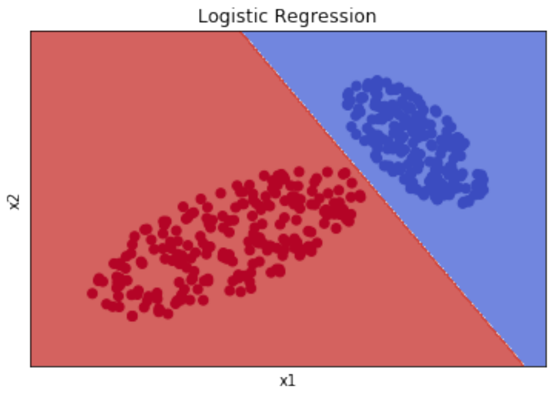
Eli Pinkus
CS 155
Set 2

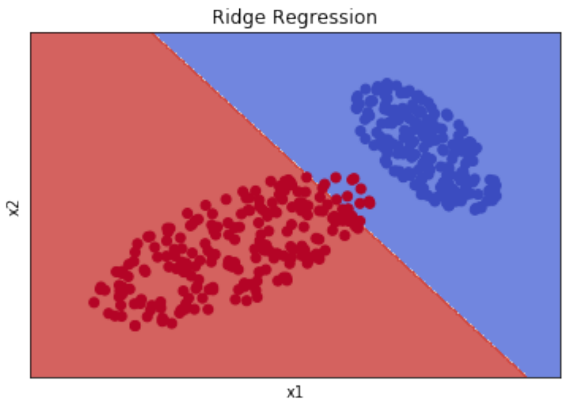
Problem 1

Part A

Least squares loss is best used to train a line to learn real values. This is because the loss is equivalent to absolute value squared of the difference from the predicted value to the actual value. As a result, loss increases faster for points where the prediction is further from the target. Thus, using it to find a classifier line can give bad results since various groupings can pull the 0 point of the line $w^T x$ away from the ideal divider in order to accommodate points that are far away from the boundary.

Part B





Logistic regression produces a better divider with a 0/1 loss of exactly 0. Ridge regression has a a handful of classification errors. This is because the red data points group in such a way that more of the points are a greater distance from where a proper divider would be. Thus, squared error penalizes the proper divider for being to far away from too many points for the reasons we have from part A, and thus the model returns a divider that is skewed further to the grouping of far away red points.

$$L_{\log} = \ln\left(1 + e^{-yw^T x}\right)$$

The exponent on the e goes as such:
If the classification is correct the sign on the exponent is negative and if the actual value, y, is large relative to $w^T x$, the exponent is a larger negative number which makes the e term smaller and the loss overall smaller. Thus, log loss doesn't overcompensate for certain correct classifications in the same way that squared loss might and the divider is better because of that.

Part C

$$S = \left\{\left(\frac{1}{2}, 3\right), (2, -2), (-3, 1)\right\}$$

Eli Pinkus
CS 155
Set 2

with

$$y = \{1,1,-1\}$$

and

$$w = (0,1,0)$$

We will add constant x_0 to all points for simplicity of notation.

for

$$x = \left(1, \frac{1}{2}, 3\right), \ \ y = 1$$

$$\nabla_w L_{\log}(x) = \frac{-yx}{1 + e^{yw^T(t)x}} = \frac{\left(-1, -\frac{1}{2}, -3\right)}{1 + e^{\frac{1}{2}}} \approx (-0.376, -0.189, -1.13)$$

$$\nabla_w L_{\text{hinge}}(x) = \begin{cases} 0, & yw^T x > 1 \\ -yx, & yw^T x < 1 \end{cases}$$

$$yw^T x = \frac{1}{2} \Rightarrow \nabla_w L_{\text{hinge}}(x) = -\left(1, \frac{1}{2}, 3\right)$$

for

$$x = (1,2,-2), \ \ y = 1$$

$$\nabla_w L_{\log}(x) = \frac{-yx}{1 + e^{yw^T(t)x}} = \frac{(-1,-2,+2)}{1 + e^2} \approx (-0.119, -0.238, 0.238)$$

$$\nabla_w L_{\text{hinge}}(x) = \begin{cases} 0, & yw^T x > 1 \\ -yx, & yw^T x < 1 \end{cases}$$

$$yw^T x = 2 \Rightarrow \nabla_w L_{\text{hinge}}(x) = 0$$

for

$$x = (1,-3,1), \ \ y = -1$$

$$\nabla_w L_{\log}(x) = \frac{-yx}{1 + e^{yw^T(t)x}} = \frac{(1,-3,1)}{1 + e^3} = (0.0474, -0.142, 0.0474)$$

