6.790 Homework 6 **Solutions**

Please hand in your work via Gradescope via the link at https://gradml.mit.edu/info/homeworks/. If you were not added to the course automatically, please use Entry Code R7RGGX to add yourself to Gradescope. Make sure to assign the problems to the corresponding pages in your solution when submitting via Gradescope.

1. Latex is not required, but if you are hand-writing your solutions, please write clearly and carefully. You should include enough work to show how you derived your answers, but you don't have to give careful proofs.
2. Homework is due on Thursday November 21 at 11PM.
3. Lateness and extension policies are described at https://gradml.mit.edu/info/class_policy/.

Contents

Solution: Don't look at the solutions until you have tried your absolute hardest to solve the problems. This is especially true for optional problems that you didn't work on—it's a good idea to come back to them when studying for exams.

1 Need a smoothie?

1. Your friend David just learned a new technique called Gaussian process and he's trying to generate some 1D examples to build some intuition on the different covariance functions for a zero-mean Gaussian process prior. Unfortunately he accidentally spilled a smoothie over his laptop lost the code he used to generate the plots. Conveniently he has printed out some plots for different covariance functions earlier and roughly remembers what kind of functions he used to generate these plots. As a good friend who excelled in 6.7900, you are trying to comfort him by labeling the plots.

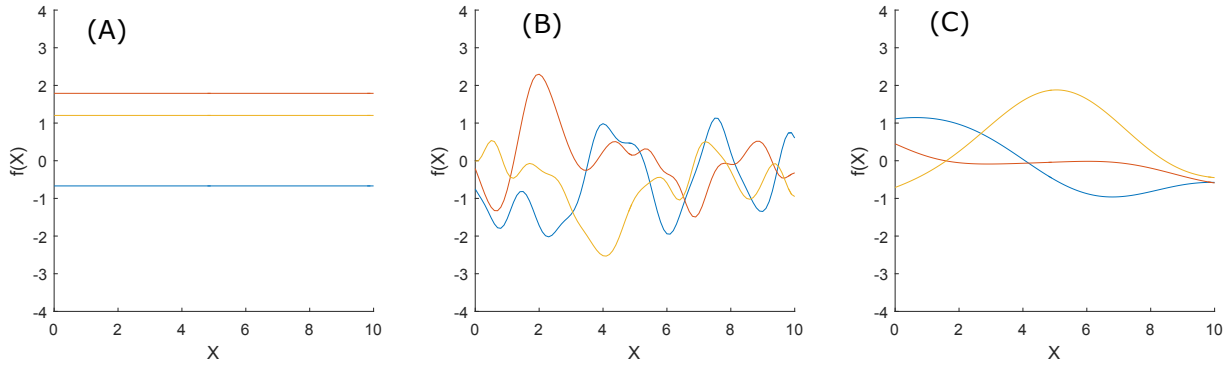
   (a) The following plots contain random functions drawn from a covariance function with a squared exponential kernel $k(x,z) = \exp\left(-\frac{1}{2\tau^2}\|x-z\|^2\right)$ with different values of τ. Indicate which one of them corresponds to:

   i. $\tau = 0.5$

   ii. $\tau = 3$

   iii. $\tau \to \text{inf}$

   Solution: (A): $\tau \to \text{inf}$. (B) $\tau = 0.5$. (C) $\tau = 3$.



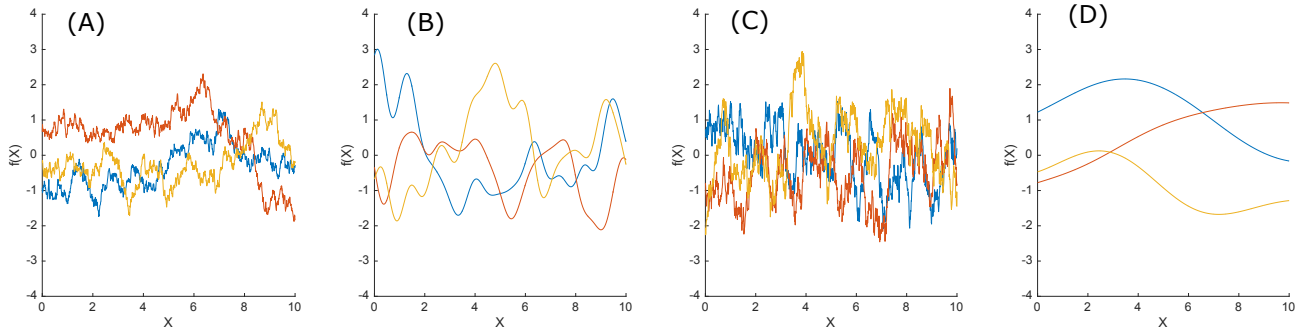
   (b) Qualitatively describe what your random function would look like if you draw with a squared exponential kernel when $\tau \to 0$.

