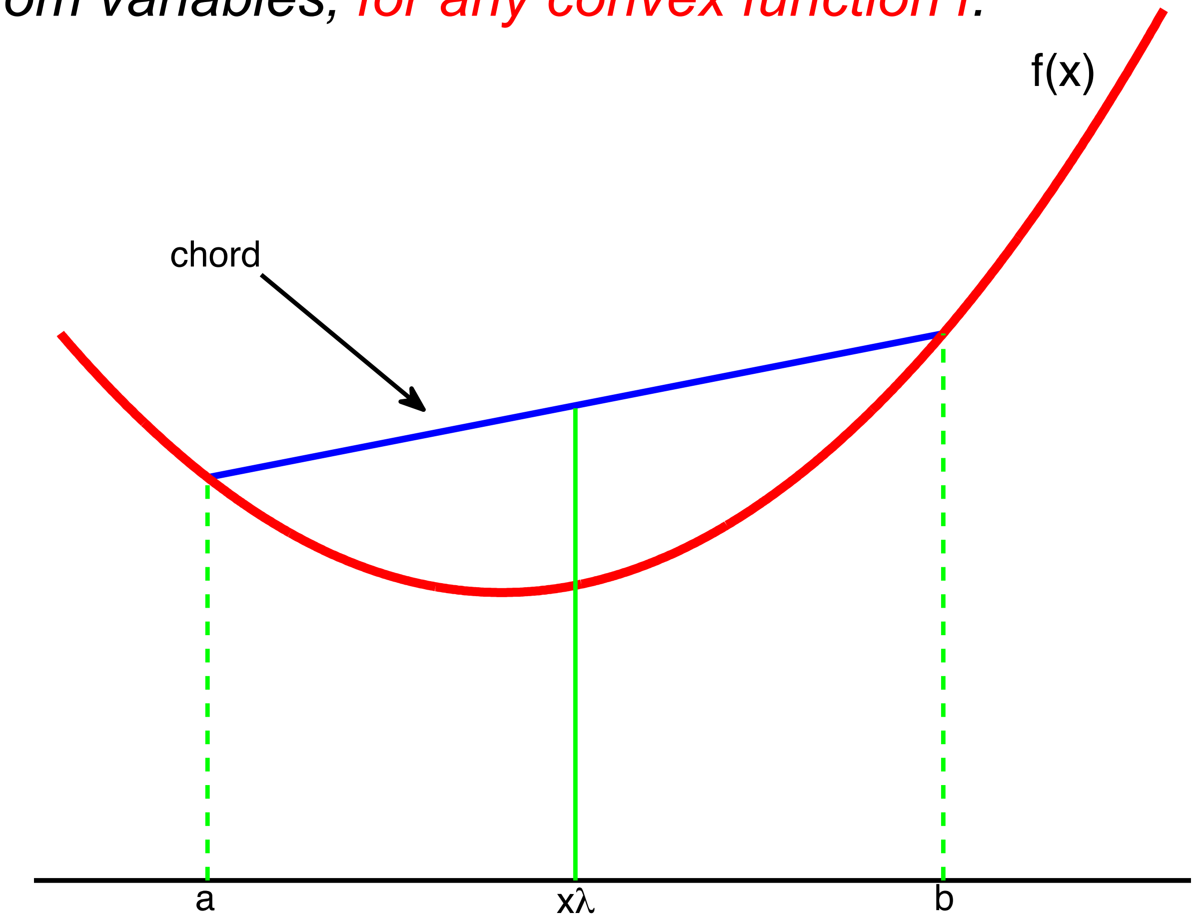
where $\lambda_i \geq 0, \sum_{i=1}^{m} \lambda_i = 1$ so $\lambda_i = \Pr[X = x_i]$



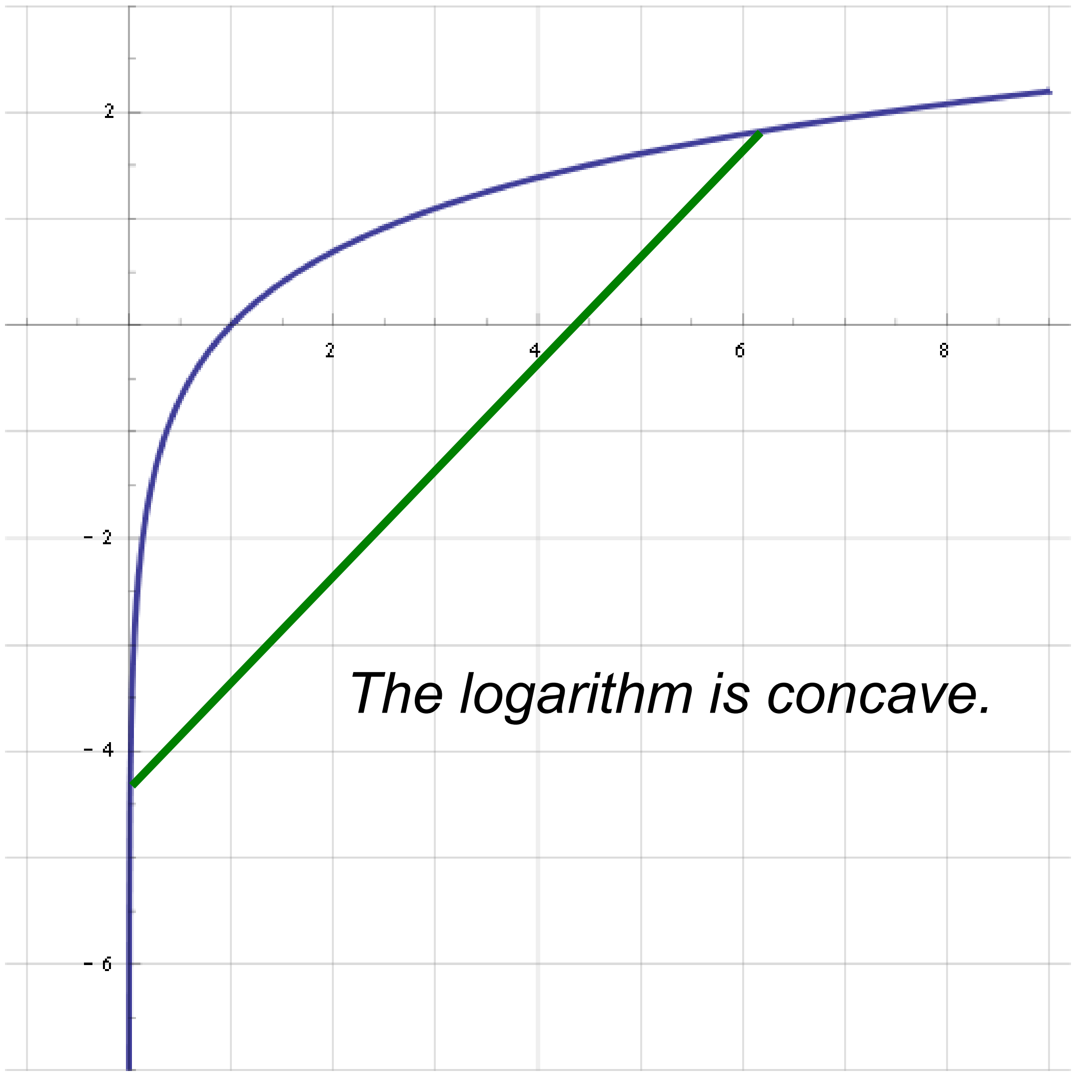
Jensen's Inequality

$$f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$$

Valid for both discrete (expectations are sums) and continuous (expectations are integrals) random variables, for any convex function f.



$$\ln(\mathbb{E}[X]) \geq \mathbb{E}[\ln(X)]$$



The logarithm is concave.

