Expectation-Maximization, GMM, and Nonparametric Methods

STAT261: Introduction to Machine Learning

Prof. Allie Fletcher

Outline

- Clustering - Deterministic versus Probabilistic Models
  - K-means
- Mixture Distributions
- Expectation Maximization Algorithm
- Convergence of EM
- Conjugate Priors*--not covered in lecture, should know
- Optimization Review-iterative methods** (Optional at this time, but it is something you should understand and may be helpful. We may cover this in the next few lectures.)

K-means: Iterative clustering

- Simple iterative algorithm to:
  - μ_k = mean of each cluster (hence "K-means")
  - $C_n \in \{1, \ldots K\}$ = cluster of sample x_n
- Step 0: Start with guess for centroids: μ_k
- Step 1: Assign x_n to closest mean cluster

$$C_n = \arg\min_k ||x_n - \mu_k||^2$$

- Step 2: Update mean of each cluster:

$$\mu_k = \text{average of } x_n \text{ for } x_n \; with \; C_n = k$$

- Return to step 1

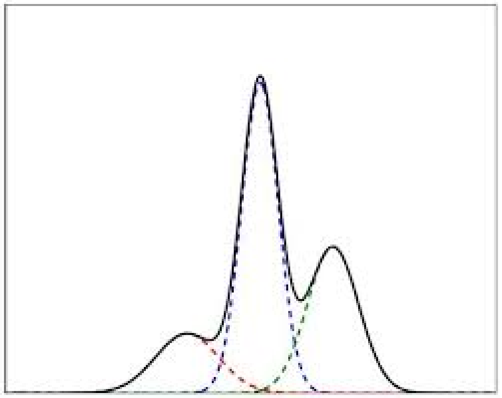
Probabilistic Mixture Model for Clusters

- Random variable $z \in \{1, \dots, K\}$
  - Discrete event with PMF: $P(z = i)$
  - Latent variable: often not directly observed
- Observed variable x, can be continuous
  - Probability depends on z, $p(x|z = i)$
  - One PDF, or component per state $z = i$
- Distribution of x: computed via total probability
  - PDF $p(x) = \sum p(x|z = i) P(z = i)$
  - CDF $F(x_0) = \sum P(x \leq x_0 | z = i) P(z = i)$
- Example: Mixture of two Gaussians

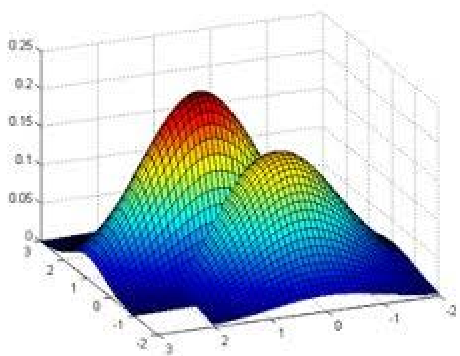
Gaussian Mixture Models

- Each $p(x|z = i)$ is a Gaussian
- Parametrized by:
  - $q_i = P(z = i) =$ Probability of each component
  - $\mu_i = E(x|z = i), P_i = var(x|z = i)$
    mean and variance in each component
- Can be vector valued

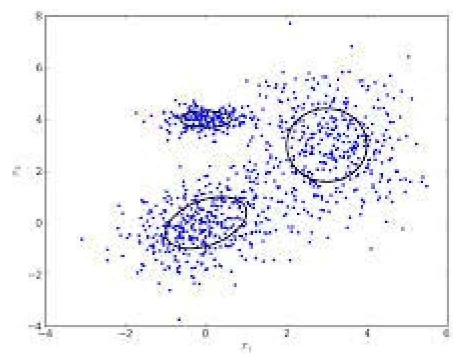
Visualizing GMMs



- 1d model with $K = 3$ components



- PDF for 2d GMM with $K = 2$ components



- Random points from a GMM with $K = 3$ components

Expectation and Variance

- Can compute expectation and variance by total probability
  - Expectation: $\mu = E(x) = \sum q_i \mu_i$
  - Variance:

$$var(x) = \sum_i \underbrace{q_i P_i}_{\text{Variance within component}} + \underbrace{q_i(\mu_i - \mu)(\mu_i - \mu)^T}_{\text{Variance between components}}$$

- Proof on board

Expectation & variance of mix. model

① Expectation: Use total probability.

$$E(x) = \sum_i E(x|z=i)\, P(z=i) \quad \text{(total prob.)}$$

$$\boxed{= \sum_i \mu_i q_i}$$

② Variance

$$\mathrm{var}(X) = E(XX^T) - \mu\mu^T$$

$$E(XX^T) = \sum_i E(XX^T|z=i)\, P(z=i)$$

$$= \sum_i \{\mathrm{var}(XX^T|z=i) + \mu_i\mu_i^T\} q_i$$

$$= \sum_i q_i P_i + q_i \mu_i \mu_i^T$$

Hence,

$$\mathrm{var}(X) = \left[\sum q_i P_i + q_i \mu_i \mu_i^T\right] - \mu\mu^T$$

Now,

$$\sum q_i (\mu - \mu_i)(\mu - \mu_i)^T$$

$$= \sum q_i \mu\mu^T - \sum q_i \mu \mu_i^T - \sum q_i \mu_i \mu^T + \sum q_i \mu_i \mu_i^T$$

$$= \mu\mu^T - 2\mu\mu^T + \sum q_i \mu_i \mu_i^T \quad \left(\text{Since } \sum q_i = 1,\ \sum q_i \mu_i = \mu\right)$$

$$= \sum q_i \mu_i \mu_i^T - \mu\mu^T$$

$$\therefore \boxed{\mathrm{var}(X) = \sum q_i P_i + q_i (\mu - \mu_i)(\mu - \mu_i)^T}$$