   Solution: Function values drawn across the X axis would be marginally independent of each other regardless of how close they are in X.

   (c) In addition to the squared exponential kernel $k(x,z) = \exp\left(-\frac{1}{2\tau^2}\|x-z\|^2\right)$, David has also tried an exponential kernel in the form of $k(x,z) = \exp\left(-\frac{\|x-z\|}{\tau}\right)$. He has tried two

values for τ for both kernels and the values are $\tau = 3$ and $\tau = 0.5$. For each of the plots below, indicate which kernel function it is generated with which τ value.

Solution: (A): Exponential kernel with $\tau = 3$. (B): Squared exponential kernel with $\tau = 0.5$. (C): Exponential kernel with $\tau = 0.5$. (D): Squared exponential kernel with $\tau = 3$.



2 Calculation with squared exponential kernel

2. Consider a Gaussian process with $\mathbb{E}[f(x)] = 0$, and a squared exponential kernel of the form

$$k(x, z) = \sigma_f^2 \exp\left(\frac{-||x - z||^2}{2l^2}\right),$$

where l is the characteristic length scale and σ_f is the signal standard deviation. Assume we get observations of y values with noise variance σ_N^2.

Let $l^2 = 0.5$, $\sigma_f^2 = 5$, $\sigma_N^2 = 0.01$. Assume you have been given observations $((1,1),(2,-1))$ (these are points in (x, y) space). What is the mean and variance of the prediction at a new query point $x_* = 1.5$? Write down the solution in terms matrices and vectors, you don't need to hand-calculate the final numerical results[1]

Solution: For our training data, we can construct the covariance matrix $K(x, x)$ as

$$K(x, x) = \begin{bmatrix} 5 & 5e^{-1} \\ 5e^{-1} & 5 \end{bmatrix}$$

We can also construct the covariance vector for our new query point x_* with respect to the training data

$$K(x_*, x) = \begin{bmatrix} 5e^{-0.25}, 5e^{-0.25} \end{bmatrix}$$

[1] If you are trying to refer to parameter estimate formula from the textbook "Gaussian Processes for Machine Learning", notice that in eq. (2.21), the predictive distribution is derived with respect to f_*, in order to derive the predictive distribution for y_*, you need to add $\sigma_n^2 I$ to the lower right block of the covariance matrix. This would also impact subsequent equations such as eq. (2.24).

Our posterior predictive mean is

$$\mu_* = K(x_*, x)\left(K(x, x) + \sigma_N^2 I\right)^{-1} y = \left[5e^{-0.25}, 5e^{-0.25}\right]\begin{bmatrix} 5+0.01 & 5e^{-1} \\ 5e^{-1} & 5+0.01 \end{bmatrix}^{-1}[1, -1]^\top = 0$$

Our posterior predictive variance is

$$\begin{aligned} \sigma_*^2 &= K(x_*, x_*) + \sigma_N^2 I - K(x_*, x)\left(K(x, x) + \sigma_N^2 I\right)^{-1} K(x, x_*) \\ &= 5 + 0.01 - \left[5e^{-0.25}, 5e^{-0.25}\right]\begin{bmatrix} 5+0.01 & 5e^{-1} \\ 5e^{-1} & 5+0.01 \end{bmatrix}^{-1}\left[5e^{-0.25}, 5e^{-0.25}\right]^\top = 0.5824 \end{aligned} \tag{1}$$

3 Covariance or not?

3. Recall that for a Gaussian process model the predictive distribution of the response y_* in a test case with inputs x_* has mean and variance given by

$$E[y_*|x_*, \mathcal{D}] = k^T C^{-1} y$$
$$Var[y_*|x_*, \mathcal{D}] = v - k^T C^{-1} k,$$

where y is the vector of observed responses in training cases, C is the matrix of covariances for the responses in training cases, k is the vector of covariances of the response in the test case with the responses in training cases, and v is the prior (co)variance of the response in the test case, and $\mathcal{D}$ is the training data.

