Conditional random fields (CRFs) may be used to learn models of how structured labels $x$ relate to observed data $y$. We consider the problem of visual object recognition and aim to predict a binary vector $x$, where $x_s = 1$ if an instance of object category $s$ is present in the image, and $x_s = 0$ if category $s$ does not appear. Our observation vectors $y \in \mathbb{R}^K$ are so-called “gist descriptors”, histograms of image gradients which have been engineered to coarsely capture the semantic content of images. For the purposes of this assignment, you may treat the feature representation of each image as just a vector of $K = 512$ numbers.

The questions below use datasets of images annotated with the presence/absence of object categories such as buildings, cars, and trees. Our goal is to model contextual (co-occurrence) relationships between object categories, and the relationships between object occurrence and image appearance. We consider a 13-category dataset, and a larger 23-category dataset.

Recall that a pairwise MRF of $N$ binary variables $x_s \in \{0, 1\}$ is defined by parameters $\theta_s$ for each node $s \in V$, and $\theta_{st}$ for each edge $(s, t) \in E$. The joint distribution is then:

$$p(x \mid \theta) = \exp \left\{ \sum_{s \in V} \theta_s x_s + \sum_{(s, t) \in E} \theta_{st} x_s x_t - \Phi(\theta) \right\},$$

$$\Phi(\theta) = \log \left( \sum_{x} \exp \left\{ \sum_{s \in V} \theta_s x_s + \sum_{(s, t) \in E} \theta_{st} x_s x_t \right\} \right).$$

A CRF instead models $p(x \mid y, \theta, \gamma)$, where $\gamma$ are additional parameters relating $x$ to the observed data $y$ (e.g., image features). The joint distribution is then:

$$p(x \mid y, \theta, \gamma) = \exp \left\{ \sum_{s \in V} \theta_s x_s + \sum_{(s, t) \in E} \theta_{st} x_s x_t + \sum_{s \in V} \sum_{k=1}^{K} \gamma_{sk} x_s y_k - \Phi(y, \theta, \gamma) \right\},$$

$$\Phi(y, \theta, \gamma) = \log \left( \sum_{x} \exp \left\{ \sum_{s \in V} \theta_s x_s + \sum_{(s, t) \in E} \theta_{st} x_s x_t + \sum_{s \in V} \sum_{k=1}^{K} \gamma_{sk} x_s y_k \right\} \right).$$

There is one parameter $\gamma_{sk}$ to capture the relationship between each image feature $y_k$ and object category $x_s$. Note that unlike the MRF, the CRF partition function $\Phi(y, \theta, \gamma)$ depends on the image $y$, and thus must be computed separately for each training image.

In the following questions, we compare four different model families:
CRF  The full distribution \( p(x \mid y, \theta, \gamma) \) of Eq. (3), allowing all parameters to be non-zero.

MRF  The distribution \( p(x \mid \theta) \) of Eq. (1), which can be seen as a CRF in which \( \gamma_{sk} = 0 \) so that there is no dependence on the observed image features \( y \).

Logistic Regression  Independent logistic regression models which predict each category \( x_s \) based on image features \( y \). This is a special case of the CRF where \( \theta_{st} = 0 \).

Independent  A model of object frequencies ignoring image features \( y \) and co-occurrence relationships. This is a special case of the CRF where \( \theta_{st} = 0, \gamma_{sk} = 0 \).

Suppose we are given fixed CRF parameters, image features \( y \) for some test example, and binary labels \( x_s \) for some subset of the object categories. We can then compute the marginal posterior probability of each unspecified label by enumerating all possible label completions, computing Eq. (3) for each one, and summing over the set of unspecified labelings. We have provided inference methods for CRF (crfInfer.m) and MRF (mrfInfer.m) models.