Estimating the Latent Variable

- Given x, can we estimate z if we knew parameters:
- Use Bayes' rule:

$$P(z = i|x) = \frac{P(x|z = i)q_i}{\sum_k P(x|z = k)q_k}$$

- Example: Scalar Gaussian
  - Illustration on board

Scalar Gaussian ($k=2$ clusters)

→ Suppose $p(x|z=i) = N(x|\mu_i, \sigma^2)$

(same variance)

→ Assume $\mu_2 > \mu_1$

→ From Bayes' rule:

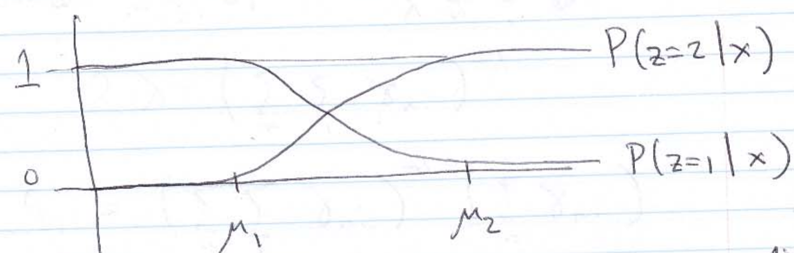
$$P(z=1|x) = \frac{P(x|z=1)\,q_1}{P(x|z=1)q_1 + P(x|z=2)q_2}$$

$$= \frac{\exp(-(x-\mu_1)^2/2\sigma^2)\,q_1}{\exp(-(x-\mu_1)^2/2\sigma^2)q_1 + \exp((-x-\mu_2)^2/2\sigma^2)q_2}$$

When $x \to \infty$ $\exp\left(-\frac{(x-\mu_1)^2}{2\sigma^2}\right) >> \exp\left(-\frac{(x-\mu_2)^2}{2\sigma^2}\right)$

$\Rightarrow P(z=1|x) \to 0$

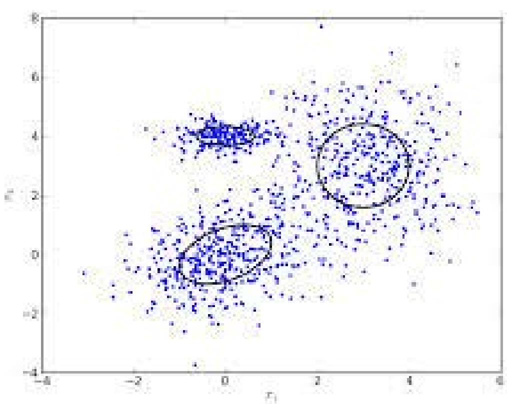
When $x \to -\infty$ $\Rightarrow P(z=1|x) \to 1$



$z=1$ more likely $\Leftarrow$ $\Rightarrow$ $z=2$ more likely

Fitting a Mixture Model

- Given data $x = (x_1, \dots, x_N)$
- Find GMM parameters
  - Mean and variance in each component
  - Probability of each component
- Can be interpreted as "clustering"
- Parametric probabilistic model versus K-means



UCLA

Maximum Likelihood Estimation

- Unknown parameters in GMM:
$$\theta = (q_1, \dots, q_K, \mu_1, \dots, \mu_K, P_1, \dots, P_K)$$
- Data $x = (x_1, \dots, x_N)$

- Likelihood of x_n:
$$p(x_n|\theta) = \sum_{k=1}^{K} p(x_n|z_n = k, \theta) P(z_n = k|\theta) = \sum_{k=1}^{K} q_k N(x_n|\mu_k, P_k)$$
- Negative log likelihood of all data
$$L(\theta) = -\ln p(x|\theta) = -\sum_{n=1}^{N} \ln \left[\sum_{i=1}^{K} q_i N(x_n|\mu_i, P_i) \right]$$
- ML estimation:
$$\hat{\theta} = \arg\min_{\theta} L(\theta)$$

Outline

- Clustering - Deterministic versus Probabilistic Models
  - K-means
- Mixture Distributions
- Expectation Maximization Algorithm
- Convergence of EM
- Conjugate Priors*--not covered in lecture but important
- Optimization Review-iterative methods