(a) Suppose we have just one training case, with $x_1 = 3$ and $y_1 = 4$. Suppose also that the noise-free covariance function is $K(x, x') = 2^{-|x-x'|}$, and the variance of the noise is $\frac{1}{2}$. Find the mean and variance of the predictive distribution for the response in a test case for which the value of the input is 5.

Solution: The equations given above doesn't take into account of the noise of our measurement, if we add in the noise term σ^2, the mean and variance for a new input x_* would be

$$E[y_*|x_*, \mathcal{D}] = k^T (C + \sigma^2)^{-1} y$$
$$Var[y_*|x_*, \mathcal{D}] = v + \sigma^2 - k^T (C + \sigma^2)^{-1} k$$

Therefore, the mean of the predictive distribution is

$$K(3,5) \cdot \left(K(3,3) + \frac{1}{2}\right)^{-1} \cdot 4 = \frac{1}{4} \cdot \left(1 + \frac{1}{2}\right)^{-1} \cdot 4 = \frac{2}{3}$$

The variance of the predictive distribution is

$$\left(K(5,5)+\frac{1}{2}\right)-K(3,5)\left(K(3,3)+\frac{1}{2}\right)^{-1}K(3,5)$$
$$=\left(1+\frac{1}{2}\right)-\left(\frac{1}{4}\right)\left(1+\frac{1}{2}\right)^{-1}\left(\frac{1}{4}\right)=\frac{35}{24}$$

(b) Repeat the calculations, but using $K(x,x')=2^{+|x-x'|}$. What can you conclude from the result of this calculation?

Solution:
The mean of the predictive distribution is

$$K(3,5)\cdot\left(K(3,3)+\frac{1}{2}\right)^{-1}\cdot 4=4\cdot\left(1+\frac{1}{2}\right)^{-1}\cdot 4=\frac{32}{3}$$

The variance of the predictive distribution is

$$\left(K(5,5)+\frac{1}{2}\right)-K(3,5)\cdot\left(K(3,3)+\frac{1}{2}\right)^{-1}\cdot K(3,5)=\left(1+\frac{1}{2}\right)-4\cdot\left(1+\frac{1}{2}\right)^{-1}\cdot\frac{1}{4}=-\frac{55}{6}$$

Notice that the variance in this case is a negative number, which is clearly wrong. We can therefore conclude that $K(x,x')=2^{+|x-x'|}$ is not a valid covariance function – it is not positive semi-definite.

4 Mixed-up mixture

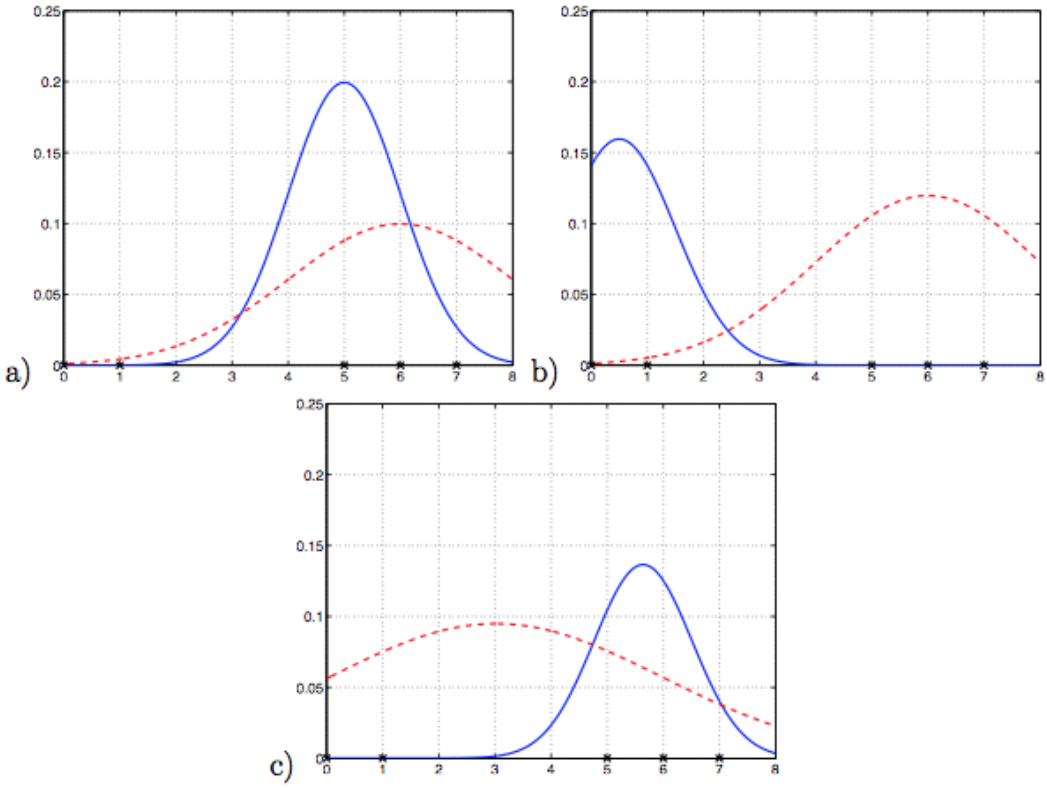
4. Here we are estimating a mixture of two Gaussians via the EM algorithm. The mixture distribution over x is given by

