CSE 512: Homework 2 **Due Wednesday September 25**

Extra practice problems, ungraded

1. *Gradients.* Compute the gradients of the following functions. Give the exact dimension of the output.

   (a) *Linear regression.* $f(x) = \frac{1}{40}\|Ax - b\|_2^2$, $A \in \mathbb{R}^{20\times 10}$

   Ans. Actually, the best way to do this is to invoke the chain rule, which you will prove in the first graded problem. Write $g(v) = \frac{1}{40}\|v - b\|_2^2$. Then since $b \in \mathbb{R}^{20}$,

   $$\nabla g(v) = \nabla_v \left(\frac{1}{40} \sum_{i=1}^{20} (v[i] - b[i])^2 \right) \overset{\text{linearity}}{=} \frac{1}{40} \sum_{i=1}^{20} \nabla_v \left((v[i] - b[i])^2 \right).$$

   Note that

   $$\nabla_v (v[i] - b[i])^2 = \begin{bmatrix} \frac{\partial}{\partial v[1]}(v[i] - b[i])^2 \\ \frac{\partial}{\partial v[2]}(v[i] - b[i])^2 \\ \vdots \\ \frac{\partial}{\partial v[20]}(v[i] - b[i])^2 \end{bmatrix}$$

   and

   $$\frac{\partial}{\partial v[k]}(v[i] - b[i])^2 = \begin{cases} 2(v[i] - b[i]) & \text{if } i = k \\ 0 & \text{else.} \end{cases}$$

   So,

   $$\sum_{i=1}^{20} \nabla_v (v[i] - b[i])^2 = 2 \begin{bmatrix} (v[1] - b[1]) \\ (v[2] - b[2]) \\ \vdots \\ (v[20] - b[20]) \end{bmatrix} = 2(v - b).$$

   and $\nabla g(v) = \frac{1}{20}(v - b)$.

   Now, we invoke the chain rule. (Note that f and g are flipped as to their position in 1.(b).) Then

   $$\nabla f(x) = A^T \nabla g(Ax) = A^T (\frac{1}{20}(Ax - b)) = \frac{1}{20} A^T (Ax - b).$$

   To get the dimension, you can do this in two ways. One, you notice that A has 10 columns, so A^T has 10 rows. Two, you notice that the gradient $\nabla f(x)$ should always have the same number of elements as x, which is 10. In either case, $\nabla f(x) \in \mathbb{R}^{10}$.

   (b) *Sigmoid.* $f(x) = \sigma(c^T x)$, $c \in \mathbb{R}^5$, $\sigma(s) = \frac{1}{1+\exp(-x)}$. Hint: Start by showing that $\sigma'(s) = \sigma(s)(1 - \sigma(s))$.

   Ans. We start with the hint, noting that

   $$\sigma'(s) = \frac{\exp(-x)}{(1 + \exp(-x))^2} = \frac{1}{1 + \exp(-x)} \cdot \left(1 - \frac{1}{1 + \exp(-x)} \right) = \sigma(s)(1 - \sigma(s)).$$

   Then using chain rule, (where $A = c^T$) we can get

   $$\nabla f(x) = \sigma'(c^T x)c = \sigma(c^T x)(1 - \sigma(c^T x))c \in \mathbb{R}^5.$$

Main assignment, graded

1. **(1 pts, 0.5 pts each)** *Linearity.* A function $f : \mathbb{R}^n \to \mathbb{R}$ is *linear* if for any x and y in the domain of f, and any scalar α and β,
$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y).$$
Are the following functions linear? Justify your answer.

   (a) $f(x) = \|x\|_2^2$
   (b) $f(x) = c^T x + b^T A x$

2. **(1 pt, 0.5 each)** A function $f : \mathbb{R}^n \to \mathbb{R}$ is a norm if it satisfies three properties:

   - Nonnegativity: $f(x) \geq 0$ for all x and $f(x) = 0$ only when $x = 0$
   - Positive homogeneity $f(\alpha x) = \alpha f(x)$ whenever $\alpha \geq 0$
   - Triangle inequality $f(x + y) \leq f(x) + f(y)$.

   Using the properties of norms, verify that the following are norms, or prove that they are not norms by finding a counterexample.

