**TTIC 31020: Intro to ML** **Winter 2024**

# Homework 5

Due: 6 PM, 6 Feb 2024

**Note** *You may discuss these problems in groups. However, you must write up your own solutions and mention the names of the people in your group. Also, please do mention any books, papers or other sources you refer to. We recommend that you typeset your solutions in LaTeX. Please submit your solutions as a PDF document on Canvas.*

**Challenge and Optional Questions** *Challenge questions are marked with* challenge *and Optional questions are marked with* optional *. You can get extra credit for solving* challenge *questions. You are not required to turn in* optional *questions, but we encourage you to solve them for your understanding. Course staff will help with* optional *questions but will prioritize queries on non-*optional *questions.*

---

## Gaussian Mixtures

1. In this question we will consider the family of generative distributions over $X \in \mathbb{R}^d, Y \in \{\pm 1\}$, where $X|Y$ is Gaussian with diagonal covariance, i.e. $X|Y = y \sim \mathbb{N}(\mu_y, \Sigma_Y)$ where $\Sigma_Y$ is a non-negative diagonal matrix. The family is parametrized by $p_+ = P(Y = +1), \mu_-, \mu_+ \in \mathbb{R}^d$ and $\mathrm{diag}(\Sigma_-), \mathrm{diag}(\Sigma_+) \in \mathbb{R}^d$.

   (a) **Parameter Estimation** Compute the maximum likelihood estimate for parameters $p_+, \mu_-, \mu_+, \Sigma_-, \Sigma_+$ based on data $(x_1, y_1), \ldots, (x_m, y_m)$ sampled i.i.d. from the mixture.

   (b) **Prediction** Compute the posterior $\eta(x) = P(Y = 1|x)$, the discriminant $r(x)$ such that $P(Y = 1|x) = \frac{1}{1+e^{-r(x)}}$, and the Bayes predictor, in terms of the parameters.

   (c) **As a Linear Predictor** Write down a feature map $\phi : \mathbb{R}^d \to \mathbb{R}^D$, for the smallest $D$ you can, such that every discriminant for a distribution in this class can be expressed as $r(x) = \langle w, \phi(x) \rangle$ for some $w \in \mathbb{R}^D$. Write down $w$ in terms of the parameters.

   (d) Show that every linear predictor $x \mapsto \langle w, \phi(x) \rangle$ can be obtained as a discriminant for a mixture of this form. That is, show that for every $w \in \mathbb{R}^D$, there exists $p_+, \mu_-, \mu_+$ and diagonal $\Sigma_-, \Sigma_+$, such that the discriminant for the mixture specified by these parameters is precisely $r(x) = \langle w, \phi(x) \rangle$. For every $w \in \mathbb{R}^D$, write down such parameters $p_+, \mu_-, \mu_+, \Sigma_-, \Sigma_+$ explicitly (as a function of $w$).

   (e) How do the Bayes optimal decision boundaries for this class look like geometrically?

## Modeling Text Documents

2. **A Simple Model**

   In this question we will discuss and study a generative model for text documents $X$ and a label $Y$, which we will think of as corresponding to the topic of the document. Our generative model will be for documents with a fixed number of words $N$ (think of, e.g., documents with $N = 100$ words), i.e. $x \in \mathcal{X}$, where $\mathcal{X} = \text{dictionary}^{100}$ is the set of all 100-word documents. The model is parametrized by a distribution $p_{\text{topic}}$ over topics, and for each topic $y$, a distribution $p_y$ over words. For now, think of having just two topics, $y \in \{\text{quilting}, \text{knitting}\}$, or perhaps $y \in \{\text{positive}, \text{negative}\}$, and so $p_{\text{topic}}$ is just a single probability, and additionally we have two word distributions $p_{\text{quilting}}$

(giving higher probability to words such as "batting", "seam" and "patch") and $p_{\text{knitting}}$ (giving high probability to words such as "yarn" and "purl" ). Each word distribution can be thought of as a vector of probabilities (summing to one) with length corresponding to the number of words in the dictionary. The generative processes is as follows: topic $y$ is chosen from a distribution $p_{\text{topic}}$. Then, conditioned on $y$, each word $x[1], \ldots, x[N]$ in the document is chosen independently (given $y$) from the distribution $p_y$.