Expectation Maximization

Find tightest lower bound of marginal likelihood,

$$\max_{\theta} \log p(\mathcal{Y} \mid \theta) \geq \max_{q,\theta} \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] \equiv \mathcal{L}(q, \theta)$$

Solve by coordinate ascent…

Initialize Parameters: $\theta^{(0)}$

At iteration t do:

Update q: $q^{(t)} = \arg\max_q \mathcal{L}(q, \theta^{(t-1)})$ — **Fix θ**

Update θ: $\theta^{(t)} = \arg\max_\theta \mathcal{L}(q^{(t)}, \theta)$ — **Fix q**

Until convergence

Expectation Maximization

Find tightest lower bound of marginal likelihood,

$$\max_{\theta} \log p(\mathcal{Y} \mid \theta) \geq \max_{q,\theta} \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] \equiv \mathcal{L}(q, \theta)$$

Solve by coordinate ascent…

Initialize Parameters: $\theta^{(0)}$

At iteration t do:

E-Step: $q^{(t)} = \arg\max_q \mathcal{L}(q, \theta^{(t-1)})$ (Fix θ)

M-Step: $\theta^{(t)} = \arg\max_\theta \mathcal{L}(q^{(t)}, \theta)$ (Fix q)

Until convergence

E-Step

$$q^{(t)}(z) = \arg\max_q \mathcal{L}(q, \theta^{(t-1)}) \equiv \mathbf{E}_q \left[\log \frac{p(z, y \mid \theta^{(t-1)})}{q(z)} \right]$$

Concave (in $q(z)$) and optimum occurs at,

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

Set q(z) to posterior with current parameters

Initialize Parameters: $\theta^{(0)}$

At iteration t do:

E-Step: $q^{(t)}(z) = p(z \mid y, \theta^{(t-1)})$

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

Until convergence

M-Step

$$\theta^{(t)} = \arg\max_{\theta} \mathcal{L}(q^{(t)}, \theta) = \arg\max_{\theta} \mathbf{E}_{q^{(t)}} \left[\log \frac{p(z, y \mid \theta)}{q^{(t)}} \right]$$

Adding / subtracting constants we have,

$$\theta^{(t)} = \arg\max_{\theta} \sum_{z} q^{(t)}(z) \log p(y \mid z, \theta)$$

Intuition We don't know Z, so average log-likelihood over current posterior q(z), then maximize. E.g. weighted MLE.

May lack a closed-form, but suffices to take one or more gradient steps. Don't need to maximize, just improve.

Expectation Maximization

Initialize Parameters: $\theta^{(0)}$

At iteration t do:

E-Step: $q^{(t)}(z) = p(z \mid y, \theta^{(t-1)})$

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

Until convergence

Expectation in E-step is kind of confusing. Think of this as alternating maximizations

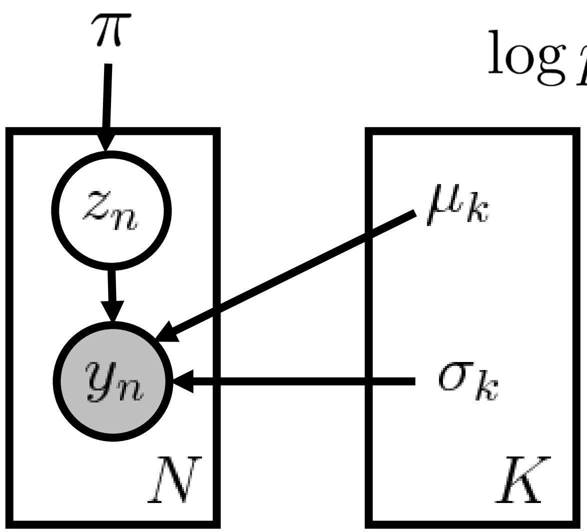
E-Step Compute **expected** log-likelihood under the posterior distribution,

$$q^{(t)}(z) = p(z \mid y, \theta^{(t-1)}) \qquad \mathbf{E}_{q^{(t)}}[\log p(y \mid z, \theta)] = \mathcal{L}(q^{(t)}, \theta)$$

M-Step Maximize expected log-likelihood,

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

Example: Gaussian Mixture Model



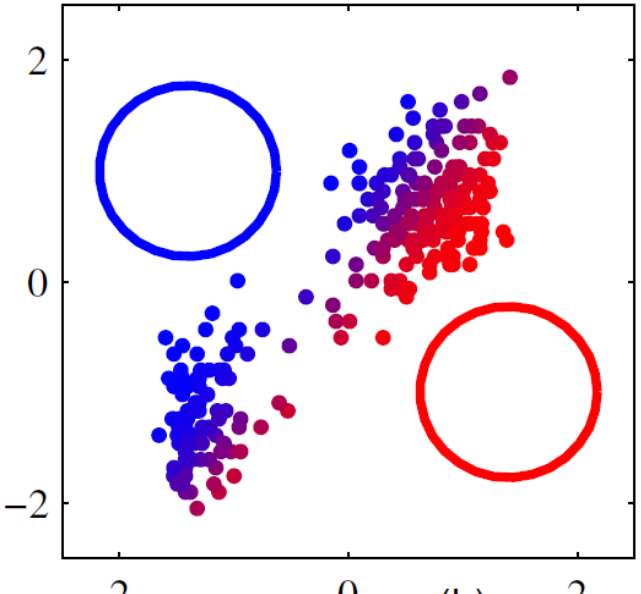
$$\log p(\mathcal{Y} \mid \pi, \mu, \Sigma) \geq \sum_{n=1}^{N} \sum_{k=1}^{K} q(z_n = k) \log \{\pi_k \mathcal{N}(y_n \mid \mu_k, \Sigma_k)\} = \mathcal{L}(q, \theta)$$

E-Step: $q^{\text{new}} = \arg\max_q \mathcal{L}(q, \theta^{\text{old}})$

$$q^{\text{new}}(z_n = k) = p(z_n = k \mid \mathcal{Y}, \mu^{\text{old}}, \Sigma^{\text{old}}, \pi^{\text{old}})$$

$$= \frac{p(z_n = k, \mathcal{Y} \mid \mu^{\text{old}}, \Sigma^{\text{old}}, \pi^{\text{old}})}{\sum_{j=1}^{K} p(z_n = j, \mathcal{Y} \mid \mu^{\text{old}}, \Sigma^{\text{old}}, \pi^{\text{old}})}$$

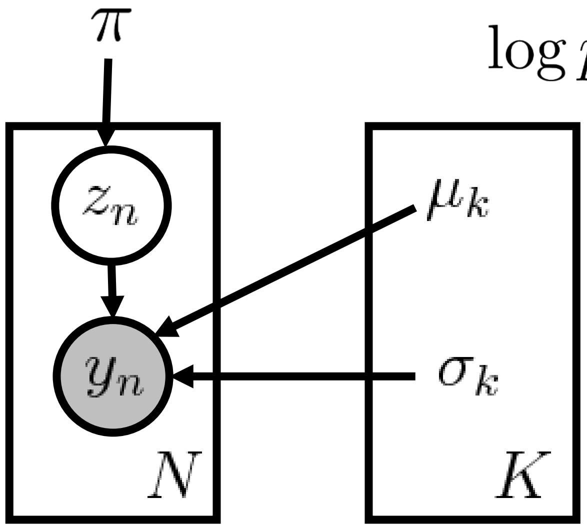
$$= \frac{\pi_k \mathcal{N}(y_n \mid \mu_k^{\text{old}}, \Sigma_k^{\text{old}})}{\sum_{j=1}^{K} \pi_j \mathcal{N}(y_n \mid \mu_j^{\text{old}}, \Sigma_j^{\text{old}})}$$



(b)

Commonly refer to $q(z_n)$ as *responsibility*

Example: Gaussian Mixture Model



$$\log p(\mathcal{Y} \mid \pi, \mu, \Sigma) \geq \sum_{n=1}^{N} \sum_{k=1}^{K} q(z_n = k) \log \{\pi_k \mathcal{N}(y_n \mid \mu_k, \Sigma_k)\} = \mathcal{L}(q, \theta)$$

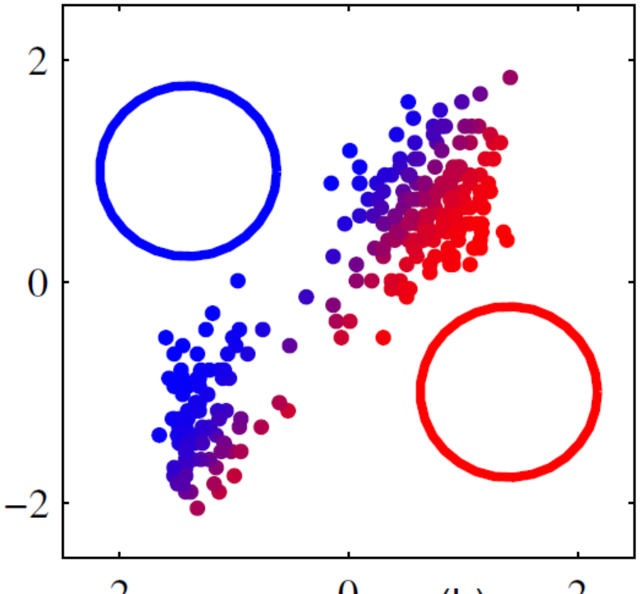
M-Step: $\theta^{\text{new}} = \arg\max_{\theta} \mathcal{L}(q^{\text{new}}, \theta)$