   (a) *Sum of square roots, squared.* $f : \mathbb{R}^d \to \mathbb{R}$, $f(x) = \left(\sum_{k=1}^d \sqrt{|x[k]|}\right)^2$
   Ans. This is not a norm, since it is cannot satisfy triangle inequality. In particular, just take $d = 2$ and
$$f\left(\begin{bmatrix}1\\0\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}\right) = f\left(\begin{bmatrix}1\\1\end{bmatrix}\right) = (\sqrt{1} + \sqrt{1})^2 = 4$$
$$f\left(\begin{bmatrix}1\\0\end{bmatrix}\right) + f\left(\begin{bmatrix}0\\1\end{bmatrix}\right) = 1 + 1 = 2$$
   and therefore we have shown that $f(x + y) > f(x) + f(y)$ for some choice of x, y.

   (b) *Weighted 2-norm.* $f : \mathbb{R}^d \to \mathbb{R}$, $f(x) = \sqrt{\sum_{k=1}^d \frac{|x[k]|^2}{k}}$
   Ans. Yes, this is a norm. To see that, first note that we can write
$$f(x) = \|Wx\|_2, \qquad W = \mathbf{diag}(1, 1/2, 1/3, ..., 1/d).$$
   Then we can just go about checking the norm conditions.
   - 0 at 0? Yes, $f(0) = \|W\,0\|_2 = \|0\|_2 = 0$
   - Positive homogeneity?
$$f(\alpha x) = \|W(\alpha x)\|_2 = |\alpha|\|Wx\|_2 = |\alpha| f(x)$$
     Check.
   - Triangle inequality?
$$f(x + y) = \|W(x + y)\|_2 = \|Wx + Wy\|_2 \overset{\Delta\text{-ineq of 2-norm}}{\leq} \|Wx\|_2 + \|Wy\|_2 = f(x) + f(y)$$
     Check!

   Therefore this is a norm.

3. *Gradient properties.* **(1 pt, 0.5 pts each.)** Prove the following two properties of gradients:

   (a) *Linearity.* If $h(x) = \alpha f(x) + \beta g(x)$, then $\nabla h(x) = \alpha \nabla f(x) + \beta \nabla g(x)$.
   Ans. Actually, this is just a direct consequence that partial derivatives are linear. That is,
$$\frac{\partial h}{\partial x[i]}(x) = \alpha \frac{\partial f}{\partial x[i]}(x) + \beta \frac{\partial g}{\partial x[i]}(x).$$

Therefore,

$$\nabla h(x) = \begin{bmatrix} \frac{\partial h}{\partial x[1]}(x) \\ \frac{\partial h}{\partial x[2]}(x) \\ \vdots \\ \frac{\partial h}{\partial x[n]}(x) \end{bmatrix} = \begin{bmatrix} \alpha \frac{\partial f}{\partial x[1]}(x) + \beta \frac{\partial g}{\partial x[1]}(x) \\ \frac{\partial f}{\partial x[2]}(x) + \beta \frac{\partial g}{\partial x[2]}(x) \\ \vdots \\ \frac{\partial f}{\partial x[n]}(x) + \beta \frac{\partial g}{\partial x[n]}(x) \end{bmatrix} = \alpha \begin{bmatrix} \frac{\partial f}{\partial x[1]}(x) \\ \frac{\partial f}{\partial x[2]}(x) \\ \vdots \\ \frac{\partial f}{\partial x[n]}(x) \end{bmatrix} + \beta \begin{bmatrix} \frac{\partial g}{\partial x[1]}(x) \\ \frac{\partial g}{\partial x[2]}(x) \\ \vdots \\ \frac{\partial g}{\partial x[n]}(x) \end{bmatrix} = \alpha \nabla f(x) + \beta \nabla g(x).$$

(b) *Chain rule.* Show that if $g(v) = f(Av)$, then $\nabla g(v) = A^T \nabla f(Av)$.
Ans. The easiest way to do this is to just brute force it. We denote the columns of $A \in \mathbb{R}^{m \times n}$ as

$$A = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix}, \quad a_i \in \mathbb{R}^m.$$

Then

$$\frac{\partial g}{\partial v[i]}(v) = \sum_k \frac{\partial f}{\partial x[k]}(a_k^T v) \cdot A[k,i] = a_i^T \nabla f(Av).$$

Therefore,

$$\nabla g(v) = \begin{bmatrix} \frac{\partial g}{\partial v[1]}(v) \\ \frac{\partial g}{\partial v[2]}(v) \\ \vdots \\ \frac{\partial g}{\partial v[n]}(v) \end{bmatrix} = \begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_n^T \end{bmatrix} \nabla f(Av) = A^T \nabla f(Av).$$

4. *Gradients.* **(2 pts, 1 pt each.)** Compute the gradients of the following functions. Give the exact dimension of the output.