(a) Compute the maximum likelihood estimate for parameters $p_{\text{topic}}, \{p_y\}$ based on data $(x_1, y_1), \ldots, (x_m, y_m)$ sampled i.i.d. from the model. It might be useful for you to define some summary statistics of the data.

(b) Assuming only two classes, compute the discriminant $r(x)$ such that $P(Y = 1|x) = \frac{1}{1+e^{-r(x)}}$,in terms of the parameters.

(c) Write down a feature map $\phi : \mathcal{X} \to \mathbb{R}^D$, for the smallest $D$ you can, such that every discriminant for a distribution in this class can be expressed as $r(x) = \langle w, \phi(x) \rangle$ for some $w \in \mathbb{R}^D$. Write down $w$ in terms of the parameters. The dimension $D$ might depend on the size of the "dictionary" (the number of possible words), but the number of non-zero coefficients in $w$, as well as the value of these coefficients, should not depend on the number of size of the dictionary.

(d) Write down $w$ that corresponds to the maximum likelihood parameters, in term of the summary statistics of the training set. Try to obtain a simple expression.

3. **Adding A Prior**

As we saw in the previous part, there are so many possible words, i.e., the size of our dictionary, it is hard to expect the maximum likelihood estimate will be reliable, especially for rare words. As an extreme, think of the following Do not submit : what is the weight associated with a word that appeared only once, in one of the topics? What is the weight associated with a word that didn't appear at all? We can alleviate the problem by assuming a prior on the topic distributions $p_{\text{topic}}$, word distributions $p_y$ that will bias us towards a uniform distribution over all words, and ensure our estimate for all word probabilities $p_y(s)$ will always be positive. Specifically, we will use symmetric Dirichlet prior[1]. The symmetric Dirichlet $Dir(\alpha)$, with parameter $\alpha \in \mathbb{R}, \alpha > 0$ is a prior over a discrete probability distribution $p = (p_1, P_2, \ldots, p_n)$ (or in other words, over non-negative vectors that sum to one) with probability density function:

$$f_{Dir(\alpha)} = \frac{1}{Z(\alpha)} \prod_{i=1}^{n} p_i^{\alpha-1} \tag{1}$$

where $Z(\alpha)$ is a normalizing constant, equal to $Z(\alpha) = \Gamma(\alpha)^n / \Gamma(n\alpha)$, but the expression for $Z(\alpha)$ will not be important and we can just leave it as $Z(\alpha)$. Note that this is a distribution over distributions (a meta-distribution)—in (1), $p = (p_1, \ldots, p_n)$ is the random variable (think of it as just a random vector) the (1) specifies the p.d.f. of the Dirichlet distribution, i.e. the density of this meta distribution at every possible setting of the random $p$.

As a prior over our parameters, we will take $p_{\text{topic}} \sim Dir(1)$ (i.e. Dirichlet with parameter 1) and $p_y \sim Dir(\alpha)$ independently for each topic $y$, where $\alpha$ is a meta-parameter we will have to choose, e.g. $\alpha = 10$ (a meta-parameter is a parameter of the prior, and is just fixed a-priori, not fit to the data, similar to a regularization tradeoff parameter $\lambda$).

(a) Write down the MAP estimate for the parameters $p_{\text{topic}}, \{p_y\}$.

(b) Write down the linear predictor $w$ (in the feature space you defined above) that corresponds to the MAP parameters. Avoid as much as possible a dependence on the number of words in the dictionary. In particular, you should be able to have all the coefficients of $w$ except for the "bias

[1]The mean of the symmetric Dirichlet is a uniform distribution, i.e. we do not a-priori favor any particular word. If we had a prior distribution over words in mind, we would use the general, non-symmetric, Dirichlet. See, e.g., the Wikipedia page for the "Dirichlet Distribution"

term" (i.e. the coefficient $w[0]$ corresponding to a constant feature $\phi(x)[0] = 1$) not depend on the size of the dictionary, with only $w[0]$ depending on the size. optional Show that as the size of the dictionary goes to infinity, the "bias term" converges to a sensible limit. That is, if we take the number of possible words to be infinite, or perhaps very large, we can describe the discriminant as a linear predictor with coefficients that do not depend on the size of the dictionary.

