MIDTERM REVIEW

Will cover:

- Unit 1: Probabilities, Discrete Data Analysis
- Unit 2: Gaussians, Linear Models for Regression, and Gradient Descent
- Part of Unit 3: K means and mixture models (but not EM algo. or later)

Logistics

- Will take in-class for entire 75 min scheduled meeting on 3/11
- Can bring: one sheet (front and back) of notes
- We will provide any needed formulas (like quizlets)

Big takeaways from the course

When estimating parameters:

- Maximum likelihood is good with lots of data, problems with little data
- MAP estimates can be better (if you have a good prior)

When making predictions about new data:

- Conditioning on a single estimated parameter p(x* | \mu)
  - Vulnerable to overfitting if not careful
- Probabilistic models let us use the marginal p(x* | X)
  - Requires integral OVER our posterior beliefs about parameters
    - This integral is “easy” for some smart choices of prior and likelihood (e.g. Beta-Bern, Gauss-Gauss)
    - This integral is “hard” in general
  - Still only as good as the model is for the data at hand

Unit 1

Probabilistic Analysis Skills

- Discrete and continuous r.v.
- Sum rule and product rule
  - Bayes rule (derived from above)
- Expectations
- Independence

Distributions

- Bernoulli distribution
- Beta distribution
  - Gamma function
- Dirichlet distribution

Optimization Skills

- Finding extrema by zeros of first derivative
- Handling Constraints via Lagrange multipliers

Data analysis

- Beta-Bernoulli for binary data
  - ML estimation of ”proba. heads"
  - MAP estimation of “proba. heads"
  - Estimating the posterior
  - Predicting new data

- Dirichlet-Categorical for discrete data
  - ML estimation of unigram probas
  - MAP estimation of unigram probas
  - Estimating the posterior
  - Predicting new data

Example Questions: Unit 1

- Write the posterior predictive distribution for new coins as a Bernoulli distribution

$$p(\mu|\alpha,\beta) = \mathrm{Beta}(\alpha,\beta)$$

$$p(x|\mu) = \prod_{n=1}^{N} \mathrm{Bern}(x_n|\mu)$$

Example Questions: Unit 1

- Write the posterior predictive distribution for new coins as a Bernoulli distribution

$$p(\mu|\alpha,\beta) = \text{Beta}(\alpha,\beta)$$

$$p(x|\mu) = \prod_{n=1}^{N} \text{Bern}(x_n|\mu)$$

Solution:

$$p(x_*|x_1,\ldots x_N,\alpha,\beta) = \text{Bern}(\frac{\text{num\_heads}(x_1,\ldots x_N)+\alpha}{N+\alpha+\beta})$$

By conjugacy of the Beta-Bern model, the posterior on mu here is also a Beta with parameters (num heads + alpha ,num tails + beta).

The marginal likelihood is then a Bernoulli with parameter equal to mean of the Beta. For a derivation, see HW1's solution