Expectation Maximization Algorithm

- Optimization of $L(\theta)$ is hard
  - No simple way to directly optimize
  - Likelihood is non-convex
  - $L(\theta) = -\ln p(x|\theta) = -\sum_{n=1}^{N} \ln\left[\sum_{i=1}^{K} q_i N(x_n|\mu_i, P_i)\right]$

- Expectation maximization:
  - Simple iterative procedure:
  - Generates a sequence of estimates $\hat{\theta}^0, \hat{\theta}^1, \ldots$
  - Attempts to approach MLE

$$\hat{\theta}^k \to \arg\min_{\theta} L(\theta)$$

UCLA

EM Steps

- E-step: Estimate the latent variables
  - Find the posterior of the latent variables given $\hat{\theta}^k$
    $$P(z|x, \theta = \hat{\theta}^k)$$
  - Compute function, Q, auxiliary function
    $$Q(\theta, \hat{\theta}^k) := E[\ln p(x, z|\theta)|\hat{\theta}^k]$$
    $$= \sum_z \ln p(x, z|\theta) \, P(z|x, \theta = \hat{\theta}^k)$$
- M-step: Update parameters
  $$\hat{\theta}^{k+1} = \arg\max_{\theta} Q(\theta, \hat{\theta}^k)$$

E-Step for a GMM: Finding the posterior

- Given parameters q_i, μ_i, P_i (estimated, where the i is the class)
- Find posterior of the latent variables (underlying sample classes) by Bayes rule
  - N samples, so we have N latent variables

$$\gamma_{ni} = P(z_{nj} = i | x) = \frac{P(x_n | z_j = i) q_i}{\sum_l P(x_n | z_j = l) q_l}$$

$$= \frac{N(x_n | \mu_i, P_i) q_i}{\sum_l N(x_j | \mu_l, P_l) q_l}$$

- A "soft" selection

E-Step for a GMM: "Expected" likelihood

- Auxilliary function separates

$$Q\left(\theta, \hat{\theta}^k\right) = E_z\left[\ln p(x, z|\theta)|\hat{\theta}^k\right]$$

$$= \sum_{i=1}^{K} \sum_{n=1}^{N} \gamma_{ni} \ln P(x_n, z_n = i|\theta)$$

$$= \sum_{i=1}^{K} \sum_{n=1}^{N} \gamma_{ni} [\ln q_i + \ln N(x_n|\mu_i, P_i)]$$

M-Step for the GMM

- Maximize $Q\left(\theta, \hat{\theta}^k\right), \sum_{i=1}^{K} \sum_{n=1}^{N} \gamma_{ni} [\ln q_i + \ln N(x_n | \mu_i, P_i)]$
- Update for q_i (proof on board)

$$q_i = \frac{N_i}{\sum_j N_j}, \qquad N_i = \sum_n \gamma_{ni}$$

- Update for μ_i

$$\mu_i = \frac{1}{N_i} \sum_n \gamma_{ni}\, x_n$$

- Update for P_i

$$P_i = \frac{1}{N_i} \sum_n \gamma_{ni}\, (x_n - \mu_i)(x_n - \mu_i)\wedge T$$

M step minimizations for GMM

$$Q(\theta, \hat{\theta}^k) = \sum_{i=1}^{k} \sum_{n=1}^{N} \gamma_{ni} \left[\ln q_i + \ln \mathcal{N}(x_n | \mu_i, P_i) \right]$$

① Min. over q_i

$$\hat{q}_1, \ldots, \hat{q}_n = \arg\min Q(\theta, \hat{\theta}^k) \quad \text{s.t. } \sum q_i = 1$$

Let $L(q) = Q(\theta, \hat{\theta}^k) + \lambda\left(\sum_i q_i - 1\right)$

$$= \sum_i \sum_n \gamma_{ni} \ln q_i + \lambda \sum_i q_i + \text{const}$$

const = terms that do not depend on q_i

$$\frac{\partial L}{\partial q_i} = \frac{\sum_n \gamma_{ni}}{q_i} + \lambda = 0$$

$$q_i = -\frac{1}{\lambda} \sum_n \gamma_{ni}$$

Since $\sum q_i = 1 \Rightarrow -\frac{1}{\lambda} \sum_n \sum_i \gamma_{ni} = 1$

$$\Rightarrow \lambda = \left(\sum_n \sum_i \gamma_{ni}\right)^{-1}$$

$$q_i = \left(\sum_n \sum_i \gamma_{ni}\right)^{-1} \left(\sum_n \gamma_{ni}\right)$$

Define $N_i = \sum_n \gamma_{ni}$ = ~~number~~ weight of samples in class i

Then $N = \sum_i N_i$

$$\boxed{\hat{q}_i = \frac{N_i}{N}} = \text{ratio of sample weights.}$$

(2) Minimization over μ_i

$$Q(\theta, \hat{\theta}) = \sum_{n\,i} \gamma_{ni} \ln \mathcal{N}(x_n | \mu_i, P_i) + \text{const}$$

$$= -\sum_{n\,i} \frac{1}{2} \gamma_{ni} (x_n - \mu_i)^T P_i (x_n - \mu_i) + \text{const}$$

$$\frac{\partial Q}{\partial \mu_i} = -\sum_n \gamma_{ni} P_i (x_n - \mu_i) = 0$$

$$\Rightarrow \left(\sum_n \gamma_{ni}\right) \mu_i = \sum \gamma_{ni} x_n$$

$$\Rightarrow \boxed{\mu_i = \frac{1}{N_i} \sum \gamma_{ni} x_n}$$

③ Minimization over P_i

~~Lin alg fact~~

Need to take deriv. w.r.t. matrix P_i

Consider perturbation $P_i + \Delta_i$

Two lin. alg facts:

(a) $v^T (P_i + \Delta_i)^{-1} v \cong v^T P_i^{-1} v - v^T P_i^{-1} \Delta_i P_i^{-1} v$
$+ O(\|\Delta_i\|^2)$

(b) $\ln \det(P_i + \Delta_i) \cong \mathrm{Tr}(P_i^{-1} \Delta_i)$

$$Q(\theta, \hat{\theta}^k) = \sum_{ni} \gamma_{ni} \ln N(x_n | \mu_i, P_i) + \text{const}$$

$$= -\sum_{ni} \frac{1}{2} \ln \det(P_i) + \frac{1}{2}(x_n - \mu_i)^T P_i^{-1} (x - \mu_i)$$

$$\frac{\partial Q(\theta, \hat{\theta}^k)}{\partial P_i} \cdot \Delta_i = \text{change in direction } \Delta_i$$

$$= -\sum_n \gamma_{ni} \frac{1}{2} \mathrm{Tr}(P_i^{-1} \Delta_i) + (x_n - \mu_i)^T P_i^{-1} \Delta_i P_i^{-1} (x_n - \mu_i)$$

~~$= \frac{1}{2} \sum_n \gamma_{ni} \mathrm{Tr}[(P_i^{-1} - P_i^{-1}(x_n - \mu_i)^T ($~~

$$= -\frac{1}{2} \sum_n \gamma_{ni} \, Tr\left\{ \left[P_i^{-1} - P_i^{-1}(x_n - \mu_i)(x_n - \mu_i) P_i^{-1} \right] \Delta_i \right\}$$

$$= -\frac{1}{2} Tr\left[S_i \cdot \Delta_i \right]$$

where $S_i = \sum_n \gamma_{ni} \left[P_i^{-1} - P_i^{-1}(x_n - \mu_i)(x_n - \mu_i)^T P_i^{-1} \right]$

Now, we need $\frac{\partial Q}{\partial P_i} \cdot \Delta_i = 0$ for all Δ_i

This occurs when $S_i = 0$

Hence

$$\sum_n \gamma_{ni} P_i^{-1} = \sum \gamma_{ni} P_i^{-1}(x_n - \mu_i)(x_n - \mu_i)^T P_i^{-1}$$

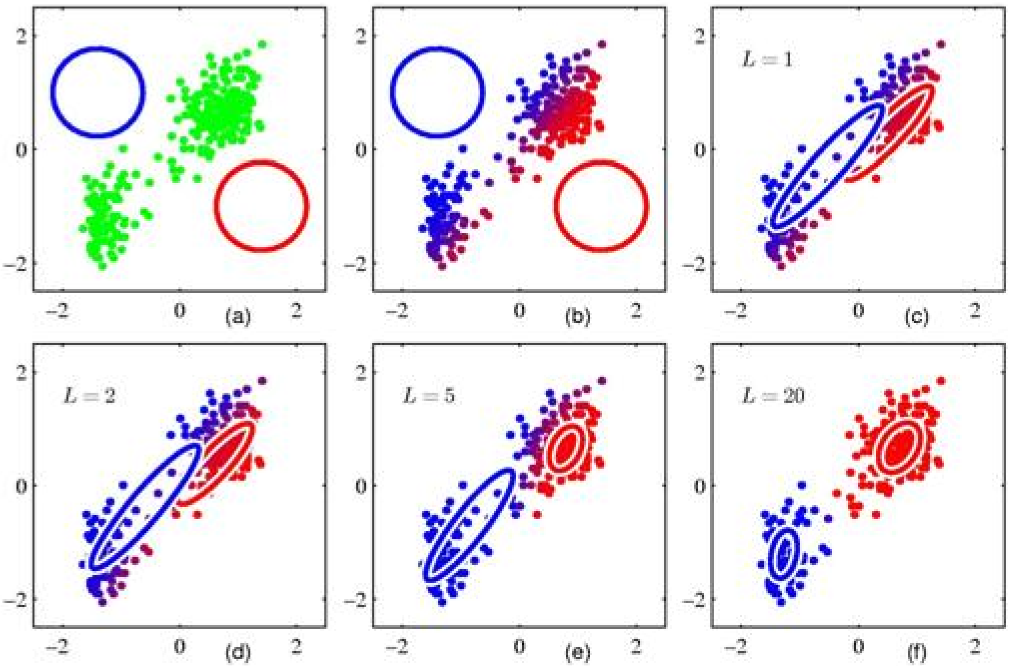
$$\Rightarrow \sum_n \gamma_{ni} P_i = \sum \gamma_{ni} (x_n - \mu_i)(x_n - \mu_i)^T$$

$$\boxed{\hat{P}_i = \left(\sum_n \gamma_{ni} \right)^{-1} \left(\sum \gamma_{ni} (x_n - \mu_i)(x_n - \mu_i)^T \right)}$$