Start with mean parameter μ_k,

$$0 = \nabla_{\mu_k} \mathcal{L}(q^{\text{new}}, \theta)$$

$$0 = \sum_{n=1}^{N} \nabla_{\mu_k} \mathbf{E}_{z_n \sim q^{\text{new}}} [\log \mathcal{N}(y_n \mid \mu_{z_n}, \Sigma_{z_n})]$$

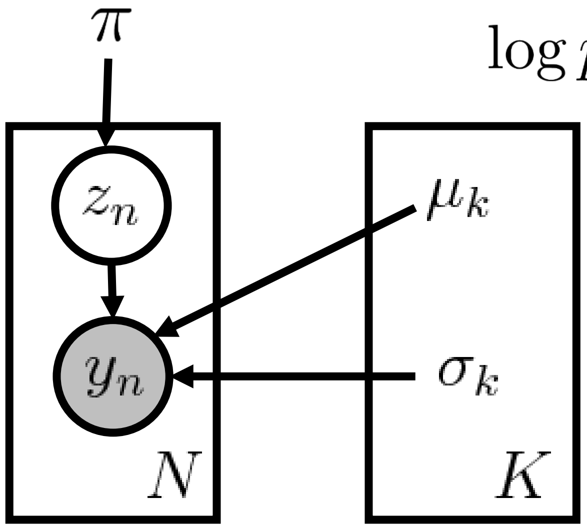
$$0 = -\sum_{n=1}^{N} q^{\text{new}}(z_n = k) \Sigma_k (y_n - \mu_k)$$



(b)

$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} q^{\text{new}}(z_n = k) y_n \quad \text{where} \quad N_k = \sum_{n=1}^{N} q(z_n = k)$$

Example: Gaussian Mixture Model



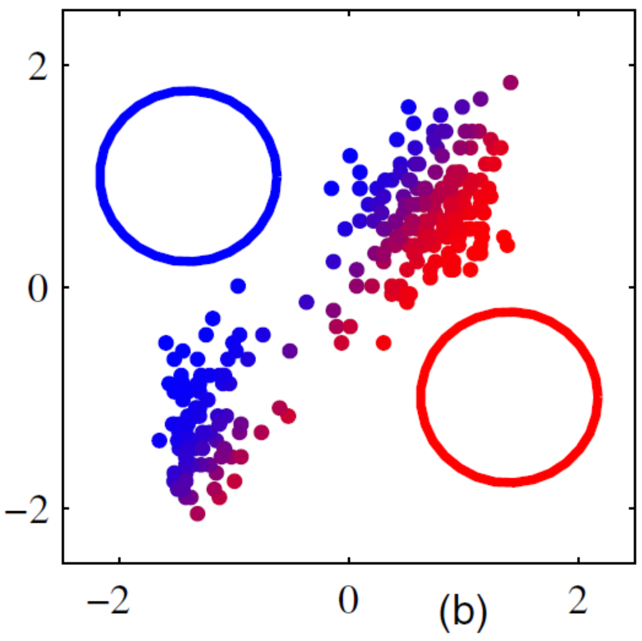
$$\log p(\mathcal{Y} \mid \pi, \mu, \Sigma) \geq \sum_{n=1}^{N} \sum_{k=1}^{K} q(z_n = k) \log \{\pi_k \mathcal{N}(y_n \mid \mu_k, \Sigma_k)\} = \mathcal{L}(q, \theta)$$

M-Step: $\theta^{\text{new}} = \arg\max_{\theta} \mathcal{L}(q^{\text{new}}, \theta)$

Repeat for remaining parameters,

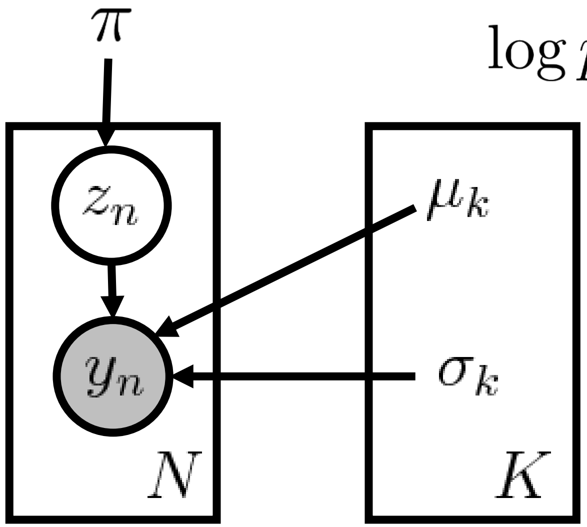
$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} q(z_n = k)(y_n - \mu_k^{\text{new}})(y_n - \mu_k^{\text{new}})^T$$

$$\pi_k^{\text{new}} = \frac{N_k}{N}$$



- Solving for mixture weights requires a bit more work
- Need constraint $\sum_k \pi_k = 1$
- Use Lagrange multiplier approach

Example: Gaussian Mixture Model



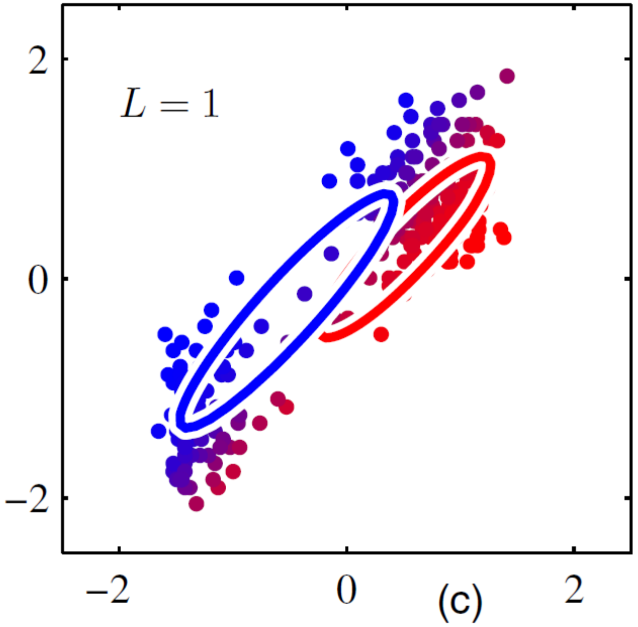
$$\log p(\mathcal{Y} \mid \pi, \mu, \Sigma) \geq \sum_{n=1}^{N} \sum_{k=1}^{K} q(z_n = k) \log \{\pi_k \mathcal{N}(y_n \mid \mu_k, \Sigma_k)\} = \mathcal{L}(q, \theta)$$

M-Step: $\theta^{\text{new}} = \arg\max_{\theta} \mathcal{L}(q^{\text{new}}, \theta)$