$$P(x;\theta)=P(1)N(x;\mu_1,\sigma_1^2)+P(2)N(x;\mu_2,\sigma_2^2)$$

Any student in this class could solve this estimation problem easily. Well, one student, devious as they were, scrambled the order of figures illustrating EM updates. They may have also slipped in a figure that does not belong. Your task is to extract the figures of successive updates and explain why your ordering makes sense from the point of view of how the EM algorithm works. All the figures plot $P(1)N(x;\mu_1,\sigma_1^2)$ as a function of x with a solid line and $P(2)N(x;\mu_2,\sigma_2^2)$ with a dashed line. The sampled data points are given along the x axis in the figures at $x=0,1,2,5,6,7$ and are denoted by the cross marks on the x axis.

(a) (True/False) In the mixture model, we can identify the most likely T posterior assignment, i.e., j that maximizes $P(j \mid x)$, by comparing the values of $P(1)N(x;\mu_1,\sigma_1^2)$ and $P(2)N(x;\mu_2,\sigma_2^2)$

Solution: True



(b) Assign two figures to the correct steps in the EM algorithm.
- Step 0: ( ) initial mixture distribution
- Step 1: ( ) after one EM-iteration

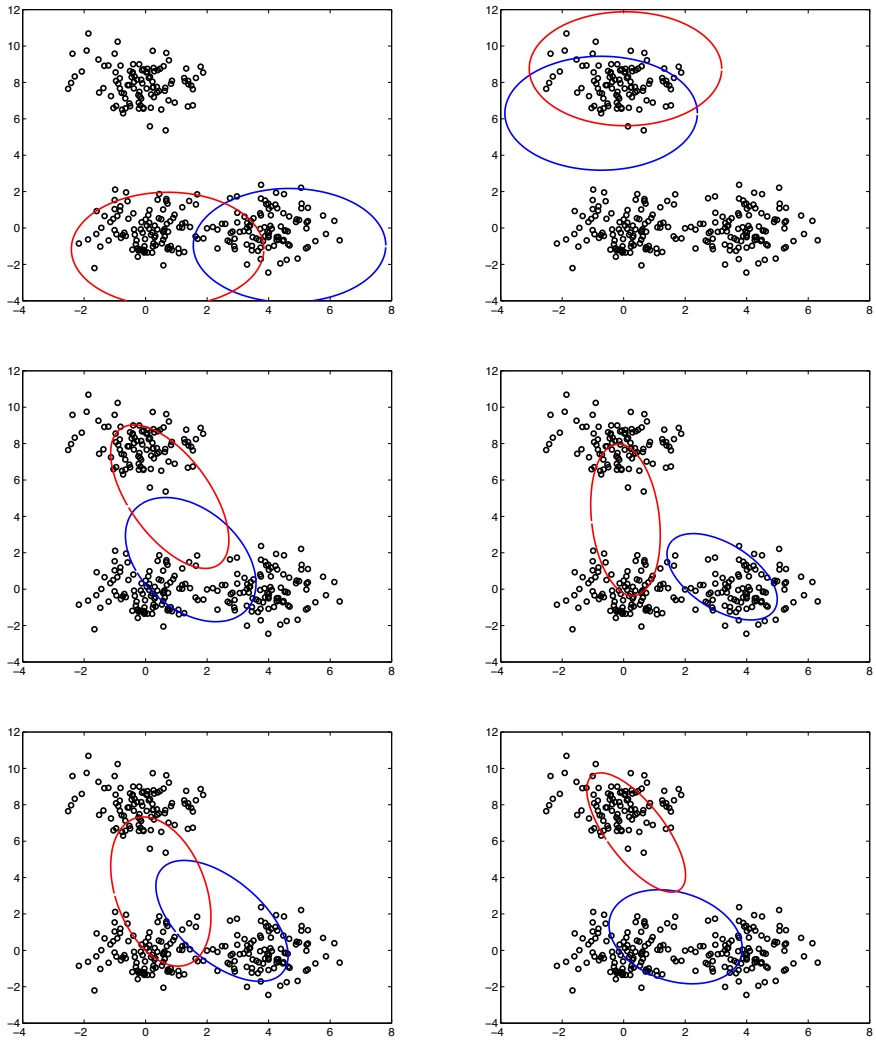
Solution: - Step 0: a
- Step 1: c

(c) Briefly explain how the mixture you chose for "step 1" follows from the mixture you have in "step 0".

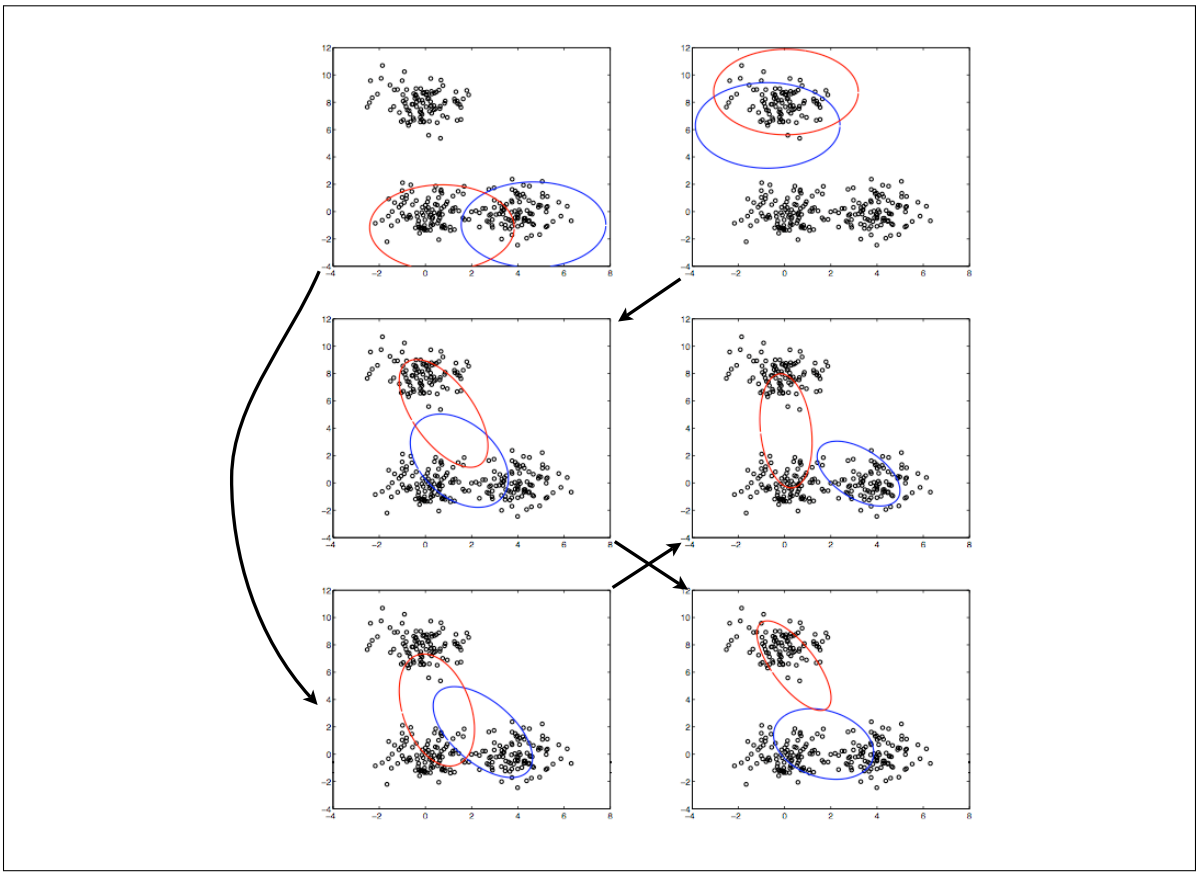
Solution: The two points on the left will be assigned more to the second (red) Gaussian since $P(1)N(x;\mu_1,\sigma_1^2) < P(2)N(x;\mu_2,\sigma_2^2)$ for those points. The points on the right, except for the very last one, will be assigned mostly to the first (blue) Gaussian. As a result, the first Gaussian will become more concentrated around the two points on the right, while the second (red) Gaussian will move to the left and will have a higher variance as, in the M-step, it is estimated essentially on the basis of the spread out points $x = 0$, $x = 1$, and $x = 7$.