Question 1

a) Consider the CRF model defined above, and a dataset of \( L \) training examples \( (x^{(l)}, y^{(l)}) \). Determine equations for the derivative of the log-likelihood

\[
\mathcal{L}(\theta, \gamma) = \sum_{l=1}^{L} \log p(x^{(l)} \mid y^{(l)}, \theta, \gamma)
\]

with respect to each parameter \( \theta_s, \theta_{st}, \) and \( \gamma_{sk} \). Simplify your answer.

b) Implement a gradient-based CRF learning algorithm using the results from part (a). We have provided a skeleton of a function getLlikCRF.m that computes the training log-likelihood and gradient with respect to the current model parameters \( \theta, \gamma \). Complete the function! Hint: Look at the getLlikMRF.m function that we have provided, which computes the objective function and gradients for MRF learning. Your CRF code should be similar, but should also include terms related to the image features \( y \).

c) Test your CRF code on the 13-category dataset by running the appropriate section of main.m. Compare the performance of the CRF, in terms of area under the ROC curve (AUC), to the three baseline methods described above: independent models of object frequencies, independent logistic regression models for each class, and an MRF which ignores image features. The template code provides solutions for all of these baseline methods, using provided training and test datasets. As in Homework 2, we use \( L_1 \) regularization with parameter \( \lambda = 0.01 \) for the MRF and CRF edge potentials. Plot the AUC of these methods as a function of the number of labels provided, as well as the training log-likelihood for the CRF, MRF, and logistic regression models after each optimization iteration.

d) Discuss the difference of performance in terms of AUC for the different methods. Which method works best in the regime where only a few labels are given? Which method works best when many labels are given?
Question 2

In this question, we explore mean-field approximations for learning MRF and CRF models. Rather than enumerating all possible states to compute the gradients needed for optimal learning, we use mean field methods to approximate the needed expectations. We use a naive mean field based on a fully factorized approximation of the true posterior \( q(x) = \prod_{s \in V} q_s(x_s) \), \( q_s(x_s) = \mu_s^{x_s}(1 - \mu_s)^{1-x_s} \). Mean field inference seeks to minimize \( D(q||p) \), the KL divergence between the variational distribution \( q(x) \) and some true MRF (Eq. (1)) or CRF (Eq. (3)).

a) For the binary MRF of Eq. (1), mean field inference seeks to maximize the following variational objective function:

\[
F(\mu, \theta) = \mathbb{E}_q[\log p(x) - \log q(x)] + \Phi(\theta) \\
= \mathbb{E}_q \left[ \sum_{(s,t) \in \mathcal{E}} \theta_{st} x_s x_t + \sum_{s \in \mathcal{V}} (\theta_s x_s - x_s \log \mu_s - (1 - x_s) \log (1 - \mu_s)) \right] \\
= \sum_{(s,t) \in \mathcal{E}} \theta_{st} \mu_s \mu_t + \sum_{s \in \mathcal{V}} (\theta_s \mu_s - \mu_s \log \mu_s - (1 - \mu_s) \log (1 - \mu_s)).
\]

(5)

Derive a closed form expression for the optimum value of the variational mean \( \mu_s \) of some node \( s \), given fixed variational means \( \mu_t, t \neq s \), for all other nodes.

b) Mean field variational inference can be directly generalized to the binary CRF of Eq. (3):

\[
F(\mu, \theta, \gamma \mid y) = \mathbb{E}_q[\log p(x \mid y, \theta, \gamma) - \log q(x)] + \Phi(y, \theta, \gamma) \\
= \sum_{(s,t) \in \mathcal{E}} \theta_{st} \mu_s \mu_t + \sum_{s \in \mathcal{V}} (\theta_s \mu_s + \sum_{k} \gamma_{sk} \mu_s y_k - \mu_s \log \mu_s - (1 - \mu_s) \log (1 - \mu_s)).
\]

Extend your derivation from part (a) to determine a closed form expression for the variational mean \( \mu_s \) of some node \( s \), given fixed variational means \( \mu_t, t \neq s \).