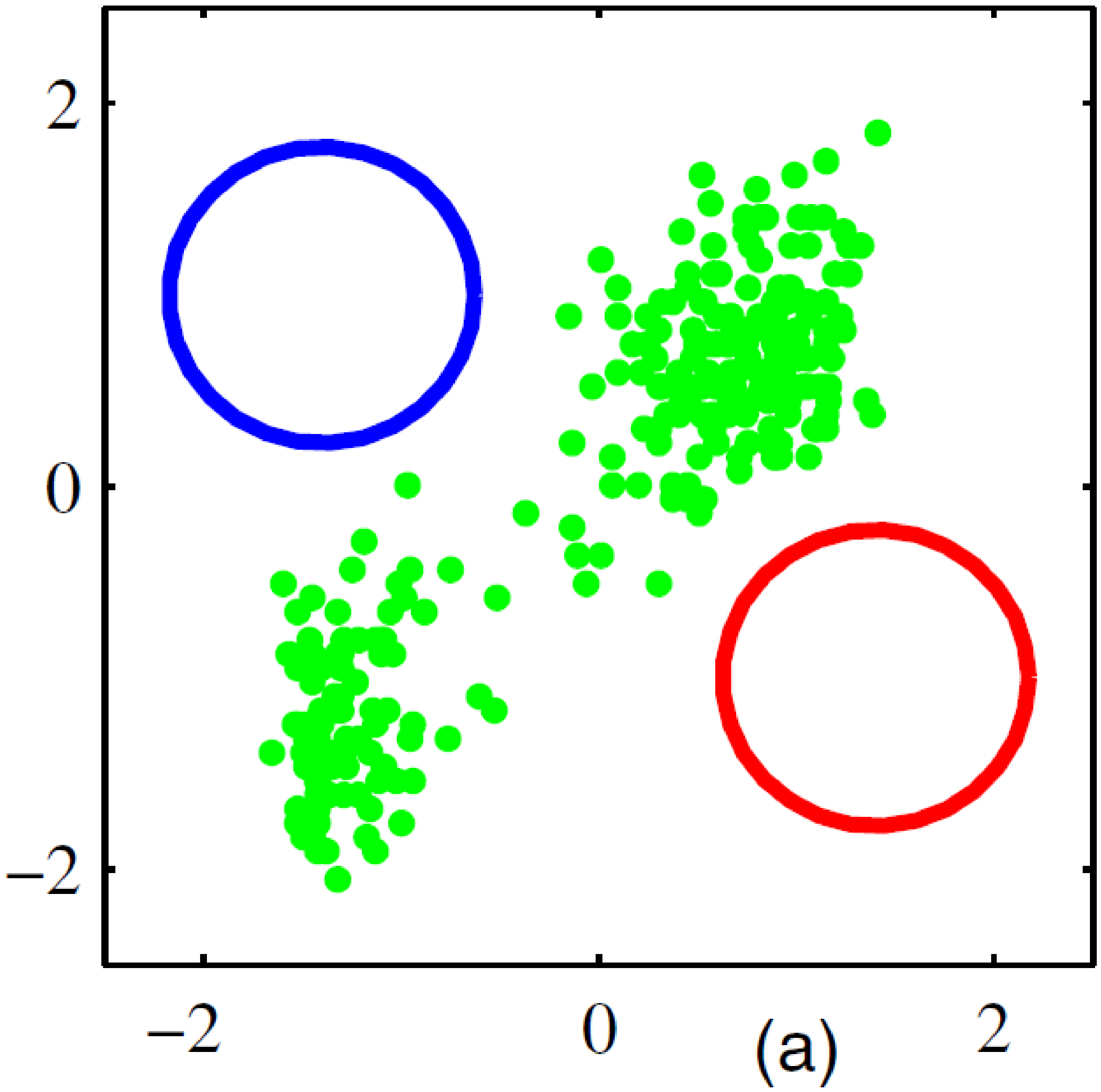
Repeat for remaining parameters,

$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} q(z_n = k)(y_n - \mu_k^{\text{new}})(y_n - \mu_k^{\text{new}})^T$$

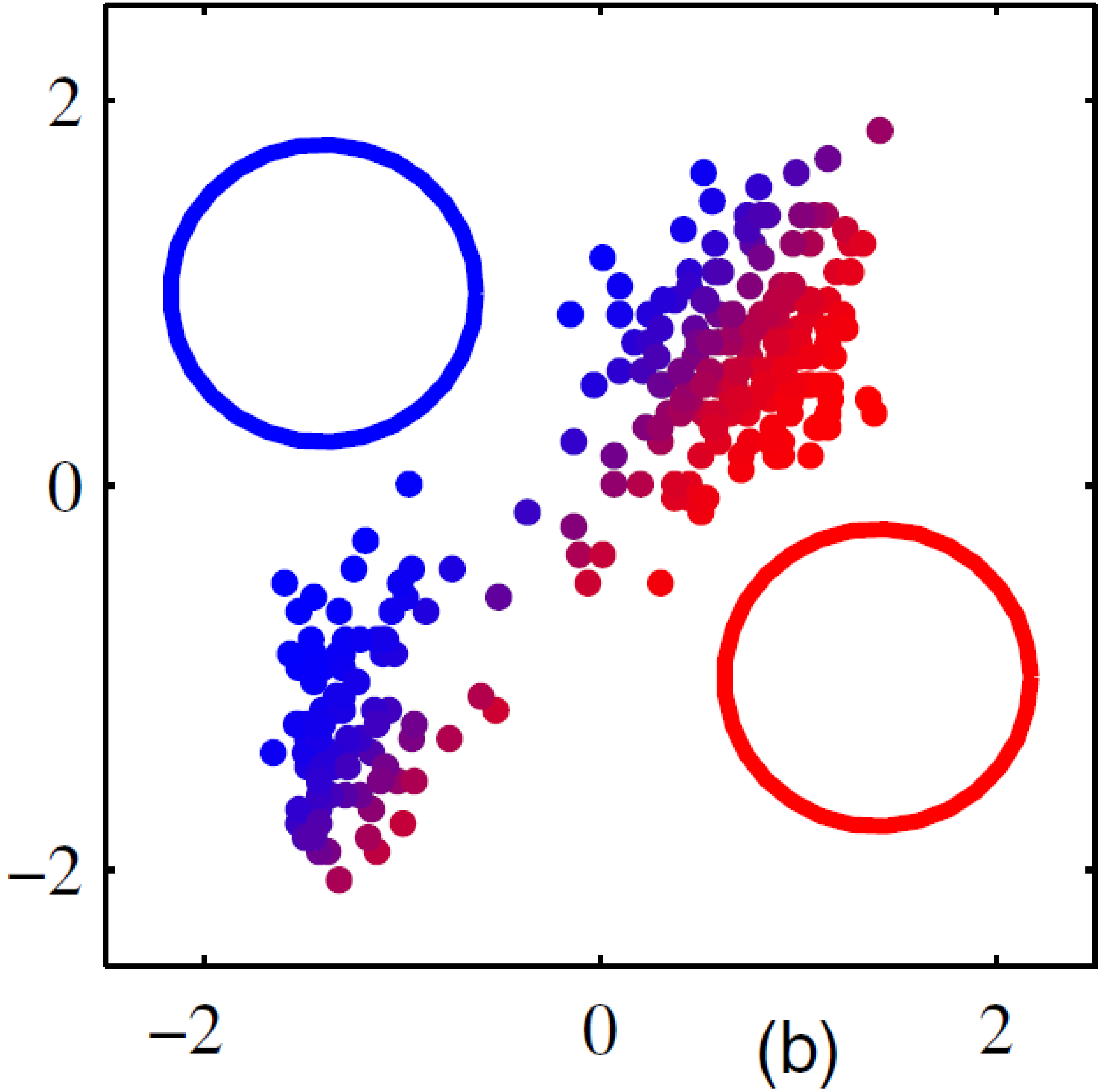
$$\pi_k^{\text{new}} = \frac{N_k}{N}$$



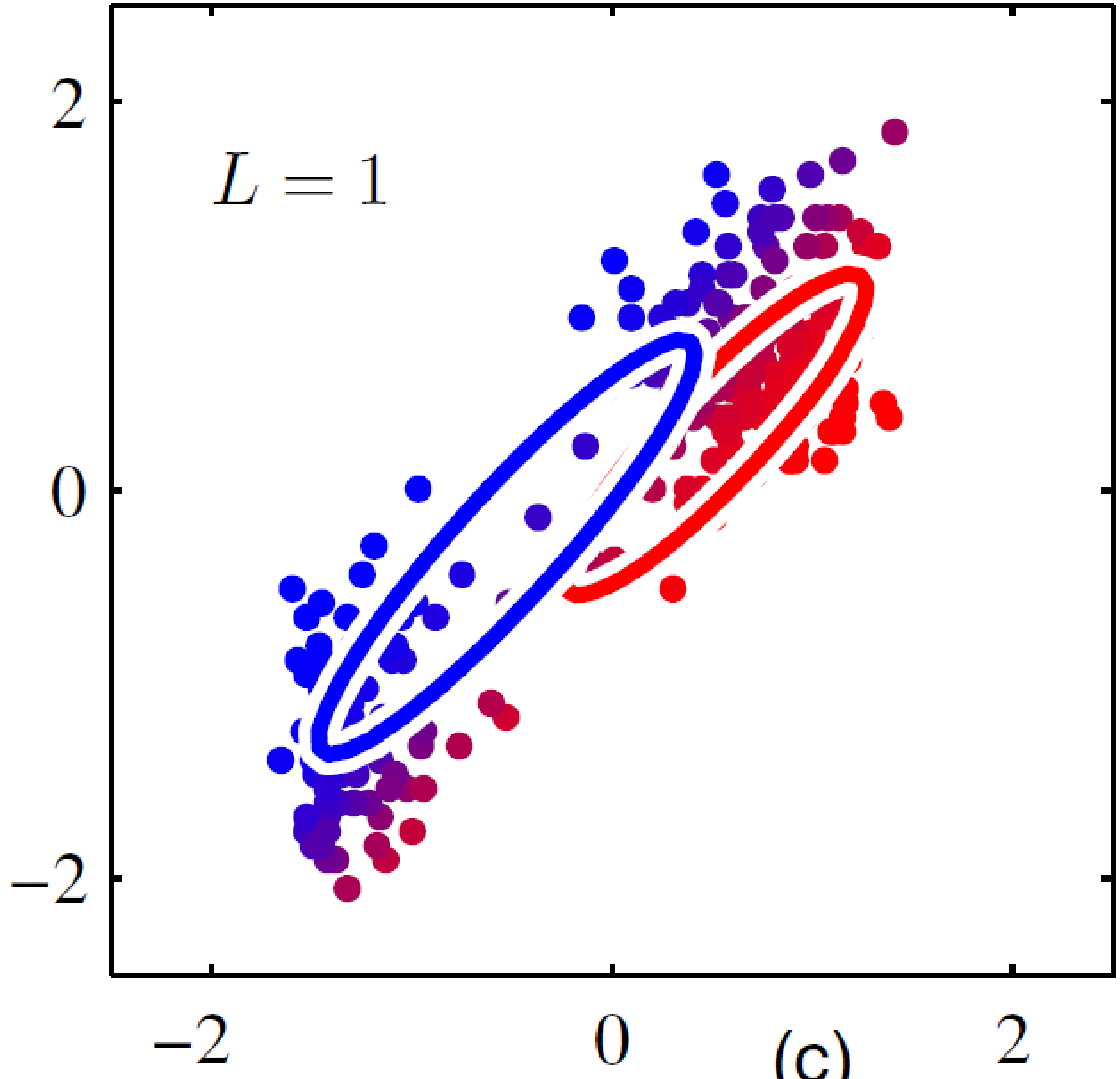
- Solving for mixture weights requires a bit more work
- Need constraint $\sum_k \pi_k = 1$
- Use Lagrange multiplier approach