(a) *Quadratic function.* $f(x) = \frac{1}{2} x^T Q x + p^T x + r$, $Q \in \mathbb{R}^{12 \times 12}$ and Q is symmetric ($Q[i,j] = Q[j,i]$).
Ans. We can do this piece by piece. First, consider

$$f_1(x) = \frac{1}{2} x^T Q x = \frac{1}{2} \sum_{i=1}^{12} \sum_{j=1}^{12} Q[i,j] x[i] x[j].$$

Then

$$\frac{\partial f_1(x)}{\partial x[k]} = \frac{1}{2} \sum_{i=1}^{12} \sum_{j=1}^{12} \frac{\partial}{\partial x[k]} (Q[i,j] x[i] x[j])$$

and

$$\frac{\partial}{\partial x[k]} (Q[i,j] x[i] x[j]) = \begin{cases} Q[k,j] x[j] & \text{if } k = i \\ Q[i,k] x[i] & \text{if } k = j \\ 0 & \text{else.} \end{cases}$$

So, we can get to

$$\frac{\partial f_1(x)}{\partial x[k]} = \frac{1}{2} \left(\frac{\partial}{\partial x[k]} Q[k,j] x[j] + \sum_{i=1}^{12} Q[i,k] x[i] \right) = \frac{1}{2} \cdot 2Qx = Qx.$$

Now let's consider $f_2(x) = p^T x = \sum_{k=1}^{12} p[k] x[k]$. Then

$$\nabla f_2(x) = \begin{bmatrix} \frac{\partial}{\partial x[1]} \left(\sum_{k=1}^{12} p[k] x[k] \right) \\ \frac{\partial}{\partial x[2]} \left(\sum_{k=1}^{12} p[k] x[k] \right) \\ \vdots \\ \frac{\partial}{\partial x[12]} \left(\sum_{k=1}^{12} p[k] x[k] \right) \end{bmatrix} = \begin{bmatrix} p[1] \\ p[2] \\ \vdots \\ p[12] \end{bmatrix} = p.$$

Sometimes, we refer to this property

$$\frac{\partial}{\partial x[j]}\left(\sum_{k=1}^{12} p[k]x[k]\right) = p[j]$$

as a "picking property", because we pick out the element of p that we're interested in. Finally, observing that r is a constant, we get

$$\nabla f(x) = \underbrace{Qx}_{\nabla f_1(x)} + \underbrace{p}_{\nabla f_2(x)} \in \mathbb{R}^{12}.$$

(b) *Softmax function.* $f(x) = \frac{1}{\mu}\log(\sum_{i=1}^{8}\exp(\mu x[i]))$, $x \in \mathbb{R}^8$, μ is a positive scalar
Ans. Again, it's useful here to use chain rule. In particular, we decompose

$$f(x) = g(\sum_i h(x_i)), \quad g(s) = \frac{1}{\mu}\log(s),\ h(z) = \exp(\mu z)$$

with derivatives

$$g'(s) = \frac{1}{\mu s}, \quad h'(z) = \mu\exp(\mu z).$$

Then using chain rule,

$$\frac{\partial f(x)}{\partial x[k]} = g'(\sum_i h(x_i))h'(x[k])$$

and plugging in everything, we get

$$\frac{\partial f(x)}{\partial x[k]} = \frac{1}{\mu\sum_i \exp(\mu x[i])}\cdot\mu\exp(\mu x[k]) = \frac{\exp(\mu x[k])}{\sum_i \exp(\mu x[i])}.$$

In matrix form, we write

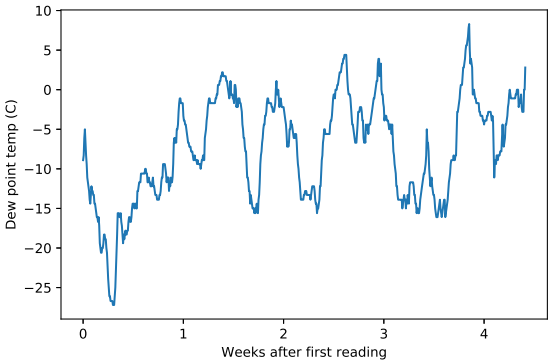
$$\nabla f(x) = \frac{1}{\sum_i \exp(\mu x[i])}\begin{bmatrix}\exp(\mu x[1])\\ \exp(\mu x[2])\\ \vdots \\ \exp(\mu x[8])\end{bmatrix} \overset{\text{abuse of notation}}{=} \frac{\exp(\mu x)}{\sum_i \exp(\mu x[i])} \in \mathbb{R}^8.$$