c) Implement a mean field variational inference algorithm for binary MRFs and CRFs, which sequentially updates the variational parameters for single nodes using the expressions derived in parts (a,b). This leads to a coordinate ascent algorithm for finding a (possibly local) maximum of Eq. (5,6). We have provided skeleton code in `margProbMean.m`.

d) The optimum value \( \hat{\mu} \) of the mean field variational objective in Eq. (5) approximates the log-normalization constant, \( F(\hat{\mu}, \theta) \approx \Phi(\theta) \). We can thus use mean field inference for approximate MRF parameter learning by maximizing the following objective:

\[
\mathcal{L}(\theta) = \sum_{\ell=1}^{L} \left[ \sum_{s \in \mathcal{V}} \theta_s x_s^{(\ell)} + \sum_{(s,t) \in \mathcal{E}} \theta_{st} x_s^{(\ell)} x_t^{(\ell)} - F(\hat{\mu}, \theta) \right], \quad \hat{\mu} = \arg\max_{\mu} F(\mu, \theta).
\]

Derive an expression for the gradient of \( \mathcal{L}(\theta) \) with respect to the model parameters \( \theta \), and extend the skeleton code in `getLlikMRFMean.m` to compute this objective and its gradient.
e) Using the variational objective in Eq. (6), we can similarly learn approximately optimal CRF parameters by maximizing the following objective:

\[
\mathcal{L}(\theta, \gamma) = \sum_{\ell=1}^{L} \left[ \sum_{(s,t) \in E} \theta_{st} x_{st}^{(\ell)} x_{st}^{(\ell)} + \sum_{s \in V} (\theta_{s} x_{s}^{(\ell)} + \sum_{k} \gamma_{sk} x_{sk}^{(\ell)} y_{k}^{(\ell)}) - F(\hat{\mu}^{(\ell)}, \theta, \gamma | y^{(\ell)}) \right],
\]

\[
\hat{\mu}^{(\ell)} = \arg \max_{\mu} F(\mu, \theta, \gamma | y^{(\ell)}).
\]

Note that because the CRF distribution depends on the image features \( y^{(\ell)} \), there are distinct variational parameters \( \hat{\mu}^{(\ell)} \) for each training image \( \ell \). Derive an expression for the gradient of \( \mathcal{L}(\theta, \gamma) \) with respect to the model parameters \( \theta, \gamma \), and extend the skeleton code in \texttt{getLlikCRFMean.m} to compute this objective and its gradient.

Suppose we are given fixed CRF parameters, image features \( y \) for some test example, and binary labels \( x_s \) for some subset of the object categories. We can then use mean field variational inference to approximately compute the posterior probability of each unspecified label. We have provided this inference code for CRF (\texttt{crfMeanInfer.m}) and MRF (\texttt{mrfMeanInfer.m}) models, building on your implementation from part (c).

f) Test your mean field MRF and CRF learning algorithms on the 13-category dataset by running the appropriate section of \texttt{main.m}. Again use \( L_1 \) regularization with parameter \( \lambda = 0.01 \) for the MRF and CRF edge potentials. Compare the performance of the mean-field MRF and CRF models to the four models from question 1. Plot the AUC of all methods as a function of the number of labels provided, as well as the mean field CRF and MRF learning objectives after each optimization iteration.

g) Test your mean field MRF and CRF learning algorithms on the larger 23-category dataset by running the appropriate section of \texttt{main.m}. Discuss the relative performance of the mean-field CRF model, the mean-field MRF model, independent logistic regression models of each class, and independent models of category frequencies. Plot the AUC of all methods as a function of the number of labels provided.

h) We derived a mean field variational inference method because when the number of object categories is large, computing exact gradients for CRF and MRF training becomes intractable. If it takes \( T \) seconds to train an MRF on a 13-category dataset, roughly how long (in terms of \( T \)) would it take to train an MRF on a 23-category dataset? Estimate how long it would take to run the MRF code on the 23-category dataset on your computer.