Example Questions: Unit 1

$$\max_{x_1 \in \mathbb{R}, x_2 \in \mathbb{R}} \; 1 - x_1^2 - x_2^2$$

$$\text{s.t.} \quad x_1 + x_2 - 1 = 0$$

Example Questions: Unit 1

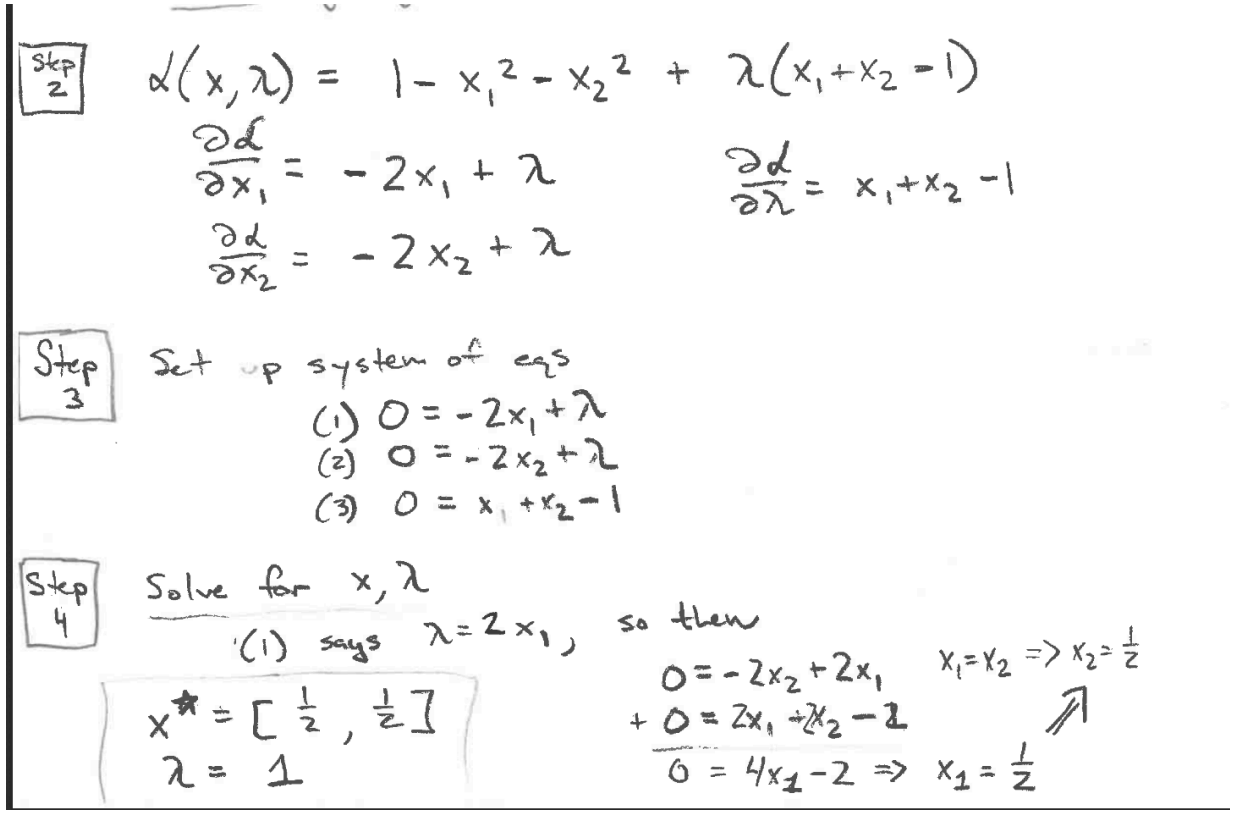
$$\max_{x_1 \in \mathbb{R}, x_2 \in \mathbb{R}} \; 1 - x_1^2 - x_2^2$$

$$\text{s.t.} \quad x_1 + x_2 - 1 = 0$$

Solution:

x1 = ½

x2 = ½



See solution on page 4 of https://www.cs.tufts.edu/comp/136/2020s/notes/day04-lagrange-multipliers.pdf

Unit 2

Probabilistic Analysis Skills

- Joints, conditionals, marginals
- Covariance matrices (pos. definite, symmetric)
- Gaussian conjugacy rules

Linear Algebra Skills

- Determinants
- Positive definite
- Invertibility

Distributions

- Univariate Gaussian distribution
- Multivariate Gaussian distribution

Optimization Skills

- Convexity and second derivatives
- Finding extrema by zeros of first derivative
- First and second order gradient descent