1

[1]For those of you who train neural networks for multiclass classification, you probably recognize this as the softmax layer. Well, this is where the name "softmax" comes from! (The function $f(x)$ gives a "soft approximation" of the maximum element of x_i.)

5. *Polyfit via linear regression.* **(3 pts)**

- Download weatherDewTmp.mat. Plot the data. It should look like the following



- We want to form a polynomial regression of this data. That is, given w = weeks and d = dew readings, we want to find $\theta_1, ..., \theta_p$ as the solution to

$$\underset{\theta \in \mathbb{R}^p}{\text{minimize}} \quad \frac{1}{2} \sum_{i=1}^{m} (\theta_1 + \theta_2 w_i + \theta_3 w_i^2 + \cdots + \theta_p w_i^{p-1} - d_i)^2. \tag{1}$$

Form X and y such that (1) is equivalent to the least squares problem

$$\underset{\theta \in \mathbb{R}^p}{\text{minimize}} \quad \frac{1}{2} \|X\theta - y\|_2^2. \tag{2}$$

That is, for w the vector containing the week number, and y containing the dew data, form

$$X = \begin{bmatrix} 1 & w_1 & w_1^2 & w_1^3 & \cdots & w_1^{p-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 1 & w_m & w_m^2 & w_m^3 & \cdots & w_m^{p-1} \end{bmatrix}.$$

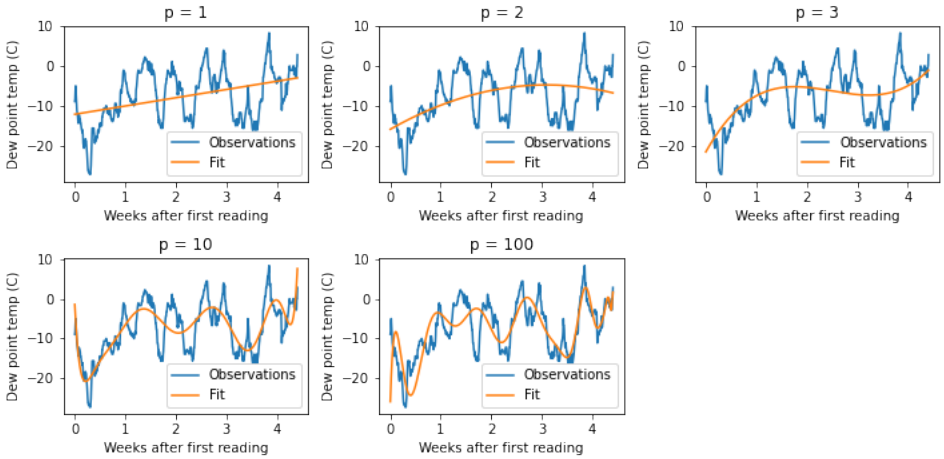
(a) *Linear regression.* **(1pt)**

i. Write down the normal equations for problem (2). **Ans.** The normal equations are characterized by the linear system that emerges from setting the gradient of (2) to 0:

$$X^T X\theta = X^T y.$$

ii. Fill in the code to solve the normal equations for θ, and use it to build a predictor. To verify your code is running correctly, the number after `check number` should be 1.759 (implemented correctly) or 1.341 (also accepted).

iii. Implement a polynomial fit of orders $p = 1, 2, 3, 10, 100$, for the weather data provided. Include a figure that plots the original signal, overlaid with each polynomial fit. Comment on the "goodness of fit" for each value of p. **Ans.**



The goodness of fit definitely improves with larger p, with no obvious defects. However, it is possible that as p gets really large, some overfitting may be occurring. (Graders, give full credit for any relevant observation that is not false.)