Source: Chris Bishop, PRML

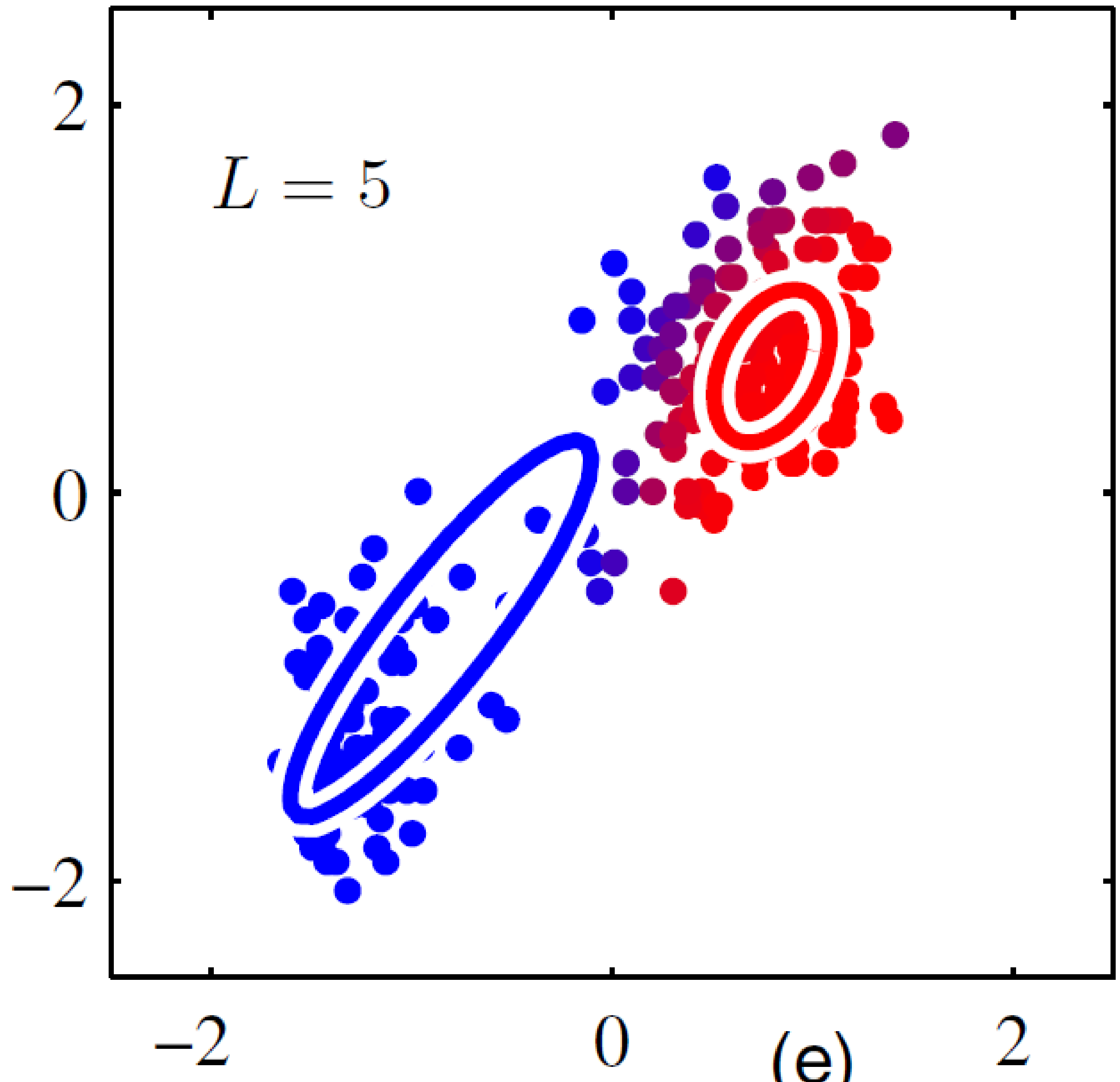


Source: Chris Bishop, PRML



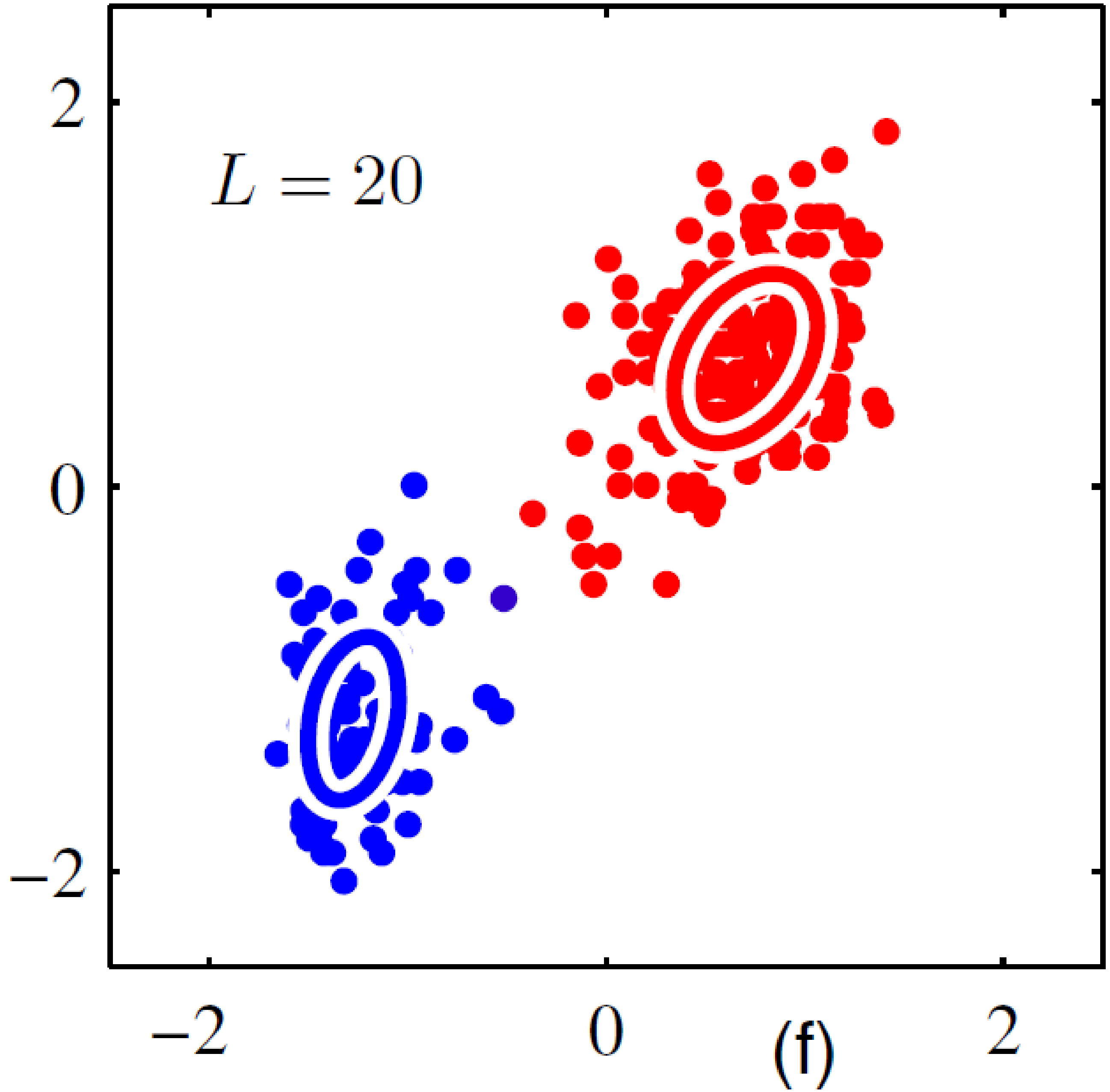
(c)