Eli Pinkus
CS 155
Set 2

$$\nabla_w L_{\text{hinge}}(x) = \begin{cases} 0, & yw^T x > 1 \\ -yx, & yw^T x < 1 \end{cases}$$

$$yw^T x = 3 \Rightarrow \nabla_w L_{\text{hinge}}(x) = 0$$

Part D

The gradient of log loss will not converge to 0 because it is of the form:

$$\nabla_w L_{\log}(x) = \frac{-yx}{1 + e^{yw^T x}}$$

Which doesn't ever equal 0 unless $x = \vec{0} \ \ \forall x$

The gradient of hinge loss will converge to 0 if we arrive at some w s.t. $y_i w^T x_i > 1 \ \ \forall i$

Log loss only goes to zero as $y_i w^T x_i \to \infty \ \forall i$ which doesn't happen so log loss cannot be eliminated altogether.

Hinge loss can be eliminated if we arrive at some w s.t. $y_i w^T x_i > 1 \ \ \forall i$

Part E

For an individual point, hinge loss is 0 whenever it is classified correctly. Thus, the overall hinge loss is 0 anytime the points are divided correctly, regardless of margin. Thus, it is not enough to minimize L_{hinge} if you want to get a maximum margin classifier.

We know that the margin is given by $\frac{2}{||w||}$
So we would like to do some maximizing with respect to $\frac{2}{||w||}$. We can do this by minimizing $||w||$ and we know that

$$\arg\min_{w,b} ||w|| \equiv \arg\min_{w,b} ||w||^2$$

For this reason, we must also minimize with respect to $\lambda||w||^2$ for some λ in order to get not just a $E_{\text{in}} = 0$, but the maximum possible margin.

Eli Pinkus
CS 155
Set 2

Problem 2

Part A

Adding a regularizer is effectively limiting the flexibility of a model to fit training data, so adding a regularization penalty term can, and often does result in an increase in training error. Adding a penalty term often does, but need not result in a decrease in out of sample error. If a model is about the correct complexity to model the nature of the target function, and there is enough training data that the model can fit said nature, restricting the model complexity with a regularizer term could result in worse generalization behavior.

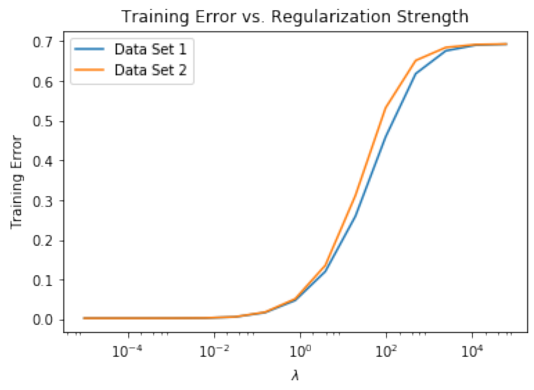
Part B

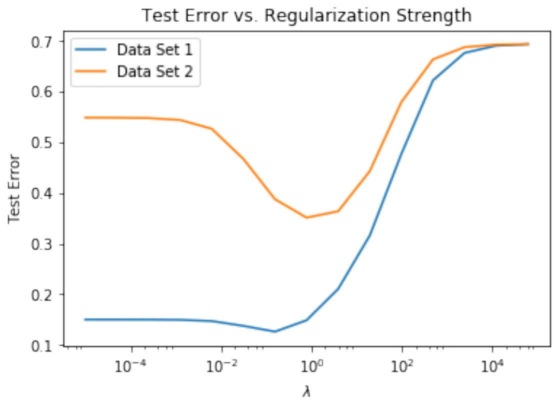
The ℓ_0 regularizer is:

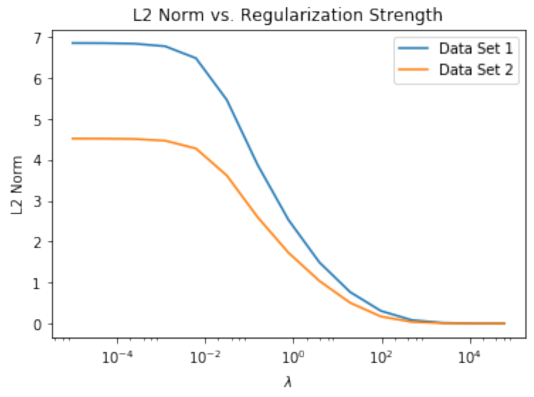
$$||w||_0 = \sum_d 1_{[w_d \neq 0]}$$

Which is either discontinuous or flat at any point which makes optimizing a term with ℓ_0 included a very difficult task

Part C







Eli Pinkus
CS 155
Set 2

Part D

Training error is fairly similar between the datasets relative to a changing λ. This makes sense because a stronger regularizer makes it more difficult for the model to perfectly fit any data set so having too strong of a regularizer would raise E_{in} in the same qualitative way for datasets of different sizes.

Test error is where the difference between the sizes of the data sets is noticeable. With fewer data points, an unregulized data set 2 is quite prone to overfitting and thus having poor generalization performance. This manifests in the graph as the E_{out} for data set 2 is far higher than that of set 1 when regularization strength is low. Data set 1 has additional data points and is therefore better off in terms of generalization by default and doesn't stand to gain as much from strong regularization. We can see this by in the graph because test error decreases less drastically when we approach the optimal amount of regularization.

Part E

Data set 2 overfits a lot with low regularization strength whereas data set1 only overfits a very little bit with small λ. This makes sense because set 2 is a proper subset of set 1 therefore we have more training examples in set 1 which allows it to better fit the nature of the target function without overfitting. Data set 2 reaches its best testing performance at around $\lambda = 1$ and Data set 1 reaches its best testing performance at around $\lambda = \frac{1}{10}$ but that best is only slightly better than the low regularization cases. This means that the data set 1 model wasn't overfitting too much. As λ increases further, both set 1 and set 2 models begin to under fit. This is because the regularizer becomes so restrictive that the data is being modeled well.

Part F

The ℓ_2 norm is strictly decreasing with increasing λ this is because a higher lambda imposes a higher penalty for large ℓ_2 norm so eventually that model is basically just minimizing ℓ_2 norm.

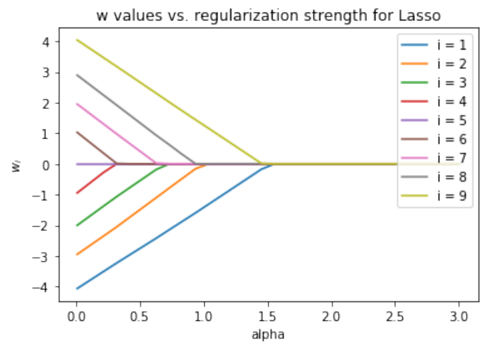
Part G

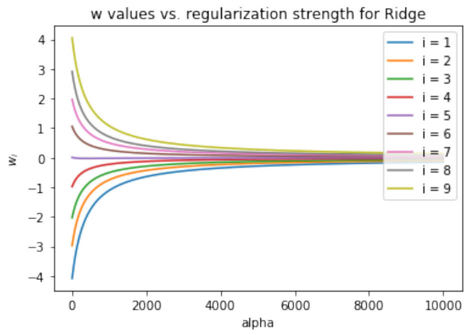
$\lambda = 1$ would be the best if we had to use only data set 2 because our graph indicates that this is close to the optimal tradeoff between model complexity and overfitting as indicated by it having the lowest testing error.

Eli Pinkus
CS 155
Set 2

Problem 3

Part A





As the regularization parameter increases, for lasso regression the weights approach 0 linearly and actually reach true 0. The number of 0 weights increases as regularization parameter increases. For Ridge, the weights approach 0 asymptotically, however they never actually reach 0.