(c) optional (Beyond scope of course) A point estimate, especially the MAP estimate, is problematic. Instead, we should consider the posterior $P\left(p_{\text{topic}}, \{p_y\} \middle| \{(x_i, y_i)\}_{i=1..m}\right)$. Derive this poesterior. Hint: the (non symmetric) Dirichlet distribution is the conjugate prior for the multinomial, and so the posterior for $p_{\text{topic}}$ and each $p_y$ will be Dirichlet, optional challenge Drive the posterior of the predictor $P\left(w \middle| \{(x_i, y_i)\}_{i=1..m}\right)$ and predictions $P\left(y_{\text{query}} \middle| x_{\text{query}} \{(x_i, y_i)\}_{i=1..m}\right)$

The feature vector $\phi(x)$ and weight vector $w$ have dimensionality that depends on the number possible words. This might seem problematic, as we will need to know in advance all possible words. But check: using weights corresponding to the MAP estimate, what is the weight for words that did not appear in the training set? This means we can just ignore such words, and even when predicting, only consider words that actually appeared in the training set.

4. **Multiple Classes** In this question, we make our problem more general. We turn to having more than two topics, labels or classes. Just like before, the model is parametrized by a distribution $p_{\text{topic}}$ over topics, and for each topic $y$, a distribution $p_y$ over words. Answer the following questions for $y \in \mathcal{Y} = \{1, 2, \ldots, k\}$ for some finite $k > 2$.

   (a) Write down the posterior $P(Y = y|x)$.

   (b) Show how to express the posterior as

   $$P(Y = y|x) = \frac{\exp(r_y(x))}{\sum_{y' \in \mathcal{Y}} \exp(r_{y'}(x))}, \tag{2}$$

   where for each $y$, $r_y(x) = \langle w_y, \phi(x) \rangle$ for some vectors $w_y \in \mathbb{R}^D$ that depend on the parameters. Write $w_y$ in terms of the parameters.

   Note (Do not submit): The form (2) is sometimes referred to as *softmax*, since if we use $r_y(x) = \beta \langle w_y, \phi(x) \rangle$ with an "inverse temperture" parameter $\beta$, and take $\beta \to \infty$, we would get that $y|x$ is concentrated at $\arg\max_y r_y(x)$.

   (c) Write down $w_y$ corresponding to the MAP parameters, in terms of the summary statistics of the training set and the meta-parameter $\alpha$, avoiding a dependence on the total number of possible words (except for the "bias term", i.e. a coefficient corresponding to a constant feature, as described above), and ensuring the weight corresponding to words that did not appear in the training set is zero.

   (d) We can now also consider discriminative learning for $\{w_y\}$. Write down the negative log conditional likelihood $-\log P\left(\{y_i\} \middle| \{x_i\}, \{w_y\}\right)$. It might be convenient to refer to $r_y(x) = \langle w_y, \phi(x) \rangle$. (Note: here we are back to NOT using a prior, neither over the model parameters, nor over $w_y$).

   (e) Express the negative log conditional likelihood as a sum of loss function

   $$-\log P\left(\{y_i\} | \{x_i\}, w\right) = \sum_i \ell(y_i; (r_1(x), \ldots, r_k(x))).$$

   Write down the form of the loss function $\ell(y_i; (r_1(x), \ldots, r_k(x)))$.

   This loss is the multi-class logistic loss (recently also sometimes referred to as the "cross entroty" loss). Note that this loss function does not depend on the specifics of the the model, i.e. the form of $r(x)$ or what function class $r(\cdot)$ belongs to, and only on using a "softmax" conditional label distribution of the form (2).