(b) *Ridge regression.* **(0.5pt)** Oftentimes, it is helpful to add a *regularization term* to (2), to improve stability. In other words, we solve

$$\underset{\theta \in \mathbb{R}^p}{\text{minimize}} \quad \tfrac{1}{2}\|X\theta - y\|_2^2 + \tfrac{\rho}{2}\|\theta\|_2^2. \tag{3}$$

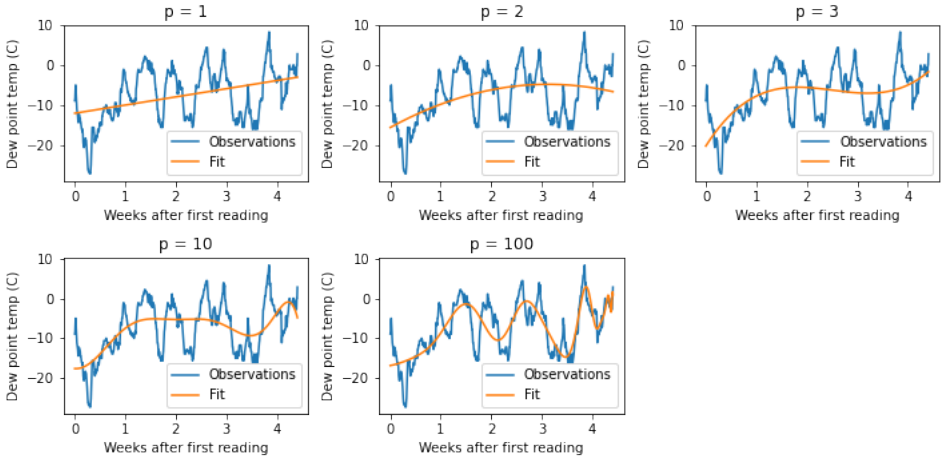
for some $\rho > 0$.

i. Again, write down the normal equations for (3). Your equation should be of form $A\theta = b$ for some matrix A and vector b that you specify.
Ans. The normal equations here just need to also include the regularization term in the gradient of (3):

$$(X^T X + \rho I)\theta = X^T y.$$

ii. Write the code for solving the ridge regression problem and run it. To verify your code is running correctly, the number after `check number` should be *Checknumber* : 1.636 (implemented correctly) or 1.206 (also accepted).

iii. Using $\rho = 1.0$, plot the weather data with overlaying polynomial fits with ridge regression. Provide these plots for $p = 1, 2, 3, 10, 100$. Comment on the "goodness of fit" and the stability of the fit, and also compare with the plots generated without using the extra penalty term.
Ans.



With such a large value of ρ, you do see more smoothing in the resulting figure. Whether or not that's a good thing is hard to decide; less overfitting may be more stable, but more overfitting may be more accurate.

(Graders, give full credit for any relevant observation that is not false.)

(c) *Conditioning.* **(1pt)**

i. An *unconstrained quadratic problem* is any problem that can be written as

$$\underset{\theta}{\text{minimize}} \quad \frac{1}{2}\theta^T Q\theta + c^T\theta + r \tag{4}$$

for some symmetric positive semidefinite matrix Q, and some vector c and some scalar r. Show that the ridge regression problem (3) is an unconstrained quadratic problem by writing down Q, c, and r in terms of X and y such that (4) is equivalent to (3). Show that the Q you picked is positive semidefinite.
Ans. Expanding the ridge regression objective function gives Q, c, and r:

$$\frac{1}{2}\|X\theta - y\|_2^2 + \frac{\rho}{2}\|\theta\|_2^2 = \frac{1}{2}\theta^T X^T X\theta - y^T X\theta + \frac{1}{2}y^T y + \frac{\rho}{2}\theta^T\theta = \frac{1}{2}\theta^T \underbrace{(X^T X + \rho I)}_{Q}\theta \underbrace{-y^T X}_{c}\theta + \underbrace{\frac{1}{2}y^T y}_{r}.$$

To see that Q is positive semidefinite, pick any vector u. Then

$$u^T Qu = u^T X^T Xu + \rho u^T u = \|Xu\|_2^2 + \rho\|u\|_2^2 \geq 0, \quad \forall u.$$

Therefore, Q is positive semidefinite.

ii. In your code, write a function that takes in X and y, constructs Q as specified in the previous problem, and returns the condition number of Q. Report the condition number $\kappa(Q)$ for varying values of p and ρ, by filling in the following table. Here, $m = 742$ is the total number of data samples. Report at least 2 significant digits. Comment on how much ridge regression is needed to affect conditioning.

p	$\rho = 0$	$\rho = m$	$\rho = 10m$	$\rho = 100m$
1				
2				
5				
10				

Ans.