Relation to K means

- EM can be seen as a "soft" version
  - In K-Means: $\gamma_{ni} = 1$ or 0
- Variance
  - In K-means: $P_i = I$
  - In EM, this is estimated
  - K-means, finds clustering assuming they all I matrix
- EM provides "scaling" of various dimensions
  - Rotated, ellipses
  - Scales difference or variances in data to shape covariance matrix
  - K-mean, you should normalize data

EM Illustrated



- Simple example with K=2 clusters
- Dimension = 2
- Convergence from a bad initial condition

Outline

- Clustering - Deterministic versus Probabilistic Models
  - K-means
- Mixture Distributions
- Expectation Maximization Algorithm
- Convergence of EM
- Conjugate Priors*--not covered in lecture but important
- Optimization Review-iterative methods

Majorization Minimization

- Suppose we wish to minimize $f(\theta)$
- MM algorithm: find a majorizing function $F(\theta, \theta^k)$:
  - $f(\theta^k) = F(\theta^k, \theta^k)$
  - $f(\theta) \leq F(\theta, \theta^k)$ for all θ
- Take $\theta^{k+1} = \arg\min_{\theta} F(\theta, \theta^k)$ (minimize majorization)
- Theorem: $f(\theta^{k+1}) \leq f(\theta^k)$
- Proof:

$$f(\theta^{k+1}) \leq F(\theta^{k+1}, \theta^k) \leq F(\theta^{k+1}, \theta^k) \leq f(\theta^k)$$

Gradient Descent as a MM

- Find $\alpha \geq f''(\theta)$
- Define

$$F(\theta, \theta^k) = f(\theta^k) + \nabla f(\theta^k)(\theta - \theta^k) + \frac{\alpha}{2}\|\theta - \theta^k\|^2$$

- By Taylor's theorem, this is a majorizing function
- Gradient descent:

$$\theta^{k+1} = \arg\min_{\theta} F(\theta, \theta^k) = \theta^k - \frac{1}{\alpha}\nabla f(\theta^k)$$

Convergence

- $p(z|x,\theta) = p(x,z|\theta)/p(x|\theta)$
- $J(\theta) = \ln p(x|\theta) = \ln p(x,z|\theta) - \ln p(z|x,\theta)$
- $J(\theta) = E\left[\ln p(x,z|\theta)\,|\theta^k\right] - E\left[\ln p(z|x,\theta)\,|\theta^k\right]$
  $= Q\left(\theta,\theta^k\right) + H(\theta,\theta^k)$
- EM algorithm: $J\left(\theta^{k+1}\right) \geq J(\theta^k)$
  - Proof on board
  - Doesn't diverge
- Algorithm may get stuck in local maxima

Outline

- Clustering - Deterministic versus Probabilistic Models
  - K-means
- Mixture Distributions
- Expectation Maximization Algorithm
- Convergence of EM
- Conjugate Priors*--not covered in lecture but important
- Optimization Review-iterative methods

Conjugate Priors

- Definition: Let A, B be any two families of densities. Then, A is the conjugate prior family to B if:

$$p(\theta) \in A, p(x|\theta) \in B \Rightarrow p(\theta|x) \in A$$

  - Posterior and prior remain in the same family
- Example:
  - $\theta =$ probability of a coin toss,
  - $x =$ number of heads out of n trials
  - $p(x|\theta) = \binom{n}{x} \theta^x (1-\theta)^{n-x}$
  - Want to estimate θ from x
  - ML estimate: $\hat{\theta} = x/n$

Conjugate Prior Example Contd

- But, what if we have prior information on θ?
- Assume prior from the Beta distribution:

$$p(\theta) = \frac{1}{B(\alpha,\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}$$

  - α, β are called hyperparameters
- Then, posterior is also a Beta random variable:

$$p(\theta|x) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}\theta^{x}(1-\theta)^{n-x}$$
$$= \theta^{\alpha\prime-1}(1-\theta)^{\beta\prime-1}$$

  - $\alpha' = \alpha + x$
  - $\beta' = \beta + n - x$

Moment Matching

- The hyperparameters can be selected via moment matching
- For Beta example:
  - $E(\theta) = \frac{\alpha}{\alpha+\beta}, var(\theta) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
  - Select α and β to match $E(\theta)$ and $var(\theta)$.
  - Compute $\alpha' = \alpha + x,\ \beta' = \beta + n - x$
  - Find $E(\theta|x)$ and $var(\theta|x)$ from α' and β'.