Source: Chris Bishop, PRML



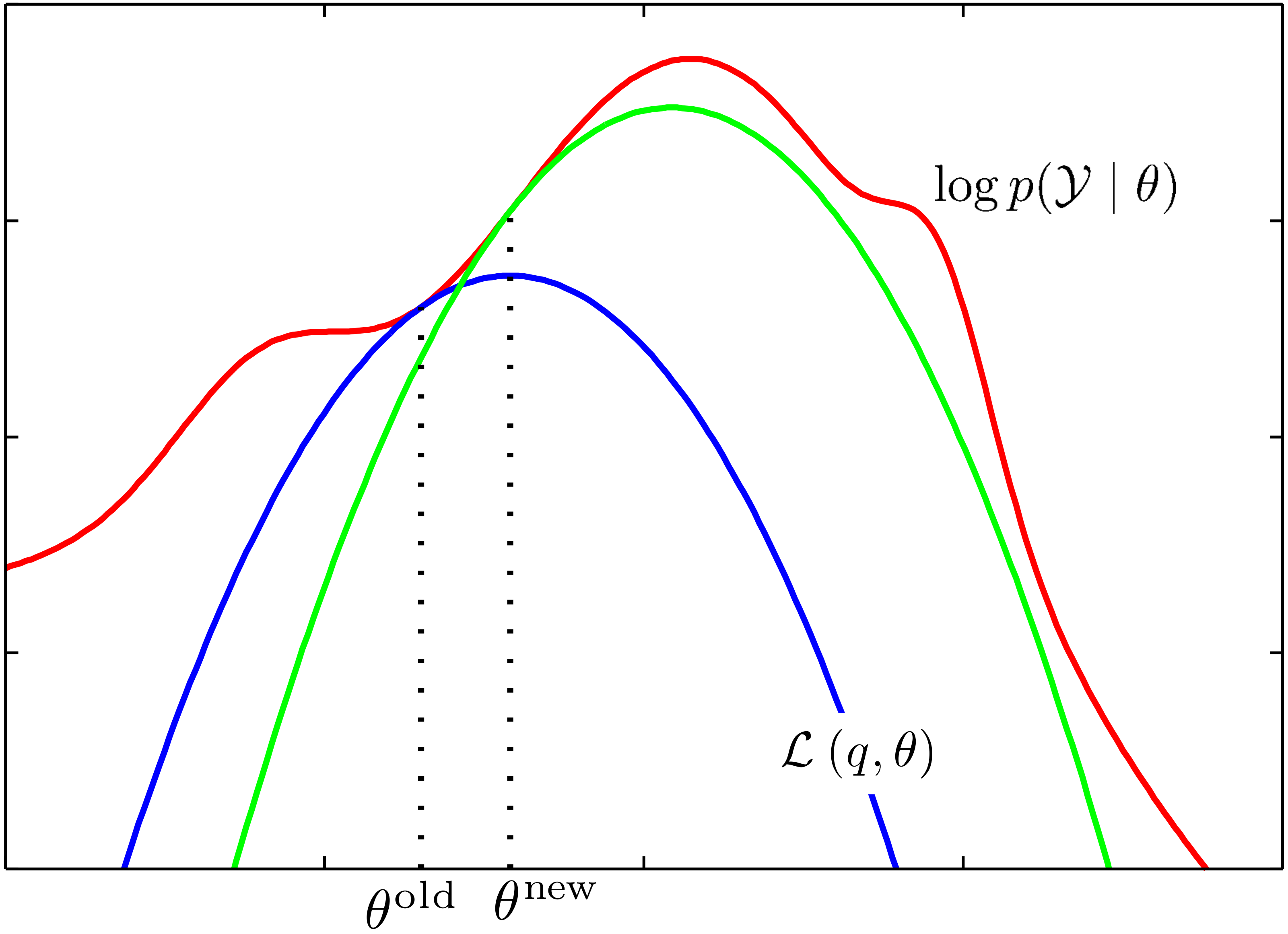
(e)

Source: Chris Bishop, PRML



Source: Chris Bishop, PRML

EM: A Sequence of Lower Bounds



C. Bishop, Pattern Recognition & Machine Learning

EM Lower Bound

$$\mathbf{E}_q\left[\log \frac{p(z,y\mid\theta)}{q(z)}\right] = \mathbf{E}_q\left[\log \frac{p(z,y\mid\theta)}{q(z)}\frac{p(y\mid\theta)}{p(y\mid\theta)}\right]$$ **(Multiply by 1)**

$$= \log p(y\mid\theta) - \mathrm{KL}(q(z)\|p(z\mid y,\theta))$$ **(Definition of KL)**

Bound gap is the Kullback-Leibler divergence KL(q||p),

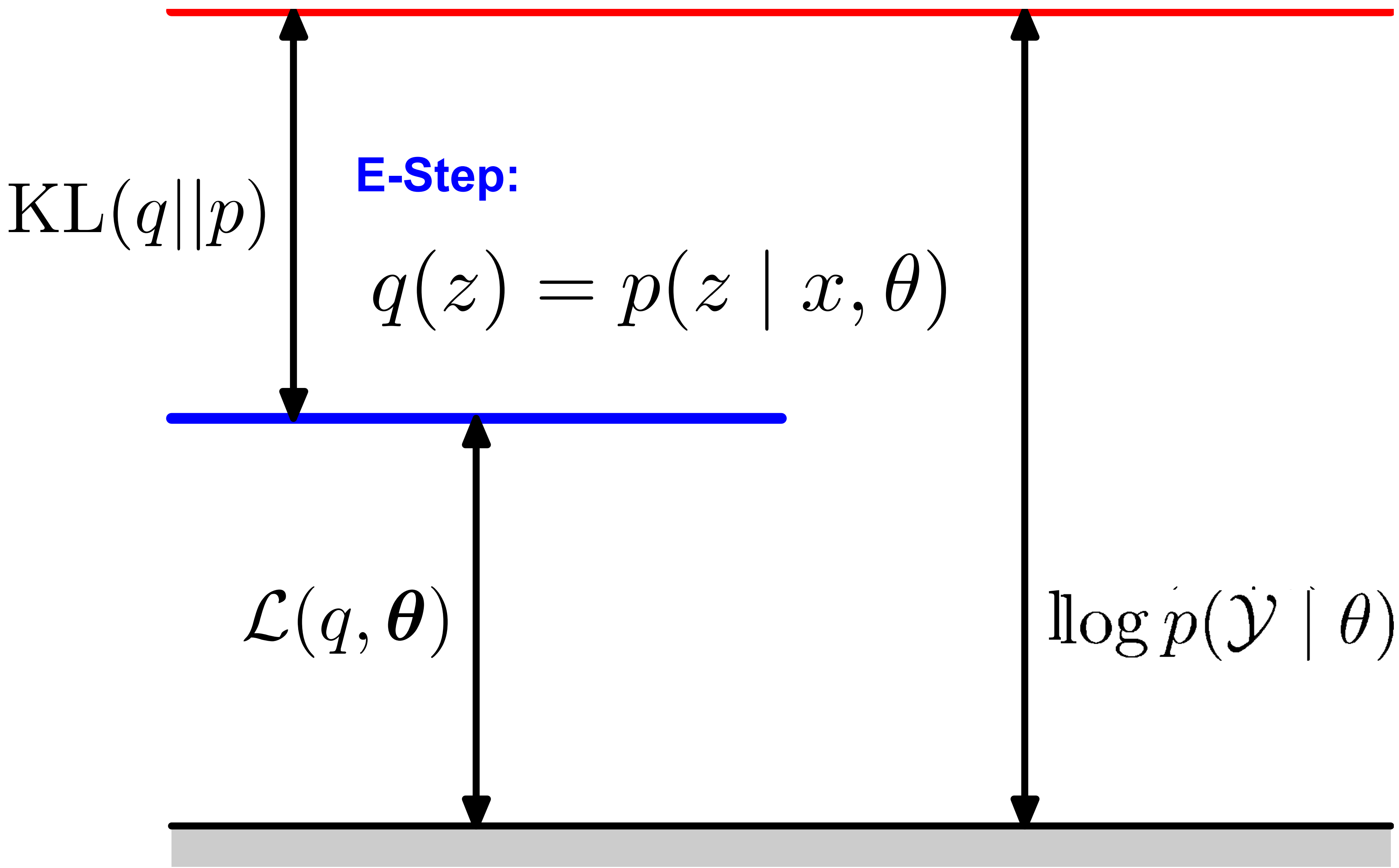
$$\mathrm{KL}(q(z)\|p(z\mid y,\theta)) = \sum_z q(z)\log\frac{q(z)}{p(z\mid y,\theta)}$$