p	$\rho = 0$	$\rho = m$	$\rho = 10m$	$\rho = 100m$
1	3.25e+01	6.75e+00	1.69e+00	1.07e+00
2	1.48e+03	7.91e+01	9.20e+00	1.82e+00
5	7.31e+08	2.72e+05	2.72e+04	2.72e+03
10	7.16e+18	3.97e+11	3.97e+10	3.97e+09

In general, bigger p results in higher condition numbers. However, bigger ρ can reduce the condition numbers.

iii. Under the *same experimental parameters* as the previous question, run ridge regression for each choice of p and ρ, and fill in the table with the mean squared error of the fit:

$$\textbf{mean squared error} = \frac{1}{m}\sum_{i=1}^{m}(x_i^T\theta - y[i])^2$$

where x_i is the ith row of X. Comment on the tradeoff between using larger ρ to improve conditioning vs its affect on the final performance.

p	$\rho = 0$	$\rho = m$	$\rho = 10m$	$\rho = 100m$
1				
2				
5				
10				

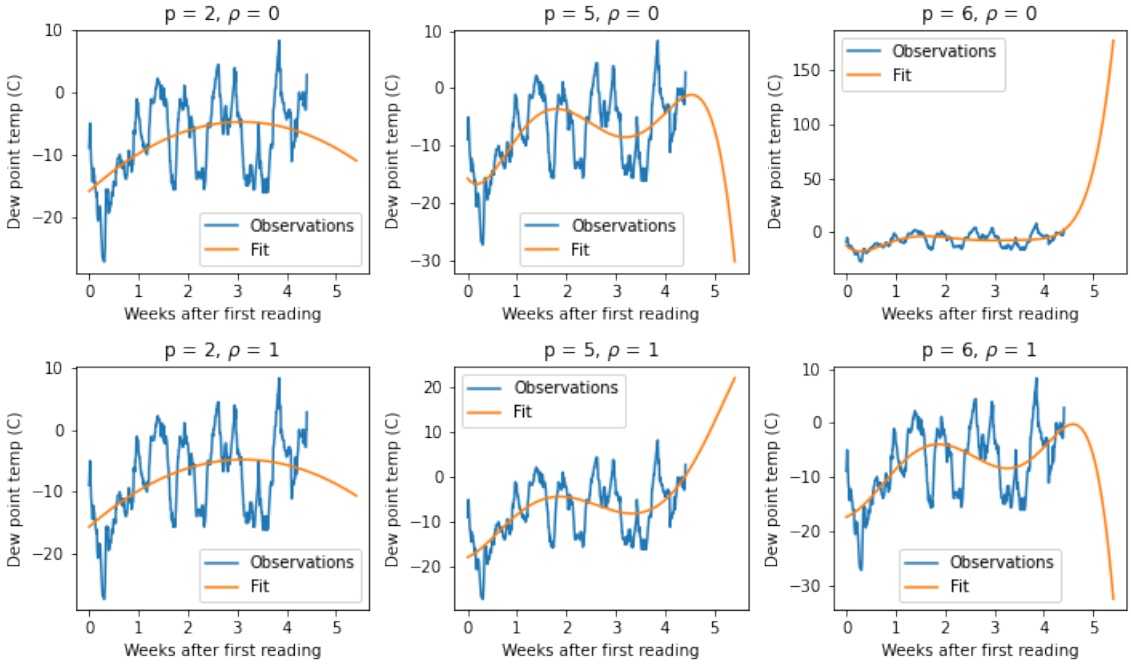
Ans.

p	$\rho = 0$	$\rho = m$	$\rho = 10m$	$\rho = 100m$
1	36.4	58.8	78.8	96.3
2	33.5	57.6	77.3	86.8
5	27.1	57.2	72.0	77.5
10	16.7	55.4	71.4	76.7

While larger ρ and smaller p give better condition numbers, it is clear that it comes as a hit in performance, as the MSE grows as well. Based on this table, I would pick $\rho = 0$ and $p = 2$ as a best fit choice, but there might be other reasons to choose otherwise.

(d) *Forcasting.* **(0.5pt)** Picking your favorite set of hyperparameters (p, ρ), forecast the next week's dew point temperature. Plot the forecasted data over the current observations. Do you believe your forecast? Why?

Ans. There are a number of possible solutions here. Here are some examples. (In general, for useful values of p, I did not notice much impact of ρ.)