5 More mixture

5. We estimated a two Gaussians mixture model based on two-dimensional data shown in the figure below. The mixture was initialized randomly in two different ways and run for three iterations based on each initialization. However, the figures got mixed up (yes, again!). Please draw an arrow from one figure to another to indicate how they follow from each other (you should draw only four arrows). The ellipses represent the 1 standard deviation equi-probability contours of the Gaussians. The small circles represent the sampled data points.



Solution:



6 Missing data

6. We'll start with a very simple problem, in which single attribute of a single data set is missing. There are two attributes, A and B, and this is our data set, $\mathcal{D}$:

```
i    A  B
1    1  1
2    1  1
3    0  0
4    0  0
5    0  0
6    0  H ***missing **
7    0  1
8    1  0
```

Assume the data is *missing completely at random* (MCAR): that is, that the fact that it is missing is independent of its value.

Our goal is to estimate $\Pr(A, B)$ from this data. We'd really like to find the maximum-likelihood parameter values, if we can. The likelihood is

$$\mathcal{L}(\theta) = \log \Pr(\mathcal{D}; \theta) = \log \left(\Pr(\mathcal{D}, H = 0; \theta) + \Pr(\mathcal{D}, H = 1; \theta)\right) \quad .$$

(a) Kim is lazy and decides to ignore $x^{(6)}$ all together, and estimate the parameters:

$$\widehat{\theta}^1 = \begin{pmatrix} P(A=0,B=0) & P(A=0,B=1) \\ P(A=1,B=0) & P(A=1,B=1) \end{pmatrix} = \begin{pmatrix} 3/7 & 1/7 \\ 1/7 & 2/7 \end{pmatrix} = \begin{pmatrix} .429 & .143 \\ .143 & .285 \end{pmatrix}$$

What is $\mathcal{L}(\widehat{\theta}^1)$?

Solution: If we do that, then

$$\begin{aligned} \mathcal{L}(\widehat{\theta}^1) &= \log\left(\Pr(00\,;\,\widehat{\theta}^1)\prod_{i\neq 6}\Pr(x^i\,;\,\widehat{\theta}^1) + \Pr(01\,;\,\widehat{\theta}^1)\prod_{i\neq 6}\Pr(x^i\,;\,\widehat{\theta}^1)\right) \\ &= 3\log 0.429 + 2\log 0.143 + 2\log 0.285 + \log(0.429+0.143) \\ &= -9.498 \end{aligned}$$

(b) Jan thinks we should let H be the 'best' value it could have, that is to make the log likelihood as large as possible, and so tries setting $H=0$ and then $H=1$ and computes the log likelihood of the complete data in both cases. What value gives the highest complete-data log likelihood? What is the likelihood value?

Solution: That value is 0. So, then we'd have

$$\widehat{\theta}^2 = \begin{pmatrix} .5 & .125 \\ .125 & .25 \end{pmatrix}$$

and

$$\mathcal{L}(\widehat{\theta}^2) = -9.481 \quad .$$

That's a little better!

(c) Evelyn thinks this is all unprincipled messing around and says we should optimize the thing we want to optimize! That is,

$$\widehat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta) \quad .$$

Evelyn also thinks we can just use the code for gradient descent that we already built in 6.7900 to do this job.

Is Evelyn right?

Solution: Evelyn is absolutely right about (if at all possible!) optimizing the thing we want to optimize.