Part B

i)

$$\arg\min_w \left|\left|y - Xw\right|\right|^2 + \lambda \left|\left|w\right|\right|_1$$
$$= \arg\min_w (y - Xw)^T(y - Xw) + \lambda w \arg\min_w [y^T y - 2wX^T y + w^2 X^T X + \lambda w]$$

We take the derivative noting that there are 3 cases that $\frac{\partial \left|\left|w\right|\right|_1}{\partial w}$ could take

When$\frac{\partial \left|\left|w\right|\right|_1}{\partial w} = 1$:

$$\frac{\partial L}{\partial w} = -2X^T y + 2wX^T X + \lambda$$

$$\Rightarrow w = \frac{2X^T y - \lambda}{2X^T X}$$

when $\frac{\partial \left|\left|w\right|\right|_1}{\partial w} = -1$:

$$\frac{\partial L}{\partial w} = -2X^T y + 2wX^T X - \lambda$$

$$\Rightarrow w = \frac{2X^T y + \lambda}{2X^T X}$$

when $\frac{\partial \left|\left|w\right|\right|_1}{\partial w} = n \in [-1,1]$

$$\frac{\partial L}{\partial w} = -2X^T y + 2wX^T X + n\lambda$$

$$\Rightarrow w = \frac{2X^Ty - \lambda n}{2X^TX}$$

ii)
So we consider the 3 cases numerators:
When$\frac{\partial ||w||_1}{\partial w} = 1$:
Numerator is: $2X^Ty - \lambda = 0 \Rightarrow \lambda = 2X^Ty$

when $\frac{\partial ||w||_1}{\partial w} = -1$:
Numerator: $2X^Ty + \lambda = 0 \Rightarrow \lambda = -2X^Ty$

when $\frac{\partial ||w||_1}{\partial w} = n \in [-1,1]$
Numerator: $2X^Ty - \lambda n = 0 \Rightarrow \frac{2X^Ty}{n}$
We know

$$|2X^Ty| \leq \left|\frac{2X^Ty}{n}\right| \quad \forall n \in [-1,1]$$

So, depending on the sign of X^Ty, one of the first two cases gives us our smallest 0 point at

$$\lambda = |2X^ty|$$

iii)

$$\arg\min_w ||y - Xw||^2 + \lambda\, ||w||^2 = \arg\min_w (y - Xw)^T(y - Xw) + \lambda ||w||^2$$

we take the derivative:

$$\frac{\partial L}{\partial w} = -2X^T(y - Xw) + 2\lambda w = -2X^Ty + 2X^TXw + 2\lambda I w = 0$$

$$-2X^Ty + (2X^TX + 2\lambda I)w = 0$$

$$\Rightarrow w = (X^TX + \lambda I)^{-1}X^Ty$$

were I is the identity matrix for $\dim w \times \dim w$ matrices.

iv)
We know that X^TX is positive semi-definite. Thus, X^TX has no negative eigenvalues.

We have that $\lambda = 0 \Rightarrow w_i \neq 0$
So, we have:

Eli Pinkus
CS 155
Set 2

$$(X^TX)^{-1}X^Ty \neq 0 \Rightarrow (X^TX)^{-1} \neq 0, \qquad X^Ty \neq \vec{0}$$

So for $w = (X^TX + \lambda I)^{-1}X^Ty = \vec{0}$ we would need $(X^TX + \lambda I)^{-1} = \mathbf{0}$

$X^TX + \lambda I$ has a zero determinant if and only if λ is the negative of an eigenvalue of X^TX
Since X^TX has only positive eigenvalues, λ would have to be negative for $X^TX + \lambda I$ to have a 0 determinant. In all other other cases $X^TX + \lambda I$ has a non zero determinant, so since we restrict ourselves to only $\lambda > 0$, we know all $X^TX + \lambda I$ have non zero determinants and thus their inverses are non-zero matrices. Thus, there is no λ for which $(X^TX + \lambda I)^{-1} = \mathbf{0}$.