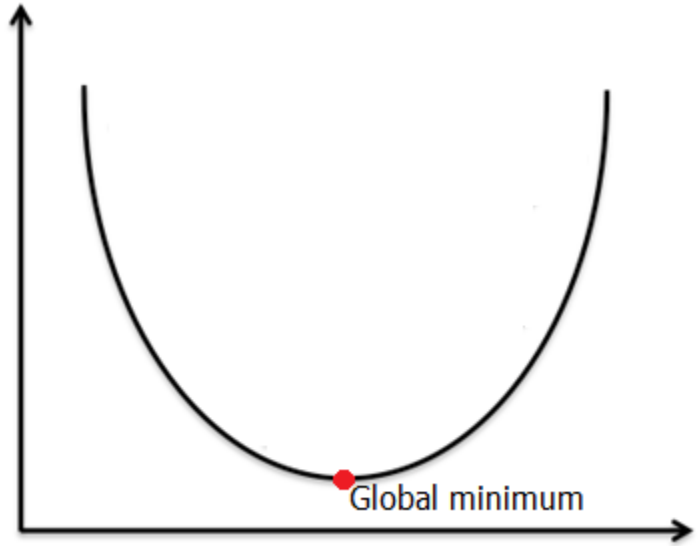
Data analysis

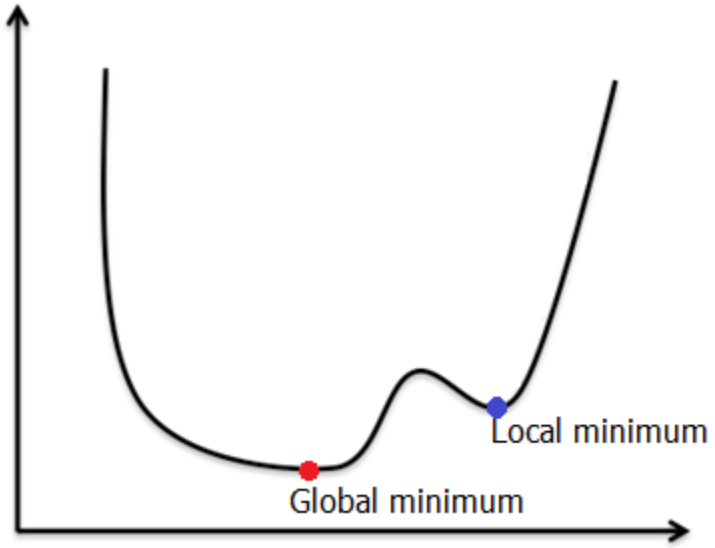
- Gaussian-Gaussian for regression
  - ML estimation of weights
  - MAP estimation of weights
  - Estimating the posterior over weights
  - Predicting new data

All equivalent statements:

- Σ is positive definite
- Σ is invertible ($\exists$ a $D \times D$ matrix Σ^{-1} s.t. $\Sigma\Sigma^{-1} = I$)
- Σ has positive determinant and all positive diagonal entries
- Σ has all positive eigenvalues ($\lambda_1 > 0$, ..., $\lambda_D > 0$)
- Σ has a unique cholesky factorization $\Sigma = LL^T$ where L is lower triangular
- inverse is positive definite

Will gradient descent always find same solution?





Second derivative must be positive everywhere

(positive definite everywhere for high dim.)

Example Problem: Gaussians

- Write p(t | w, x) for linear regression as one multivariate normal over all N observations {x_n, t_n}_{n=1}^N

Hint: Recall that we've made an iid assumption

Example Problem: Gaussians

- Write p(t | w, x) for linear regression as one multivariate normal over all N observations {x_n, t_n}_{n=1}^N

Solution:

$$p(t|x, w) = \mathcal{N}(t|\Phi w, \Sigma)$$

$$\Sigma = \begin{bmatrix} \beta^{-1} & 0 & \dots & 0 \\ 0 & \beta^{-1} & \dots & 0 \\ & & \ddots & \\ 0 & 0 & & 0 \\ 0 & 0 & \dots & \beta^{-1} \end{bmatrix}$$

Example Problem: Data Analysis

- In CP2, we assume that

$$p(w) = \mathcal{N}(w|0, \alpha^{-1} I_M)$$

Is this a good choice for all problems? For example, if we are predicting housing prices in Medford area (where average home value t_n is something like $500,000)?

Example Problem: Data Analysis

- In CP2, we assume that

$$p(w) = \mathcal{N}(w|0, \alpha^{-1} I_M)$$

Is this a good choice for all problems? For example, if we are predicting housing prices in Medford area (where average home value t_n is something like $500,000)?

Solution: if we have a prior bias towards w coefficients being close to zero, it will penalize the large weight needed on the "bias/intercept" feature needed to make good predictions when t is on average large.
Practical remedy is to either preprocess the data so that "t" has zero mean on the training set, or change the prior (use known good mean on w_bias)

Unit 3: (Only limited coverage since ongoing)

Distributions

- Mixture models
- Mixtures of Gaussians (GMMs)

Optimization Skills

- K-means objective and algorithm
- Coordinate ascent / descent algorithms
- Writing optimization objectives with and without assignment variables

Data analysis

- K-means on a dataset
  - How to pick K via cross validation
- Gaussian mixtures
  - Overall motivation only
  - No need to understand any algorithms

Example problem from Unit 3

True or False: K-means algorithm always converges to a global minimum.

Example problem from Unit 3

True or False: K-means algorithm always converges to a global minimum.

Solution: FALSE

K-means has local optima. Think of the rectangle example we discussed in class (4 data points, arranged on a rectangle with one side much longer than the other. Multiple solutions exist for this example with different costs.