6.036: Midterm, Fall 2018

Solutions

- This is a closed book exam. Calculators not permitted.
- The problems are not necessarily in any order of difficulty.
- Record all your answers in the places provided. If you run out of room for an answer, continue on a blank page and mark it clearly.
- If a question seems vague or under-specified to you, make an assumption, write it down, and solve the problem given your assumption.
- If you absolutely *have* to ask a question, come to the front.
- **Write your name on every page.**

Name: _____ Athena ID: _____

Question	Points	Score
1	12	
2	8	
3	10	
4	16	
5	12	
6	16	
7	14	
8	12	
Total:	100	

Name: _____

Originality

1. (12 points) Consider the following classification data set with two-dimensional inputs: there is one positive example at (1, 0) and one negative example at (2, 0).

   (a) Is this data linearly separable *through the origin*? ◯ yes √ **no**

   (b) If so, give the parameters θ of a separator.

   Solution: Nope.

   (c) Describe the sequence of hypotheses the perceptron algorithm *through origin* generates on this data set, assuming the positive example is chosen first.

   Solution:

```
Start:  0, 0
After error on positive example:  1, 0
After error on negative example: -1, 0
After error on positive example: 0, 0
After error on negative example: -2, 0
After error on positive example: -1, 0
After error on positive example: 0, 0
```

   And it keeps cycling between (-2, 0), (-1, 0), (0, 0).

   (d) Is this data linearly separable, but *not* through the origin? √ **yes** ◯ no

   (e) If so, give the parameters θ and θ_0 of a separator.

   Solution: $\theta = (-1, 0)$ $\theta_0 = 3/2$
   There are many other solutions.

   (f) Without simulating the execution, what specific thing can you say about the value of θ_2 when the perceptron algorithm terminates?

   Solution: It is 0.

Name: _____

All Greek to me!

2. (8 points) Let's consider solving a ridge regression problem using stochastic gradient descent. For simplicity, we will ignore the offset. Our hypothesis has the form

$$h(x;\theta) = \theta^T x \ ;$$

our objective function has the form

$$J(\theta) = \left(\frac{1}{n} \sum_{i=1}^{n} \left(h(x^{(i)};\theta) - y^{(i)} \right)^2 \right) + \lambda \|\theta\|^2 \ ;$$

and we will do T steps of gradient descent using a rule of the form

$$\theta = \theta - \eta \nabla_\theta J(\theta) \ ,$$

where η has a fixed value throughout the execution.

What is with all these Greek letters!? Each of θ, λ, and η has a role in what happens.

In the following questions, mark all answers that apply.

(a) Which parameter(s) would be included when using the hypothesis to make predictions?
√ θ ◯ λ ◯ η ◯ none

(b) Which parameter(s) are primarily intended to improve generalization?
◯ θ √ λ ◯ η ◯ none

(c) Can T play a similar role to λ ?
√ **yes** ◯ no
Explain your answer.

> **Solution:** By stopping the optimization early, we force the algorithm to use less of the training data, hence we can prevent it from being overly specialized (overfit) to the training data.

(d) Can η play a similar role to λ ?
√ **yes** ◯ no
Explain your answer.

> **Solution:** Using a very small step size has a behavior somewhat like stopping early; using a large step size (or not decaying it appropriately) will also not allow the optimization to go into a "narrow valley," which might also help to prevent overfitting.

Name: _____

Margin

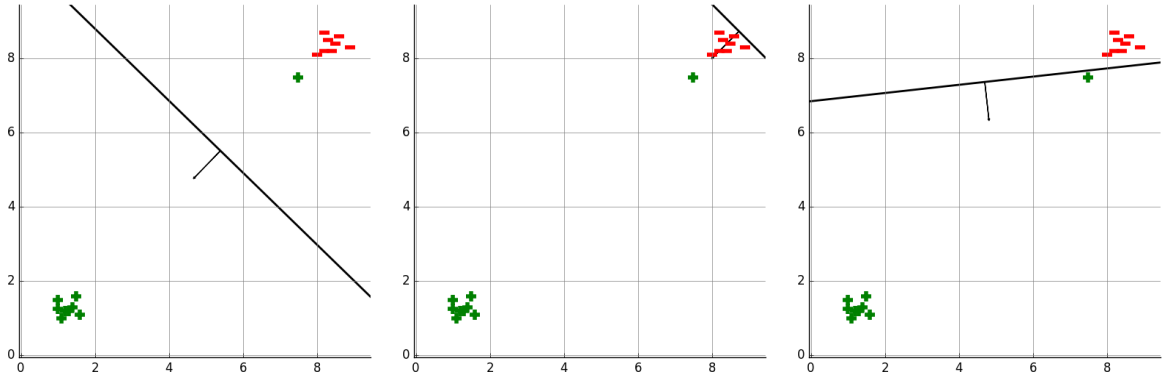
3. (10 points) We trained several classifiers using the SVM objective function,

$$J(\theta, \theta_0) = \left(\frac{1}{n} \sum_{i=1}^{n} \max\left(0, 1 - y^{(i)}(\theta^T x + \theta_0)\right) \right) + \lambda \|\theta\|^2 \ ,$$

while varying the parameter λ.