We can do this with gradient methods, but it gets tricky because of constraint that $\widehat{\theta}$ be a valid probability distribution; that constraint is *not* maintained by our basic gradient descent code. So, we'd have to investigate constrained optimization algorithsm, or try to formulate the problem using Lagrange multipliers.

(d) Ariel was paying close attention in lecture and thinks this problem is an example of estimation in the presence of a latent variable and that we should use EM.

Let's start with the guess

$$\theta_0 = \begin{pmatrix} .25 & .25 \\ .25 & .25 \end{pmatrix}$$

What is the formula for the E step in this problem? What is the numerical result in this particular case?

Solution:

$\tilde{P}(H = 1) = \Pr(H = 1 \mid \mathcal{D}\ ;\ \theta_0) = \Pr(H = 1 \mid x^{(6)}\ ;\ \theta_0) = \Pr(B = 1 \mid A = 0\ ;\ \theta_0) = 0.5$

(e) Ariel's roommate Angel joins in the EM game and computes the M step, to get θ_1. What is the numerical value in this case, and why?

Solution:

$$\begin{aligned} \theta_1 &= \arg\max_{\theta} \left(0.5 \log \Pr(\mathcal{D}, H = 0\ ;\ \theta) + 0.5 \log \Pr(\mathcal{D}, H = 1\ ;\ \theta)\right) \\ &= \begin{pmatrix} 7/16 & 3/16 \\ 2/16 & 4/16 \end{pmatrix} \end{aligned}$$

This step is not immediately obvious: to derive it, we need to take the derivative with respect to each of the parameters, set to 0, and solve for θ. We find that we can treat the estimation problem as one in which we have a data item for each possible value of H, weighted by the probability that H has that value. We get such a decomposition because, for each particular value of H, only one of the parameter estimates is affected.

We get the same result by doing estimation as usual, but treating $\Pr(H)$ as giving us fractional counts on both data cases:

i	A	B	count = P~(H)
1	1	1	
2	1	1	
3	0	0	
4	0	0	
5	0	0	
6a	0	0	0.5
6b	0	1	0.5
7	0	1	
8	1	0	

On subsequent EM iterations, we have $\mathcal{L}(\theta) = -10.39, -9.47, -9.4524, -9.4514, \ldots$.

(f) Will EM always find a solution that maximizes $\mathcal{L}$?

Solution: No. It will converge monotonically to a *local optimum* but it may not be a global optimum, and will depend, in general, on your initial guess.

7 Gradient descent for Gaussian mixture

7. It is typical to fit a Gaussian mixture model to data using EM (expectation maximization) but we can also use gradient descent!

   Assume we have a latent discrete variable Z with values in $\{1, \ldots, K\}$ and an observable continuous variable X with values in $\mathbb{R}^d$.

   The likelihood of the data, as a function of the parameters, is

$$\mathcal{L}(\pi, \mu, \Sigma) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \log \mathcal{N}(x^{(i)} \; ; \; \mu_k, \Sigma_k)$$

   Assuming we know K, we would like to find π, μ, and Σ to **maximize** this quantity.

   (a) The first problem we face is that our parameters are constrained to be in a limited space: the π have to constitute a probability distribution (be in the range $[0, 1]$) and the σ have to be valid covariance matrices (positive definite). For simplicity, let's assume that

$$\Sigma_j = I\sigma_j^2$$

   for $\sigma_j > 0$ (that is, that the covariances are round).

   What is a different parameterization for π and the σ_j values so that we can do *unconstrained* gradient descent on them?

   Solution: Let

$$\pi_j = \frac{\exp(a_j)}{\sum_{j=1}^{k} \exp(a_j)}$$

   and let

$$\sigma_j = \exp(b_j)$$

   Now we can just optimize the a_j's and b_j's and μ's.

   (b) Now, let's look at a very simple version of this problem, with a single data point in 1D, with two components. We get

$$\mathcal{L}(\pi_1, \pi_2, \mu_1, \mu_2, \sigma_1, \sigma_2) = \pi_1 \mathcal{N}(x \; ; \; \mu_1, \sigma_1) + \pi_2 \mathcal{N}(x \; ; \; \mu_2, \sigma_2)$$

   We find that

$$\frac{\partial \mathcal{L}}{\partial \mu_1} = \pi_1 (x - \mu_1) \frac{\exp(-\frac{(x-\mu_1)^2}{2\sigma_1^2})}{\sqrt{2\pi}\sigma_1^3}$$

$$\frac{\partial \mathcal{L}}{\partial \mu_1} = \pi_1 \cdot \frac{(x - \mu_1)}{\sigma_1^2} \cdot \mathcal{N}(x\ ;\ \mu_1, \sigma_1)$$

and, of course, get a symmetric result for $\partial \mathcal{L}/\partial \mu_2$.

i. For a training example x, if we do one SGD update, in what directions will μ_1 and μ_2 move?