Convergence/Iterative Methods: Overview

- Definition: A set X is convex if for any $x, y \in X$,

$$tx + (1-t)y \in X \text{ for all } t \in [0,1]$$

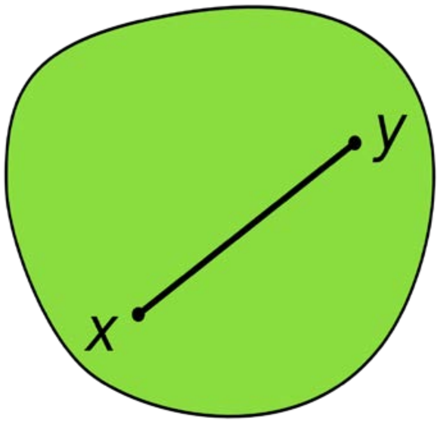
- Any line between two points remains in the set.
- Examples:
  - Square, circle, ellipse
  - $\{x \mid Ax \leq b\}$ for any matrix A and vector b
  - Not a start
- Will draw pictures on board

Outline

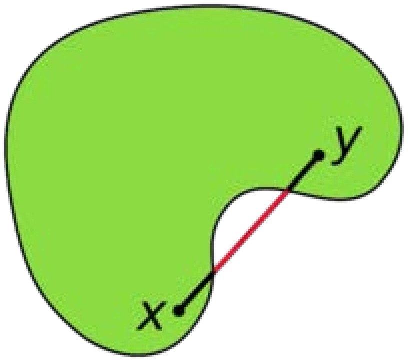
- Clustering - Deterministic versus Probabilistic Models
  - K-means
- Mixture Distributions
- Expectation Maximization Algorithm
- Convergence of EM
- Conjugate Priors*--not covered in lecture, should know
- Optimization Review-iterative methods** (Optional at this time, but it is something you should understand and may be helpful. We may cover this in the next few lectures.)

Convex Set Visualized

- Convex

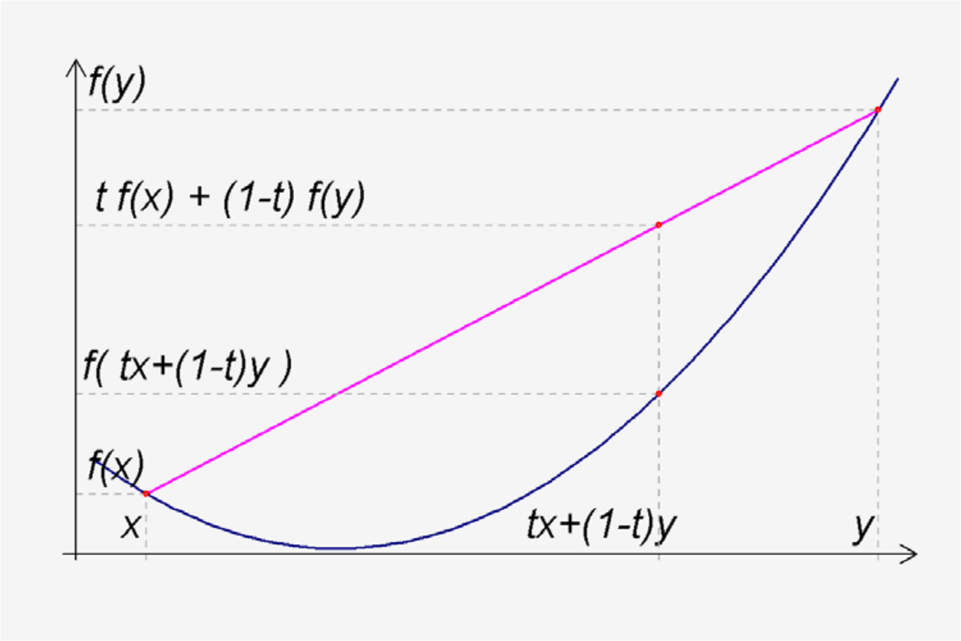


- Not convex



Convex Functions

- A real-valued function $f(x)$ is convex if:
  - Its domain is a convex set, and
  - For all x, y and $t \in [0,1]$:
    $$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$$



Convex Function Examples

- Linear function of a scalar $f(x) = ax + b$
- Linear function of a vector $f(x) = a^T x + b$
- Quadratic $f(x) = \frac{1}{2}ax^2 + bx + c$ is convex iff $a \geq 0$
- If $f''(x)$ exists everywhere, $f(x)$ is convex iff $f''(x) \geq 0$.
  - When x is a vector $f''(x) \geq 0$ means the Hessian must be positive semidefinite
- $f(x) = e^x$
- If $f(x)$ is convex, so is $f(Ax + b)$

Properties

- If $f(x)$ is convex, it is continuous
- If $f(x)$ has a derivative, then

$$f(y) \geq f(x) + \nabla f(x) \cdot (y - x)$$

Unconstrained optimization

- Problem: Given $f(x)$ find the minimum:

$$x^* = \arg\min_x f(x)$$

  - $f(x)$ is called the objective function
  - $x = (x_1, \cdots, x_p)$ is a vector of decision variables or parameters

- Called unconstrained since there are no constraints on x
- Will discuss constrained optimization briefly later