In all the plots below, we are showing a unit normal to the separator; it does not reflect the magnitude of θ.

(a) **Data set 1:** In the following examples, we show the resulting separator. Label them with the λ values from the set $(10^{-7}, 0.1, 5.0)$. Note that there is a single positive example up near the cloud of negative examples.



0.1 **5** 10^{-7}

Solution: Let's start by thinking of what "goal" each of the two terms in SVM objective J represents.

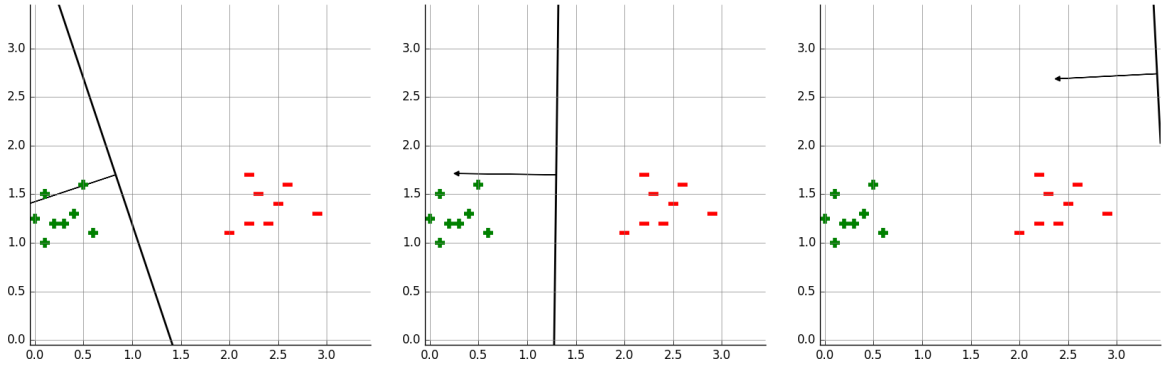
- $\left(\frac{1}{n} \sum_{i=1}^{n} \max\left(0, 1 - y^{(i)}(\theta^T x + \theta_0)\right)\right)$ represents the sum of the hinge-loss for all data points, which is the sum of the penalties for data misclassified or within γ_{ref} of the boundary, which is generally interpretable as the correctness of the classification;
- (minimizing) $\|\theta\|^2$ represents the size of the margin (which is quantitatively $\frac{1}{||\theta||}$, and qualitatively the minimum, **out of all points**, of the **signed** distance from each point the boundary.)
- λ is the trade-off between these two goals.

Now let's solve the problem.

- If λ is almost nonexistent, J is almost entirely about the first goal (classification), SVM will find a boundary that fulfills its first goal (correctly classifies all the points); within the set of classifiers that make no mistakes, it will prefer one that maximizes the margin. This yields **Plot 3**. (Notice that the green outlier is correctly classified.)
- If λ is not nonexistent but not too big, J represents both goals. SVM will find a boundary that trades off both parts of the objective, yielding **Plot 1** . (Notice that the green outlier is incorrectly classified, and that the margins for most points, on average, is large.)

- If λ is big, the SVM objective J becomes more about minimizing $||\theta||$ and not so much about its original intended objective, correct classification. SVM will now find some boundary that makes $||\theta||$ small and ignores classification, yielding **Plot 2** .

(b) **Data set 2:** In the following examples, we show the resulting separator. Label them with the λ values from the set $(0, 0.1, 2.0)$



0 **0.1** **2.0**

Solution: We can use logic similar to the previous part. With $\lambda = 0$, we will accept any separator with no regard for the margin.

(c) Approximately how many mistakes would you expect the perceptron algorithm to make on data set 1? ◯ 10 ◯ 100 √ **2500** ◯ 25,000

Solution: Remember that an upper bound on the number of mistakes made by the perceptron is given by $(\frac{R}{\gamma})^2$ where R is the data's maximum distance from the origin, and γ is the margin of the data to **a perfectly linearly classified boundary**(quantitative interpretation of condition (a) of Theorem 3.1, See Theorem 3.1 of lecture note.) Look at the scales on the axes carefully and note that $R \approx 10$, $\gamma \approx 0.2$. Thus,

$$\left(\frac{R}{\gamma}\right)^2 \approx 2500 \text{ mistakes}$$

(d) Approximately how many mistakes would you expect the perceptron algorithm to make on data set 2? √ **10** ◯ 100 ◯ 2500 ◯ 25,000

Solution: Look at the scales on the axes carefully and note that $R \approx 3$, $\gamma \approx 0.75$. Thus,

$$\left(\frac{R}{\gamma}\right)^2 \approx 16 \text{ mistakes}$$

Name: _____

Whither the weather?