5. **Adding Dependencies: A Markov Model** We now turn to a more complex model, with the following generative process: we first select a topic $y \sim p_{\text{topic}}$ as before. We select the first word in the document, $w[1]$ according to a topic-specific distribution $w[1] \sim p_{y,\text{init}}$. Each subsequent word $w[i]$ is then

selected based on the topic $y$ and the preceding word $w[i-1]$, according to a conditional distribution $p_{y,\text{tran}}(w[i]|w[i-1])$ (that does *not* depend on the position $i$), but, conditioned on $y$ and $w[i-1]$, independent of all preceding words. That is, conditioned on the topic $y$, the words in the document form a stationary Markov chain, parametrized by an initial distribution $p_{y,\text{init}}$ and the transition distribution $p_{y,\text{tran}}$. Overall, the parameters of the model are $p_{\text{topic}}$, and $\{p_{y,\text{init}}(\cdot), p_{y,\text{tran}}(\cdot,\cdot)\}_{y\in\mathcal{Y}}$, where you can think of $p_{y,\text{init}}$ as a non-negative vector that sums to one, and $p_{y,\text{tran}}$ as a stochastic (i.e. with columns summing to one) non-negative matrix.

When we consider a prior, we will consider a $p_{\text{topic}} \sim Dir(1)$ as before, $p_{y,\text{init}} \sim Dir(\alpha)$ and for each word $s$, $p_{y,\text{tran}}(\cdot|s) \sim Dir(\alpha)$ independently. that is, the columns of the matrix representing the transition distribution are each independently Dirichlet distributed, all with the same parameter $\alpha$.

(a) Derive the maximum likelihood estimator for the parameters.

(b) Derive the MAP estimator for the parameters.

(c) Assuming only two topics, compute the discriminant $r(x)$ such that $P(Y=1|x) = \frac{1}{1+e^{-r(x)}}$,in terms of the parameters.

(d) Write down a feature map $\phi : \mathcal{X} \to \mathbb{R}^D$, for the smallest $D$ you can, such that every discriminant for a distribution in this class can be expressed as $r(x) = \langle w, \phi(x)\rangle$ for some $w \in \mathbb{R}^D$. Write down $w$ in terms of the parameters.

(e) Write down $w$ that corresponds to the MAP parameters.

6. optional **Further Extensions: Variable Document Length and Higher Order Markov**

(a) To simplify presentation, we considered documents with a fixed number of words. Instead, consider a model where we also have a distribution $p_N$ over the number of words in each document, and we choose the length, then a topic, then write it based on the independent or Markov model. Show that as long as the document length is independent of the topic (and the word choice), then the discriminant remains of the same form.

(b) Alternatively, if perhaps documents on different topics ave different lengths we can consider a distribution on $N|y$, e.g. that the conditioned on the topic, the document length is Poission with parameter $\lambda_N$. Derive the maximum likelihood parameter estimates and the discriminant, and show that the discriminant is linear *using the same features*.

(c) optional optional If we want to put a prior on $\lambda_N$ we should use a Gamma distribution—derive the MAP estimate and the corresponding weigh vector $w$.

(d) We can go beyond a Markov model and consider a higher order Markov model, that is where each word depends on the preceding $k$ words, for some small $k > 1$. Derive the maximum likelihood and MAP estimates in this case (again assuming a Dirichlet prior for each conditional distribution), the feature map require to express the discriminant as a linear function, and the weight vector corresponding to the MAP parameters.

We discussed "words" here, but especially when going to higher order Markov models, using characters (or letters) is more attractive. E.g., an order 10 Markov model can capture words, short phrases, and also common endings, subwords, etc. A good prior would be necessary here, and we might also want the prior for different conditional distributions to be related rather than independent. This is beyond the scope of this course.

7. challenge Implement MAP estimation for an independent word model, Markov word model, and $k$-order Markov (try different $k$s, perhaps around 5–10) and train these on the airline tweet data from Homework 4. Compare to using a linear model with the corresponding features, either without regularization or like you did last week with regularization (in the form of the SVM). Follow the same protocol as last time, testing on the test data only on the very end. Submit a python notebook showing how well these generative models perform compared to the SVM from last time, or other discriminate linear methods.

**Experimentation:** Look at the Jupyter Notebook, you will work with Gaussian mixture generative and linear discriminative models, review the mechanics of the models, and compare their relative advantage and disadvantage.