CSCI 381
Review sheet for the midterm

The questions on the midterm will be primarily factual, like the quizzes. Typical question format will be fill-in-the-blank; for example, you may be given a paragraph with blanks in it, or you may be asked to identify the parts of a formula or algorithm. Possible answers for the blanks can be chosen from a word bank, though there might be a single word bank for the entire test rather than separate words banks for each problem/section.

There also may be a wide variety of problems. I reserve the right to to include a question that asks you to jot down some code, derive a formula, analyze complexity, prove something, etc.

You will not be asked to produce formulas from scratch, but you should be familiar enough with the main formulas to be able to recognize them and identify the significance of their parts. You should also be able to determine whether something is an array, vector, or scalar and, from context, determine dimensions.

Prolegomena/basic terms and concepts

- Training/learning/fitting
- Model
- Data
- Observations/data points
- Targets
- Supervised vs unsupervised
- Classification
- Regression
- Density estimation
- Training set, test set
- Generalization, overfitting
- Parameters, model family
- Hyperparameters
- Error, loss function

The nature of data and KNN

- Variable/feature/attribute
- D for dimensionality of the data set
- N for number of observations
- $\mathbf{X}$, $\boldsymbol{y}$, conventional use of i and n
- The curse of dimensionality
- The premise of KNN
- Algorithm outline for KNN
- Norms
- Hyperparameters for KNN
- Analysis/cost of KNN
- Extreme cases ($k = 1$, $k = N$)
- What it means for KNN to be a *non-parametric* technique

Linear regression

- Progression from the simplest version (a line, $y(x) = \theta_0 + \theta_1 x$) through multidimensional versions (multiple regression), feature selection, and basis functions to $y(\mathbf{x}) = \sum_{j=1}^{k} \theta_j \phi_j(\mathbf{x})$.
- What is "linear" about linear regression.
- The loss functions for the various versions of linear regression.
- Regularization (meaning, purpose; kinds: ridge, LASSO).
- What a closed form solution is, and which versions of linear regression have one.
- The purpose of gradient descent and why you would use it.
- The general outline of gradient descent.

Logistic regression

- The premise of adapting linear regression for classification.
- The logistic function, its use and useful properties.
- The loss function (mean log loss) and its gradient (don't need to memorize, just recognize them, be able to identify their parts and significance).
- Logistic regression as a density estimation; how to adapt it to binary classification.
- How to adapt binary classification to multiclass classification using one-vs-rest.

Gaussian mixture models and expectation-maximization

- What a *Gaussian* is. What a *mixture model* is.
- The parameters in a Gaussian mixture model.
- Latent variables (in general: features of the data that are missing or hidden; in the context of GMM: which underlying process each observation came from).
- Log likelihood, and why we take the log instead of just computing the product.
- What the parts of the GMM formula mean.
- Expectation maximization:
  - General structure of the algorithm.
  - Termination condition, based on average log likelihood.
  - E-step: *responsibilities* computed as each component model's contribution to the model's computed probability density for each data point and interpreted as the expected value of the latent variables.
  - M-step: using responsibilities as proxies for the latent variables to (re-)compute the parameters for the model.

Support vector machines

- Hyperplanes; margin.
- What support vectors are; how they "support" classification.
- The formula for classifying data points using a hyperplane: $\text{sign}(\boldsymbol{w}^T\boldsymbol{x} + b)$.
- Dealing with non-linearly separable data.
  - Soft margin (as opposed to hard margin); slack variables.
  - Kernels; the kernel trick.
- Constrained optimization.
  - What constrained optimization is; objective, constraints.
  - What Lagrangian multipliers and the Lagrangian function are (a means by which we unify the objective and constraints into a single formula).
  - The dual problem (a new constrained optimization problem that has the same solution as the original, which, under some circumstances, is easier or less computationally expensive to solve).
  - What linear programming and quadratic programming are.
- SVMs as a quadratic programming problem.
  - Each data point as a constraint and, hence, having a corresponding Lagrangian multiplier.
  - The idea of using an off-the-shelf QP solver.
  - Non-zero Lagrangian multipliers indicating support vectors.
  - Computing $\boldsymbol{w}$ and b.