- Similar to a “distance” between q and p

$$\mathrm{KL}(q\,||\,p) \geq 0 \text{ and } \mathrm{KL}(q\,||\,p) = 0 \text{ if and only if } q = p$$

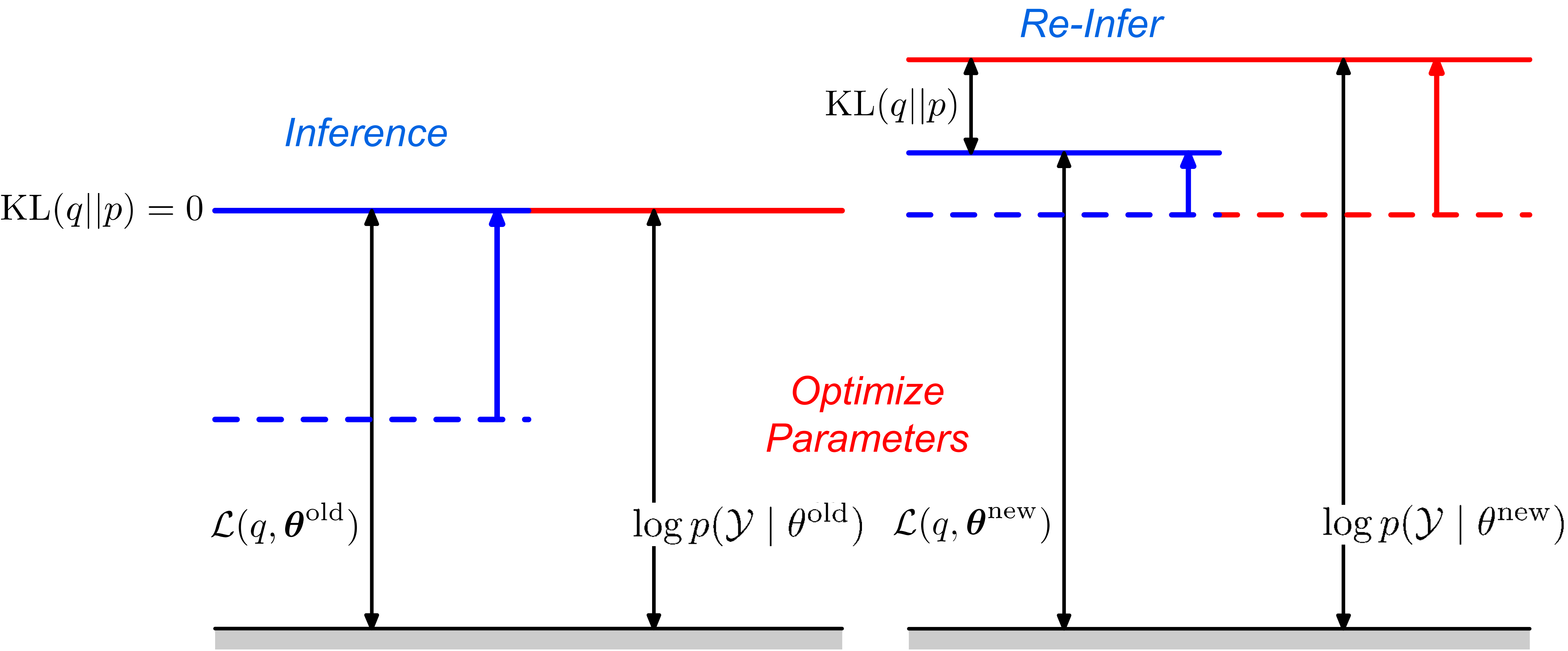
- This is why solution to E-step is $q(z) = p(z\mid y,\theta)$

Lower Bounds on Marginal Likelihood



C. Bishop, Pattern Recognition & Machine Learning

Expectation Maximization Algorithm



E Step: *Optimize distribution on hidden variables given parameters*

M Step: *Optimize parameters given distribution on hidden variables*

Properties of Expectation Maximization Algorithm

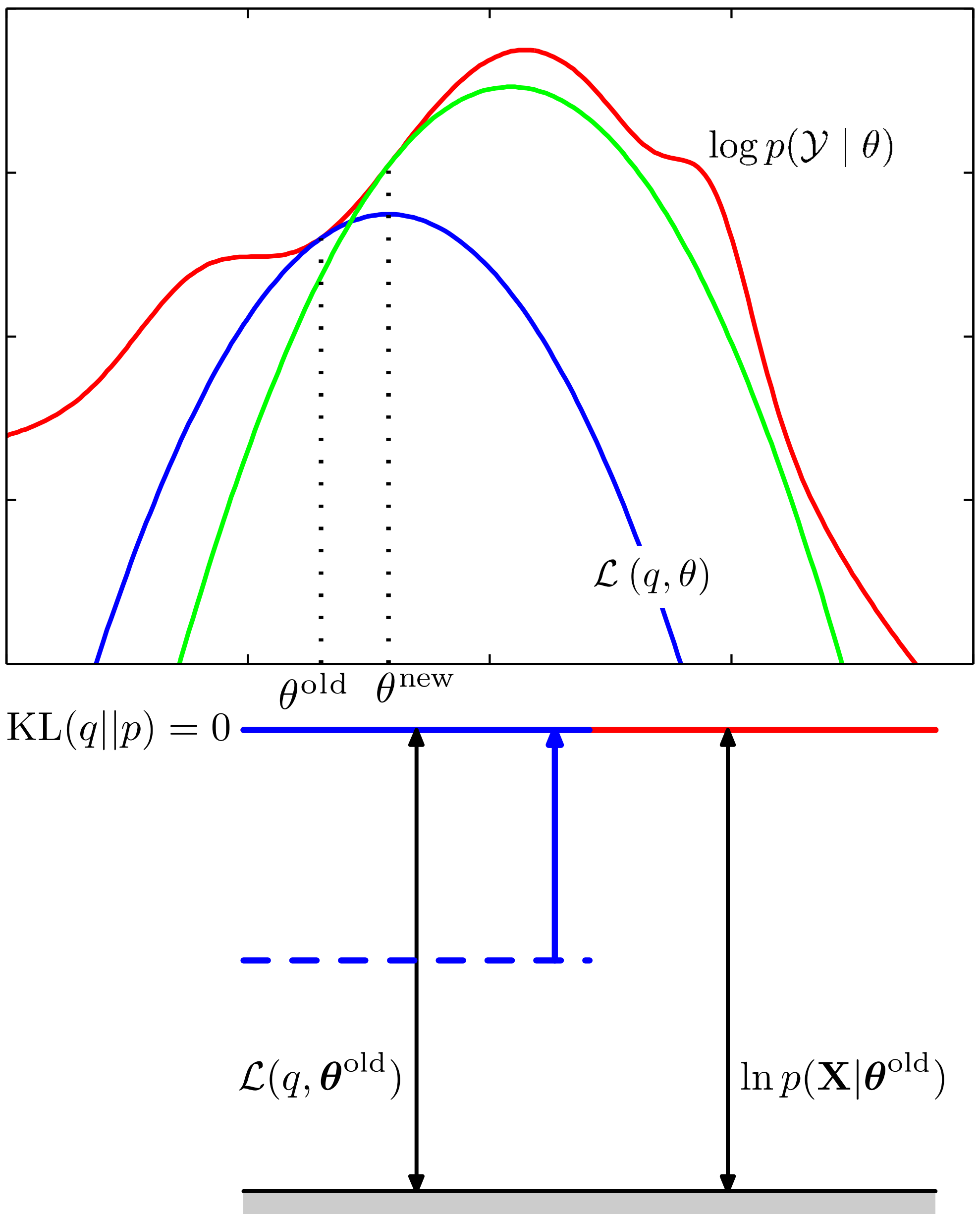
Sequence of bounds is monotonic,

$$\mathcal{L}(q^{(1)}, \theta^{(1)}) \leq \mathcal{L}(q^{(2)}, \theta^{(2)}) \leq \ldots \leq \mathcal{L}(q^{(T)}, \theta^{(T)})$$

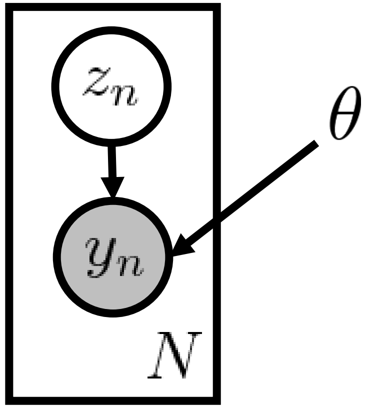
Guaranteed to converge
(**Pf.** Monotonic sequence bounded above.)