There are definitely many cases where I don't believe the numbers, especially if $\rho = 0$ since the forcasting seems pretty wild. But, when $\rho = 1$ and larger values of p, the curves look plausible–but if I were a betting person I still wouldn't put down my mortgage!

You could reach two conclusions here: either polynomial fitting is not a great tool because there's no reason why dew temperature should follow a polynomial structure, or we could say that dew temperature is not that predictable, period, using only historical data.

Graders, any discussion here that is reasonable should get full credit.

6. **PAC learning. (2pts)** Consider the following hypothesis class in $\mathbb{R}^2$:

$$\mathcal{H} = \left\{ h_a : [-2,2]^2 \to \mathbb{R} : h_a(x) = \begin{cases} 1 & \text{if } |x[1] - x[2]| \le a \\ 0 & \text{else.} \end{cases}, \quad 0 \le a \le 1. \right\}$$

The notation $h_a : [-2,2]^2 \to \mathbb{R}$ means that the inputs x are restricted in the two-dimensional domain

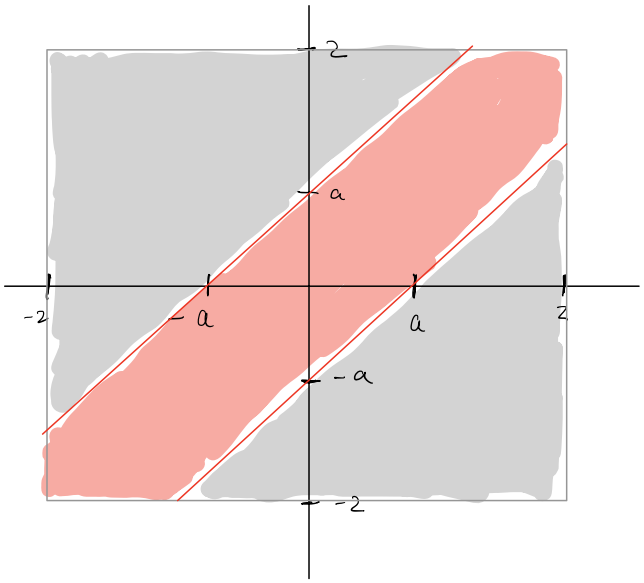
$$\begin{bmatrix} -2 \\ -2 \end{bmatrix} \le x \le \begin{bmatrix} 2 \\ 2 \end{bmatrix}.$$

We now consider a scenario where the true function $y = f(x)$ is *realizable*, e.g. $f \in \mathcal{H}$. We draw samples $\mathcal{S} = \{(x_1, y_1), ..., (x_m, y_m)\}$ where $y_i = f(x_i)$ and compute an ERM

$$h_{\mathcal{S}} = \underset{h \in \mathcal{H}_{\text{in}}}{\operatorname{argmin}} \, \mathcal{L}_{\mathcal{S}}(h)$$

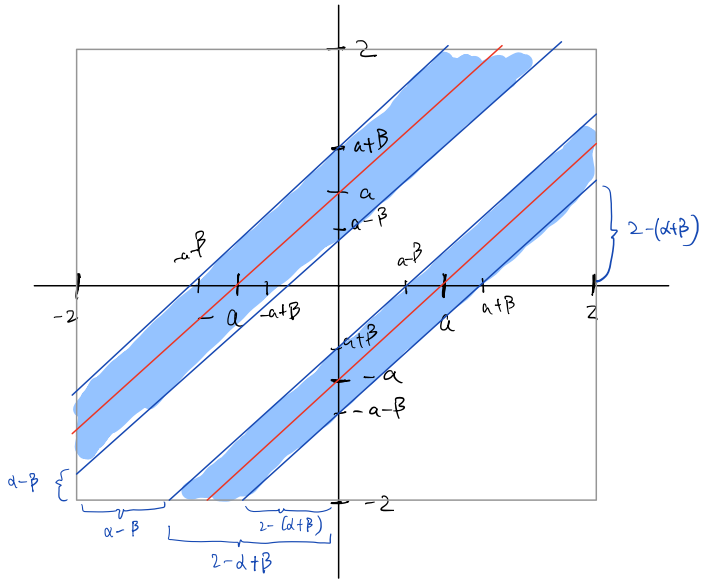
where $\mathcal{L}_{\mathcal{S}}$ is the empirical risk.

