CSE 291: Unsupervised learning

Homework 4

This homework is due on Thursday May 11 at 11.59pm.

- All homeworks must be typewritten and uploaded to Gradescope.
- No late homeworks will be accepted.

1. *Mahalanobis distance.* Let A be a $d \times d$ symmetric matrix that is *positive definite*: that is,
   - $x^T Ax \geq 0$ for all $x \in \mathbb{R}^d$, with equality if and only if $x = 0$.
   - Or equivalently, A can be written in the form UU^T where U is a $d \times d$ full rank (that is, invertible) matrix.

   Show that
   $$\|x\|_A = \sqrt{x^T Ax}$$
   is a norm on $\mathbb{R}^d$ (recall the definition of *norm* from an earlier homework). This yields the Mahalanobis distance
   $$\|x - y\|_A = \sqrt{(x-y)^T A(x-y)}.$$
   Hint: It is helpful to observe that $\|x\|_A = \|U^T x\|_2$.

2. *The ball associated with a norm.* Let $\|\cdot\|$ be an arbitrary norm on $\mathbb{R}^d$. The unit ball associated with this norm is the set
   $$K = \{x \in \mathbb{R}^d : \|x\| \leq 1\}.$$
   (a) *Sketch* the unit ball in $\mathbb{R}^2$ associated with the Mahalanobis norm ($\|x\| = (x^T Ax)^{1/2}$) with matrix
   $$A = \begin{pmatrix} 9 & 0 \\ 0 & 4 \end{pmatrix}.$$
   (b) Show that for any norm, the associated unit ball K is a *convex* set: that is, whenever K contains two points x, y, it also contains the line segment joining them (formally, if $x, y \in K$ then $\theta x + (1-\theta)y \in K$ for all $0 < \theta < 1$).
   (c) Let $\|\cdot\|$ and $\|\cdot\|'$ be two different norms with corresponding unit balls K and K', respectively. If $\|x\| \leq \|x\|'$ for all $x \in \mathbb{R}^d$, what can we conclude about the relationship between K and K'?
   (d) Suppose that norm $\|\cdot\|$ assigns length 1 to the d coordinate vectors $e_1, \ldots, e_d$. What is the smallest possible unit ball that it can have, and for what norm specifically is this minimum realized?

3. *Location parameter for KL-divergence.* The probability simplex Δ_m consists of all distributions $p = (p_1, \ldots, p_m)$ over m outcomes. Suppose we have a (finite) collection $S \subset \Delta_m$ of such distributions,

and we would like to summarize them by a single distribution $q \in \Delta_m$. Specifically, we want to find $q \in \Delta_m$ that minimizes the total KL divergence between q and each of the distributions in S, that is,

$$\sum_{p \in S} K(p, q).$$

(Recall that for $p, q \in \Delta_m$, the KL divergence is $K(p, q) = \sum_{i=1}^{m} p_i \ln(p_i/q_i)$.) What is the desired distribution q? *Hint:* You can do this by calculus; and feel free to assume the natural logarithm.

4. *KL divergence is non-negative.* The KL divergence between distributions p and q can be written in the form

$$K(p, q) = \mathbb{E}_{X \sim p} \ln \frac{p(X)}{q(X)},$$

where $X \sim p$ means that X is drawn at random from p. This is convenient because it works for both discrete and continuous distributions.

Using this formulation, show that $K(p, q) \geq 0$. *Hint:* Apply *Jensen's inequality*, which says that $\mathbb{E}[f(Z)] \geq f(\mathbb{E}[Z])$ for any convex function $f : \mathbb{R} \to \mathbb{R}$ and any random variable Z. In this case, you should use $f(z) = -\ln z$.

5. *Poisson example.* A call center keeps track of the number of phone calls they receive: over a period of 500 hours, they record the number of calls received during every one-hour interval (the number of calls during the first hour, during the second hour, and so on). Let N_k be the number of one-hour intervals during which k calls were received, for $k = 0, 1, 2, \ldots$. Here is their data:

k	0	1	2	3	4	5	6	7	8	≥ 9
N_k	22	66	106	115	85	55	28	13	10	0

Notice that $N_0 + N_1 + \cdots = 500$.

   (a) You decide to model the number of calls received in an hour by a Poisson(λ) distribution. What value of λ should you choose?