Converges to a local maximum of the marginal likelihood

After each E-step bound is tight at θ^{old} so likelihood calculation is exact (for those parameters)



MLE vs. MAP Estimation

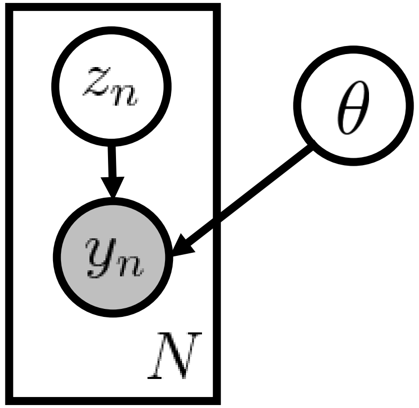


Conditional model,

$$p(z, y \mid \theta) = \prod_{n=1}^{N} p(z_n) p(y_n \mid z_n, \theta)$$

MLE estimate of unknown non-random parameters,

$$\theta^{\text{MLE}} = \arg\max_{\theta} \log p(\mathcal{Y} \mid \theta)$$



Generative model,

Corresponds to regularized MLE

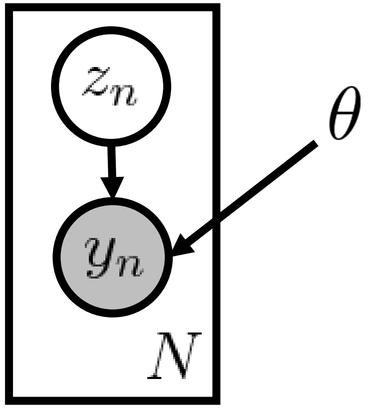
$$p(z, y, \theta) = p(\theta) \prod_{n=1}^{N} p(z_n) p(y_n \mid z_n, \theta)$$

MAP estimate of random parameters,

$$\theta^{\text{MAP}} = \arg\max_{\theta} \log p(\theta) + \log p(\mathcal{Y} \mid \theta)$$

EM Lower Bound

Recall EM lower bound of marginal likelihood



$$\arg\max_\theta \log p(\mathcal{Y} \mid \theta) = \arg\max_\theta \log \sum_z p(z, \mathcal{Y} \mid \theta)$$

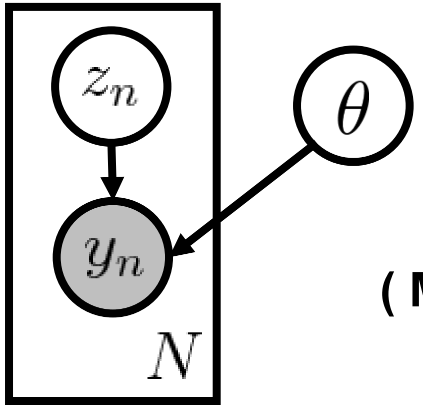
(Multiply by q(z)/q(z)=1) $$= \log \sum_z p(z, \mathcal{Y} \mid \theta) \left(\frac{q(z)}{q(z)} \right)$$

(Definition of Expected Value) $$= \log \mathbf{E}_q \left[\frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right]$$

(Jensen's Inequality) $$\geq \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right]$$

MAP EM

Bound holds with addition of log-prior



$$\arg\max_\theta \log p(\theta \mid \mathcal{Y}) = \arg\max_\theta \log \sum_z p(z, \mathcal{Y} \mid \theta) + \log p(\theta)$$

(Multiply by q(z)/q(z)=1) $$= \log \sum_z p(z, \mathcal{Y} \mid \theta) \left(\frac{q(z)}{q(z)} \right) + \log p(\theta)$$

(Definition of Expected Value) $$= \log \mathbf{E}_q \left[\frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] + \log p(\theta)$$

(Jensen's Inequality) $$\geq \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] + \log p(\theta)$$

MAP EM

$$\max_{\theta} \log p(\theta, \mathcal{Y}) \geq \max_{q,\theta} \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] + \log p(\theta)$$

E-Step: Fix parameters and maximize w.r.t. q(z),

$$q^{\text{new}} = \arg\max_{q} \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta^{\text{old}})}{q(z)} \right] + \boxed{\log p(\theta^{\text{old}})}$$

Constant in q(z)

Same solution as standard maximum likelihood EM,

$$q^{\text{new}} = p(z \mid \mathcal{Y}, \theta^{\text{old}})$$

M-Step: Fix q(z) and optimize parameters,

$$\theta^{\text{new}} = \arg\max_{\theta} \mathbf{E}_{q^{\text{new}}} \left[\log p(\mathcal{Y} \mid z, \theta) \right] + \log p(\theta)$$

MAP EM

Initialize Parameters: $\theta^{(0)}$

At iteration t do:

E-Step: $q^{(t)}(z) = p(z \mid y, \theta^{(t-1)})$

M-Step: $\theta^{(t)} = \arg\max_\theta \mathcal{L}(q^{(t)}, \theta) + \log p(\theta)$

Until convergence

E-Step Compute **expected** log-likelihood under the posterior distribution,

$$q^{(t)}(z) = p(z \mid y, \theta^{(t-1)}) \qquad \mathbf{E}_{q^{(t)}}[\log p(y \mid z, \theta)] = \mathcal{L}(q^{(t)}, \theta)$$

M-Step Maximize expected log-likelihood,

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

EM Summary

Approximate MLE for intractable marginal likelihood via lower bound,

$$\max_{\theta} \log p(\mathcal{Y} \mid \theta) \geq \max_{q,\theta} \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] \equiv \mathcal{L}(q, \theta)$$

Coordinate ascent alternately maximizes $q(z)$ and θ,

E-Step

$$q^{\text{new}} = \arg\max_{q} \mathcal{L}(q, \theta^{\text{old}})$$

M-Step

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

Solution to E-step sets q to posterior over hidden variables,

$$q^{\text{new}}(z) = p(z \mid \mathcal{Y}, \theta^{\text{old}})$$

M-step is problem-dependent, requires gradient calculation

EM Summary

Easily extends to (approximate) MAP estimation,

$$\max_{\theta} \log p(\theta \mid \mathcal{Y}) \geq \max_{q,\theta} \mathbf{E}_q \left[\log \frac{p(z, \mathcal{Y} \mid \theta)}{q(z)} \right] + \log p(\theta) + \text{const.}$$

E-step unchanged / Slightly modifies M-step,

E-Step

$$q^{\text{new}} = \arg\max_{q} \mathcal{L}(q, \theta^{\text{old}}) = p(z \mid \mathcal{Y}, \theta^{\text{old}})$$

M-Step

$$\theta^{\text{new}} = \arg\max_{\theta} \mathcal{L}(q^{\text{new}}, \theta) + \log p(\theta)$$

Properties of both MLE / MAP EM

- Monotonic in $\mathcal{L}(q, \theta)$ or $\mathcal{L}(q, \theta) + \log p(\theta)$ (for MAP)
- Provably converge to local optima (hence approximate estimation)

Learning Summary

Maximum likelihood estimation (MLE) maximizes (log-)likelihood func,

$$\theta^{\mathrm{MLE}} = \arg\max_{\theta} \log p(\mathcal{Y} \mid \theta) \equiv \mathcal{L}(\theta)$$

Where parameters are *unknown non-random* quantities

Tendency to *overfit* training data mitigated by inclusion of regularizer,

$$\hat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta) - \lambda \mathcal{R}(\theta)$$

For linear-Gaussian models θ^{MLE} and $\hat{\theta}$ have closed-form leading to:

- Least-squares estimation
- Ridge regression (L2 regularized least-squares)
- LASSO regression (L1 regularized least-squares)

Learning Summary

Maximum a posteriori (MAP) maximizes posterior probability,

$$\theta^{\mathrm{MAP}} = \arg\max_{\theta} \log p(\theta \mid \mathcal{Y}) = \arg\max_{\theta} \mathcal{L}(\theta) + \log p(\theta)$$

Parameters are *random* quantities with prior $p(\theta)$.

Corresponds to regularized MLE for specific prior/regularizer pair,

$$\hat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta) - \lambda \mathcal{R}(\theta)$$

Gaussian prior=L2, Laplacian prior=L1

Straightforward sequential updating, e.g. Bayesian linear regression

Learning Summary

- Most models will not yield closed-form MLE/MAP estimates
- Gradient-based methods optimize log-likelihood function

$$\theta^{k+1} = \theta^k + \beta \nabla_\theta \mathcal{L}(\theta^k)$$

- Expectation Maximization (EM) alternative to gradient methods
- Both approaches approximate for non-convex models