(a) **(0.25 pts)** Draw a picture of one possible hypothesis in $\mathcal{H}$. That is, draw the 2-D region where the area for x where $h_a(x) = 1$ is shaded, and $h_a(x) = 0$ is not shaded, for some plausible a.
Ans.



(b) **(0.25 pts)** Propose a training sampling strategy (e.g. a distribution $\mathcal{D}$ where we draw $x_i \sim \mathcal{D}$) that guarantees PAC learning.
Ans. Here, we consider a uniform sampling strategy over the domain $[-2,2]^2$.

(c) **(0.25 pts)** On the image above, indicate the region where no samples in $\mathcal{S}$ exist in order for the ERM estimate of $\hat{a}$ to be wrong by β, e.g. $|a - \hat{a}| = \beta$. Calculate the area of that region. (Your answer will be in terms of a.)
Ans.



To compute the area of the blue shaded, we take difference of triangles

$$\textbf{one band area} \;= \frac{(4-a+\beta)^2}{2} - \frac{(4-a-\beta)^2}{2} = 2\beta(4-a)$$

$$\textbf{two bands area} \;= 4\beta(4-a)$$

(d) **(0.25 pts)** Next, suppose that $\hat{a} = a + \beta$. What is $\mathcal{L}_{\mathcal{D}}(h_{\hat{a}})$? (Your answer will be in terms of a.)
Ans. The answer here is basically the fraction of the outer two blue slivers, over the total domain area. The area is

$$2 \cdot \frac{(4-a)^2 - (4-a-\beta)^2}{2} = 2\beta(4-a) - \beta^2$$

so

$$\mathcal{L}_{\mathcal{D}}(h_{\hat{a}}) = \frac{2\beta(4-a) - \beta^2}{16}$$

(e) **(0.25 pts)** Next, suppose that $\hat{a} = a - \beta$. What is $\mathcal{L}_{\mathcal{D}}(h_{\hat{a}})$? (Your answer will be in terms of a.)
Ans. The answer here is basically the fraction of the inner two blue slivers, over the total domain area. The area is

$$2 \cdot \frac{(4-a+\beta)^2 - (4-a)^2}{2} = 2\beta(4-a) + \beta^2$$

so

$$\mathcal{L}_{\mathcal{D}}(h_{\hat{a}}) = \frac{2\beta(4-a) + \beta^2}{16}$$

(f) **(0.75 pts)** Put the pieces together to prove that $\mathcal{H}$ is PAC-learnable by computing the number of samples m needed such that

$$\mathbf{Pr}(\mathcal{L}_{\mathcal{D}}(h_{\mathcal{S}}) \geq \epsilon) \leq \delta$$

for general $0 \leq (\delta, \epsilon) \leq 1$. At this point your answer should *not* depend on a, so you need to find the most extreme value of a such that your bound holds tight.

Hint: use $(1-x)^m \leq \exp(-xm)$

Ans. Note that this question reduces to finding the probability that a point lands in the error region, whose area we had previously computed. Given that the entire domain has area 16, a random point has a probability of $\frac{\beta(4-\beta)}{4}$ of landing in the error region.

For this to not happen with m i.i.d. chosen training samples, the probability is

$$\mathbf{Pr}(m \text{ points do not land in error region }) = (1 - \frac{\beta(4-a)}{4})^m \leq \exp(-m\beta(4-a)/4)$$

Now let's say β represents the largest possible value such that no point lands in this region. Then the worst possible guess of hypothesis class is either if $a - \hat{a} = 2\beta$ or $\hat{a} - a = 2\beta$. In either case, the error rate is exactly the size of the "bad region", normalized by the total area of 16.

Overall, this gives $\epsilon = \beta(4-a)/4$, so

$$\mathbf{Pr}(\mathcal{L}_{\mathcal{D}}(f_{\mathcal{S}}) \geq \beta(4-a)/4) \leq \exp(-m\beta(4-a)/4).$$

Now if I want to express this more generally in terms of ϵ, I take $\epsilon = \beta(4-a)/4 \iff \beta = \frac{4\epsilon}{4-a}$ and

$$\mathbf{Pr}(\mathcal{L}_{\mathcal{D}}(f_{\mathcal{S}}) \geq \epsilon \leq \exp(-m\epsilon) = \delta.$$

In other words,

$$m = \frac{1}{\epsilon}\log(\frac{1}{\delta})$$

is the sample complexity.