   (b) Under this choice of λ, what are the expected entries in the table above, i.e. the expected number of one-hour intervals (out of 500) during which k calls are received, for $k = 0, 1, \ldots$?

6. *Fitting a uniform distribution.* For any real number $\lambda > 0$, let U_λ denote the uniform distribution over $[0, \lambda]$.

   (a) Write down the formula for the density of U_λ.

   (b) Given a set of observations $x_1, x_2, \ldots, x_n > 0$, we decide to fit a U_λ distribution to them. What is the maximum-likelihood choice of λ?

7. *Gamma distribution.*

For $\alpha, \beta > 0$, the gamma(α, β) distribution over $(0, \infty)$ has density

$$p(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x},$$

where $\Gamma(\cdot)$ is the gamma function. (Notice that this implies $\int_0^\infty x^{\alpha-1} e^{-\beta x} = \Gamma(\alpha)/\beta^\alpha$.)

   (a) Suppose $X \sim$ gamma(α, β). Show $\mathbb{E}X = \alpha/\beta$. You might find it helpful to use the relation $\Gamma(z + 1) = z\Gamma(z)$.

(b) Show $\text{var}(X) = \alpha/\beta^2$. (Recall that $\text{var}(X) = \mathbb{E}X^2 - (\mathbb{E}X)^2$.)

8. *Some simple visualization methods.* For this problem, we'll be using the *animals with attributes* data set that you downloaded earlier. Recall that this is a small data set that has information about 50 animals. The animals are listed in `classes.txt`. For each animal, the information consists of values for 85 features: does the animal have a tail, is it slow, does it have tusks, etc. The details of the features are in `predicates.txt`. The full data consists of a 50×85 matrix of real values, in `predicate-matrix-continuous.txt`. Load this real-valued array.

   (a) We would like to visualize these animals in 2-d. Do this with a PCA projection from $\mathbb{R}^{85}$ to $\mathbb{R}^2$. Show the position of each animal, and label each with its name.

   Python notes: You will need to make the plot larger by prefacing your code with

   ```
   from pylab import rcParams
   rcParams['figure.figsize'] = 10, 10
   ```

   (or try a different size if this doesn't seem right).

   (b) A popular visualization method is the *t-SNE algorithm*. This method takes a numerical parameter called the *perplexity* and then obtains an embedding by solving a non-convex optimization problem.

   The t-SNE algorithm is built into `scikit-learn`. You can invoke it to get a 2-d embedding of a data set X by using:

   ```
   from sklearn.manifold import TSNE
   Z = TSNE(n_components=2, perplexity=10.0).fit_transform(X)
   ```

   The *perplexity* has a significant effect on the output. Try different perplexity values (5, 10, 25, 50) and show each of these embeddings, as you did with the PCA embedding.

   Bear in mind that t-SNE is a local search algorithm with randomized initialization, and thus even for a fixed perplexity value, different calls to it can return different results.

   (c) How can we evaluate these embeddings? Some might seem more visually pleasing than others, or might seem to group animals in a way that agrees more with our own intuitions.

   Let's look at a somewhat more objective measure. Say we have a data set of n points $x_1, \ldots, x_n \in \mathbb{R}^d$ and we somehow obtain a 2-d visualization $z_1, \ldots, z_n \in \mathbb{R}^2$ of them. How accurately does this 2-d embedding capture the original interpoint distances? To see this,

   - Define $D_{ij} = \|x_i - x_j\|$ and $\widehat{D}_{ij} = \|z_i - z_j\|$.
   - In general, the D values might be scaled differently from the $\widehat{D}$ values; so define the scaling factor to be $c = \text{mean}(D)/\text{mean}(\widehat{D})$, where $\text{mean}(\cdot)$ denotes the average over all n^2 entries of the matrix.
   - The multiplicative factor by which the distance between x_i and x_j is distorted can be defined by

     $$\Delta_{ij} = \max\left(\frac{D_{ij}}{c \cdot \widehat{D}_{ij}}, \frac{c \cdot \widehat{D}_{ij}}{D_{ij}}\right).$$

     This ratio is always ≥ 1.
   - The *average distortion* is then $\text{mean}(\Delta)$.

   Compute the average distortion for each of the five embeddings you found (PCA and four t-SNE embeddings). Which of them fares best under this measure?