> **Solution:** μ_1 and μ_2 will move in the direction of x.

ii. What governs which μ parameter will be changed the most?

> **Solution:** The magnitude of change is governed by the likelihood of being assigned to each component divided by the standrad deviation. We compare which one is larger to determine which μ will be changed the most.

(c) Staying with the simple 1D 2-component version, letting $\pi_1 = \exp(a_1)/(\exp(a_1)+\exp(a_2))$, we find that

$$\frac{\partial \mathcal{L}}{\partial a_1} = \frac{\exp(a_1 + a_2)}{(\exp(a_1) + \exp(a_2))^2} \big(\mathcal{N}(x\ ;\ \mu_1, \sigma_1) - \mathcal{N}(x\ ;\ \mu_2, \sigma_2)\big)$$

and get a symmetric result for $\partial \mathcal{L}/\partial a_2$. Let's assume that given the current parameters, x is much more likely given μ_1, σ_1 than given μ_2, σ_2. In one SGD update with input x, describe how π_1 and π_2 would be changed.

> **Solution:** In the expression for a_1, the difference in PDFs is positive, so a_1 would increase. a_2 would symmetrically decrease, so we move π_1 up and π_2 down.

(d) Finally, in this same problem, but letting $\sigma_1 = \exp(b_1)$, we find that

$$\frac{\partial \mathcal{L}}{\partial b_1} = \pi_1 \mathcal{N}(x\ ;\ \mu_1, \exp(b_1)) \left(\frac{(\mu_1 - x)^2}{\exp(2b_1)} - 1\right)$$

and get a symmetric result for $\partial \mathcal{L}/\partial b_2$ In one SGD update with input x, describe how σ_1 and σ_2 would be changed.

> **Solution:** All terms in the expression are positive except for $\left(\frac{(\mu_1 - x)^2}{\exp(2b_1)} - 1\right)$. This is positive if $\frac{(\mu_1 - x)^2}{\exp(2b_1)} = \frac{(\mu_1 - x)^2}{\sigma_1^2} > 1$. So if x_1 is more than 1 standard deviation from μ_1, σ_1 will increase, otherwise it will decrease. A symmetric argument holds for σ_2.

8 Principles of principal components

8. Principal components are related to several other ideas we have come across in class so far. We'll explore this in two dimensions. Imagine we have a data set $\mathcal{D} = \{(x_1^{(i)}, x_2^{(i)})\}_{i=1}^n$.

(a) In homework 0, we observed that the eigenvectors of the covariance matrix of a multi-dimensional Gaussian corresponded to axes of ellipses describing equi-probability contours.

Show that the eigenvector of the covariance matrix with the largest corresponding eigenvalue is equivalent to the first "principal" component of the data.

Solution: Let the first principal component be the vector v. The definition of first principal component is

$$v = \arg\max_{||v||=1} \sum_{i=1}^{n} (x_i \cdot v)^2$$
$$\implies v = \arg\max_{||v||=1} v^T X^T X v.$$

This is precisely the definition of the eigenvector corresponding to the largest eigenvalue of $X^T X$, the covariance matrix.

(b) One way to describe the first principal component is that it is the line such that the sum of the perpendicular distances of the points to the line is minimized. This sounds sort of like what's happening in ordinary least squares. Explain why they are different and draw a picture of a small data-set (4 points) in which the solutions are substantially different.

Solution: In ordinary least squares, we are trying to minimize the sum of the squared *vertical* distances of the points to the line. In principal components, we are trying to minimize the sum of the squared *perpendicular* distances of the points to the line.

Cconsider the data set $\{(0,0),(0,1),(1,0),(1,1)\}$. The line that minimizes the sum of the squared vertical distances is the line $y = x$. A line that minimizes the sum of the squared perpendicular distances is the line $y = 1 - x$.