Numerical Optimization

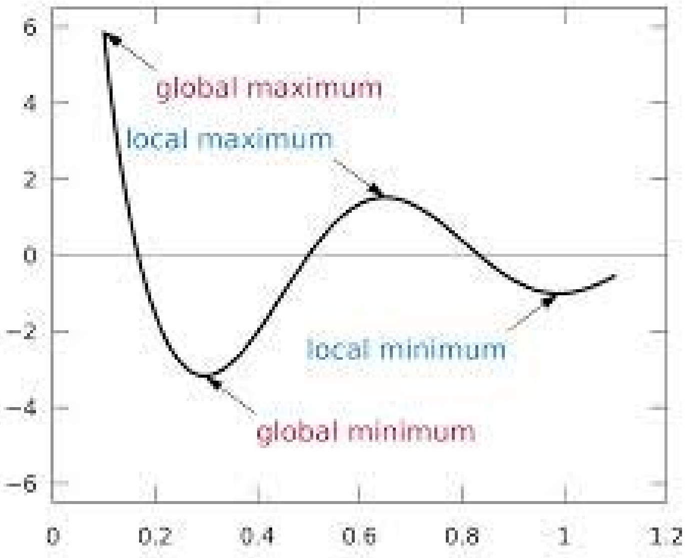
- We saw that we can find minima by setting $\nabla f(x) = 0$
  - p equations and p unknowns.
  - May not have closed-form solution

- Numerical methods: Finds a sequence of estimates x^k

$$x^k \to x^*$$

  - Or converges to some other "good" minima
  - Run on a computer program, like MATLAB

Local vs. Global Minima



- Definitions:
  - x^* is a global minima if $f(x) \geq f(x^*)$ for all x
  - x^* is a local minima if $f(x) \geq f(x^*)$ for all x in some open neighborhood of x^*
- Most numerical methods only guarantee convergence to local minima

Local Minima and Convex Function

- Theorem: If $f(x)$ is convex and x^* is a local minima, then it is a global minima
- Also, if $f(x)$ is strictly convex, then the global minima is unique

- Implication: If $f(x)$ is convex, a numerical method that converges to a local minima, will converge to a global minima.
- Many methods can find local minima
- For convex objectives, these methods will find global minima

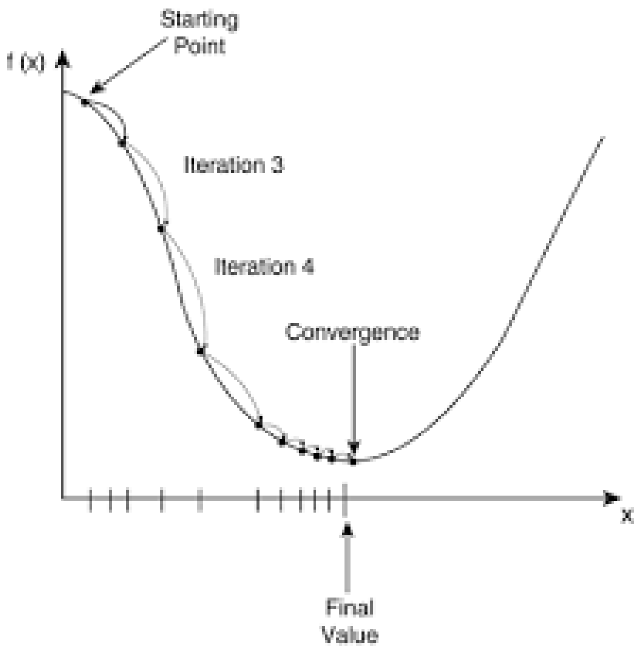
Gradient Descent

- Most simple method for unconstrained optimization
- Recall gradient:

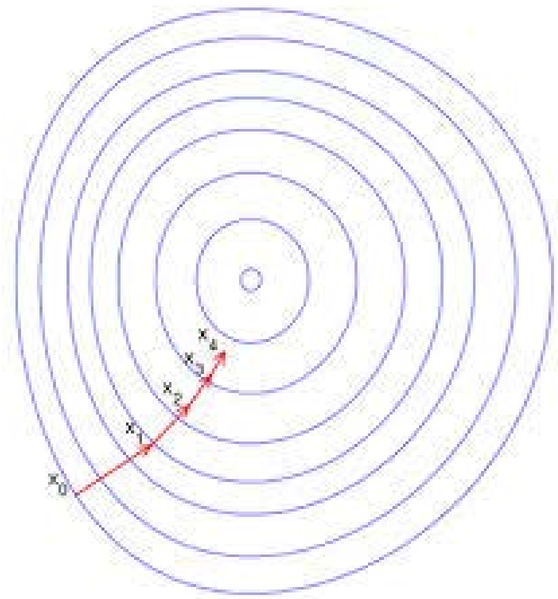
$$\nabla f(x) = \left(\partial f(x)/\partial x_1, \cdots, \partial f(x)/\partial x_p\right)^T$$

  - Column vector
- Gradient descent algorithm:
  - Start with initial x^0
  - $x^{k+1} = x^k - \alpha_k \nabla f(x^k)$
  - Repeat until some stopping criteria
- α_k is called the step size

