The second midterm covers in-class material days 14-24, labs 6-8, reading weeks 6-12, and material that carries over from the first midterm (see sections 1 and 2 from study guide 1 + **implementation**). It is in class and closed notes/books/internet/other, but you may use a 1 page (front and back), hand-written "resource sheet" (created by *you*). You will not need a calculator. I have put vocab in blue.

1. From before: softmax, cross-entropy and other loss functions, gradient descent, confusion matrices
2. Sources of Error in and ML pipeline
   - Traditional measures of error focus on prediction vs. label
   - What other sources of error are there throughout the pipeline?
   - AI Bill of Rights (see materials from Day 14)
3. Perceptron and Support Vector Machines
   - Idea and equation of a separating hyperplane (weight vector points toward the + side)
   - Perceptron algorithm and derivation of the weight updates; perceptron cost function
   - Perceptron weight updates: geometric interpretation and gradient descent interpretation
   - Guarantees and limitations of the perceptron algorithm
   - Support Vector Machines (SVMs) can find the maximum margin hyperplane
   - What are support vectors? What is the geometric (γ) vs. the functional ($\hat{\gamma}$) margin?
   - How we used the functional and geometric margins to cast SVMs as an optimization problem
   - Motivation and method of Lagrange multipliers, application to SVMs
   - High-level steps of transforming the SVM Lagrangian into a problem involving only α values
   - What do these α values represent and how can we use them to find $\vec{w}^*$?
   - Reformulation of SVMs as maximizing $W(\vec{\alpha})$ uses only inner products between examples
   - Idea of a kernel function and how it can replace the dot product (not Gaussian kernel details)
4. Neural Networks: fully-connected and CNN
   - What is a Neural Network (NN)? Motivation and goals when using them
   - High level idea of training using gradient descent on the loss function
   - Fully Connected architectures, dimensionality analysis, parameters vs. hyperparameters
   - Choice of activation function, pros and cons of sigmoid, tanh, and ReLU
   - Softmax function as the activation function for the last layer, cross-entropy loss after that
   - Training: how to initialize the weights/biases, what is the point of mini-batches?
   - Motivation behind Convolutional Neural Networks (CNNs); application to images
   - CNN architectures: idea of 3D volumes, typical steps CONV, RELU, POOL, FC
   - CONV layer details: filters computing cross-correlations, slide filter over width and height
   - Dimensionality analysis (shapes of filter weights/biases, shapes of input/output)
   - Backpropagation: both high-level purpose and mathematical details
   - Skip: dropout, regularization, any pooling besides 2×2 with stride 2

5. Neural Networks: RNNS, Transformers, and GANs
    - Idea of recurrent neural networks (RNNs), purpose, input/output format, flexibility
    - RNN variations: LSTMs, GRUs (ability to remember relevant information)
    - Details of how we used RNNs for text generation in Lab 8, including data preprocessing
    - Risks of large language models (LLMs)
    - Text generation from vocab probabilities, idea of temperature to control randomness
    - Math and intuition behind attention mechanisms
    - Big-picture ideas of transfomers and how they use attention mechanisms
    - Transformers are permutation-invariant but we can add positional encodings
    - Generative Adversarial Networks (GANs) as another way to generate synthetic data
    - GANs are comprised of two networks: generator and discriminator
    - GAN loss functions and relationship to binary cross-entropy loss; training difficulties
    - For image creation, GAN generator is usually a CNN with transposed convolutions
6. Interpretability
    - Global vs. local interpretability
    - LIME interpretability method
    - Saliency maps
    - Model-of-the-model approaches
7. Unsupervised Learning
    - Basics of K-means and Gaussian mixture models (GMMs)
    - Autoencoders and variational autoencoders (VAEs)
    - Connection between unsupervised learning and generative methods