# Problem 1

We have:

and and

We want

So, we solve:

So, the smallest of the options that satisfies the condition is C

# Problem 2

We have:

If we consider the signal of this model we have:

So the corresponding boundary in is given by

If :

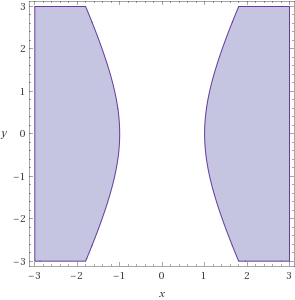
If

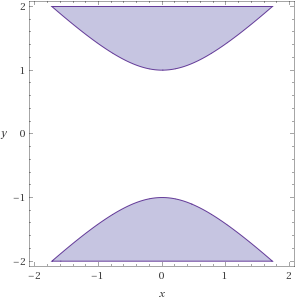
In order to get a hyberbola instead of an elipse (occurs when and are both positive) or no surface at all (occurs when and are both negative), we must have and opposite signs and have opposite signs.

We have 2 cases:

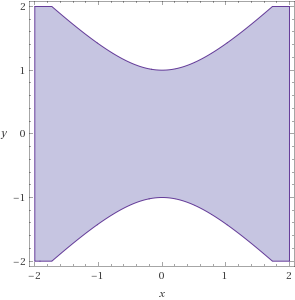
Either we have and in which case we have:

If we have: for which is this:

 Which is the opposite hypothesis to what we want

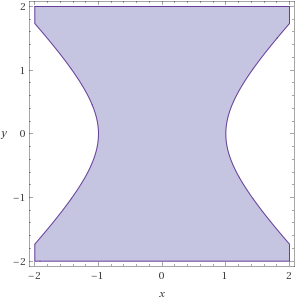
if we have: for which is this:which is wrong

or the other case is when and where we have:

if we have: for which is this:which is wrong

Finally, if we have and AND we have:

for which is this

 So the answer is D

# Problem 3

We know from lecture that after a non-linear transformation the new of the model becomes bounded by where is the dimensionality of the space.

In this case, the dimensionality of the space is 14 (excluding ).

Thus so 15 is the smallest value that isn’t less than the so the answer is C

# Problem 4

We have:

so taking the partial derivative we get:

So the answer is E

# Problem 5

We also have:

It took the simulation 10 iterations to fall below

So the answer is D

# Problem 6

After the simulation we have:

(0.04473629039778207, 0.023958714099141746)

Which is by far closest to E

# Problem 7

After the simulation we have

.13981379199615324 so the answer is A

# Problem 8

The simulation gave average approx. so the answer is D

# Problem 9

The simulation gave average epochs needed

So the answer is A

# Problem 10

We know that C is the error function for linear regression which is different from PLA

We know that D is the cross entropy error function for logistic regression which is different from PLA.

B has a similar gradient updating form as required for PLA however in PLA we only update when classification is incorrect and in this case we update every time. So, it’s wrong.

A would have update rule:

which does not agree with PLA.

E has two distinct cases that collectively agree with PLA. If i.e. The signs of and the signal are the same and we have a properly classified point, than which agrees with PLA. So the update rule of stochastic gradient descent in this case is:

as desired

In the case where , the classification was wrong and we have:

So, we would have: so the update rule for stochastic gradient descent is:

so, for we have the update rule for PLA.

So, the answer is E