Gradient Descent Illustrated



- $p = 1$



- $p = 2$

UCLA

Gradient Descent Analysis

- Using gradient update rule
  $$f\left(x^{k+1}\right)$$
  $$= f\left(x^k\right) + \nabla f\left(x^k\right) \cdot \left(x^{k+1} - x^k\right) + O\left\|x^{k+1} - x^k\right\|^2$$
  $$= f\left(x^k\right) \; - \alpha \, \nabla f\left(x^k\right) \cdot \left(x^{k+1} - x^k\right) + O\left(\alpha^2\right)$$
- Consequence: If step size α is small, then $f\left(x^k\right)$ decreases

- Theorem: If $f''(x)$ is bounded above, $f(x)$ is bounded below, and α is chosen sufficiently small, then gradient descent converges to local minima

UCLA

Step Size Selection

- Theorem shows we can always converge to a local minima
  - Global minima if $f(x)$ is convex
- But, step size selection is problematic
  - Need to know $f''(x)$ to find maximum step size
    - (Smaller than 1/f''(x) for all x)
  - Practical choice tends to be conservative
- Very slow step size, many steps to convergence

Adaptive Step Size Selection

- Practical algorithms change step size adaptively

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k)$$

- Tradeoff: Selecting large α_k:
  - Larger steps, faster convergence
  - But, may overshoot

Armijo Rule

- Recall that we know if $x^{k+1} = x^k - \alpha \nabla f(x^k)$

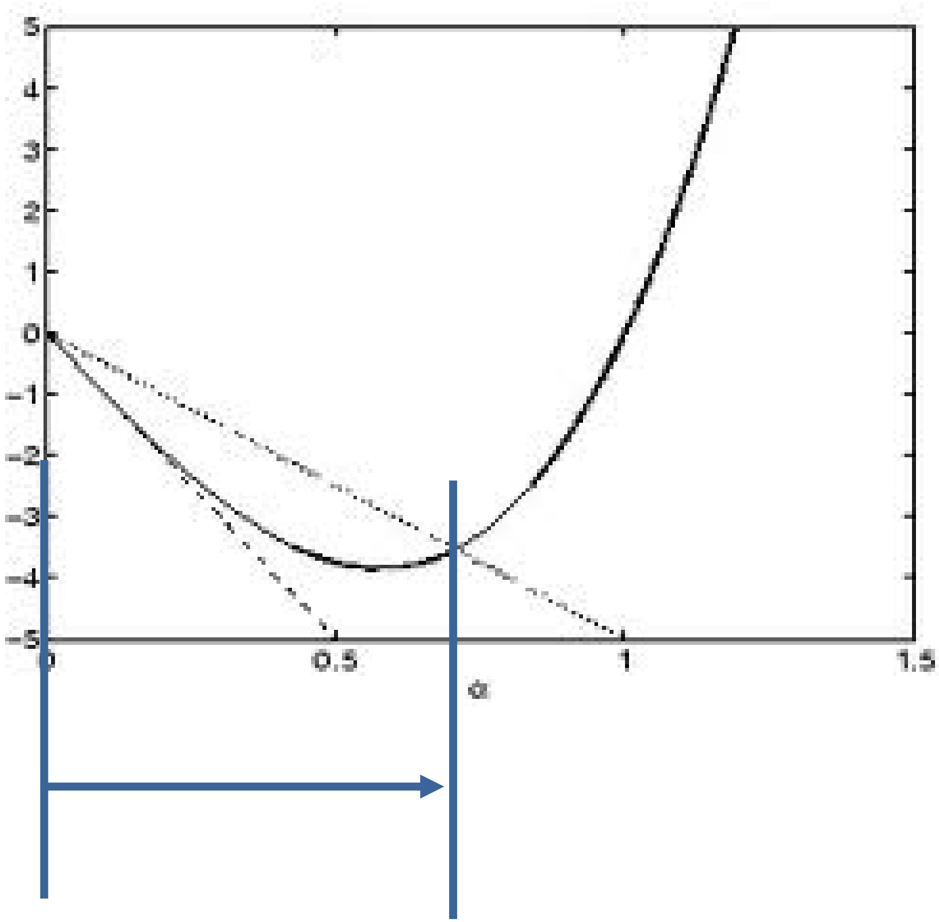
$$f(x^{k+1}) = f(x^k) - \alpha \left\|\nabla f(x^k)\right\|^2 + O(\alpha^2)$$

- Armijo Rule:
  - Select some $c \in (0,1)$. Usually $c = 1/2$
  - Select α such that

$$f(x^{k+1}) \leq f(x^k) - c\alpha \left\|\nabla f(x^k)\right\|^2$$

  - Decreases by at least at fraction c predicted by linear approx.
- Step size α selected by a line search to find largest α satisfying above conditions

Armijo Rule Illustrated



Feasible region for x^{k+1}

- Armijo rule:
  $f(x^{k+1})$
  $\leq f(x^k) - c\alpha \left\|\nabla f(x^k)\right\|^2$
- Guarantees function decrements in each iteration
- No overshoot