5. (12 points) You have been hired for your mad ML skillz to do weather predictions for the Boston Glob newspaper. You need to predict precipitation type and minimum temperature for the day, based on a large amount of atmospheric data. You decide to use a neural network, but need to define several things related to the output.

   (a) Assume that precipitation type can be one of *None*, *Rain*, or *Snow* and that temperature can be treated as a real number of degrees Celsius.

   Select a way to encode the output into a fixed number of output dimensions and show what values of those outputs would you use to encode that it will snow and be -4 degrees C.

   Solution: (0, 0, 1, -4). The first three features, i.e. *None*, *Rain*, or *Snow*, are one-hot encoded. The fourth feature is taken as is (could also be normalized).

   (b) Explain what activation functions you would use in the last layer of the network. Unlike most of the examples we have discussed, it might be appropriate in this case to use different activation functions for different units in the output layer.

   Solution: Softmax on the first three outputs and linear on the last one.

   If you assume (or are told) that temperatures were normalized (which is not the case here), you could also have used a hyperbolic tangent.

   (c) Assume that assigning a probability of 0.5 to the correct precipitation type is as bad as being off by 10 degrees C in temperature. Write a loss function in the form $L(\text{guess}, \text{actual})$ to express an appropriate loss for this network.

   Solution:

$$L(g, a) = \sum_{j=1}^{3} -a_j \log g_j + \alpha(g_4 - a_4)^2$$

   We have to set α to get the trade-off right. If we are off by 10 degrees, then the second term will have value 100.

   Our cost of assigning prob 0.5 to the right precipitation type is $-\log 0.5$. To make these equal, $-\log 0.5 = 100\alpha$, so $\alpha = -(\log 0.5)/100$.

Name: _____

Adversarial example

6. (16 points) Willy Makeit has trained up a one-layer neural network with a sigmoid activation function to classify ferns based on several important features of their leaves. The hypothesis is:

$$h(x; W, W_0) = \sigma(W^T x + W_0) \quad .$$

Willy is particularly excited to find that this network correctly classifies an important but unusual-looking species of fern (which we will call x^*) as a positive example.

Betty Wont wants to defeat Willy's classifier by finding another fern, x_A, that is very similar to x^* but which his classifier predicts is negative.

The problem Betty wants to solve can be framed as finding a new input $x_A = \arg\min_x J(x)$, where

$$J(x) = \alpha \|x - x^*\|^2 + \max(0, h(x; W, W_0) - 0.5) \quad .$$

(a) Which term in the objective J depends on the class of x? Explain in words what it is computing and why it makes sense in this problem.

> **Solution:** The second term depends on the class of x. If the class of x is negative, Willy's classifier h outputs a value < 0.5, in which case, the second term evaluates to 0. Otherwise, for x of positive class, h outputs a value > 0.5, and the second term is this output minus 0.5, a positive value. Thus, this term will penalize positive predictions, with a greater loss the farther these predictions are from 0.5. However, once the prediction goes below 0.5, Betty is happy since the fern is now misclassified, so the loss goes to 0.
>
> The first term has no depency on the class of x, only on the distance between x and x^*.

(b) Betty thinks gradient descent would be a good way to solve this problem. If $x \in \mathbb{R}^d$, what are the dimensions of $\nabla_x J(x)$?

By spring23 convention this would've been d by 1

> **Solution:** $1 \times d$. Since we are taking the gradient of a scalar with respect to a d-dimensional vector x, the shape of the output should match the shape of x.

(c) Write an expression for

$$\nabla_x J(x)$$

in terms of W, W_0, and x. Recall that $\sigma(z) = \frac{e^z}{e^z+1}$ and $\sigma'(z) = \sigma(z)(1 - \sigma(z))$.

> **Solution:** We take the gradient of each term separately:
>
> $$J(x) = J_1(x) + J_2(x),$$
>
> where $J_1(x) = \alpha\|x - x^*\|^2 = \alpha(x - x^*)^T(x - x^*)$ and $J_2(x) = \max(0, h(x; W, W_0) - 0.5)$. We have
>
> $$\nabla_x J_1(x) = 2\alpha(x - x^*).$$

If $h(x; W, W_0) \leq 0.5$, this term evaluates to 0 so the gradient is the 0 vector (with the shape of x). Otherwise, applying the chain rule,

$$\begin{aligned}\nabla_x(h(x; W, W_0) - 0.5) &= \sigma(W^T x + W_0)(1 - \sigma(W^T x + W_0))\nabla_x(W^T x + W_0) \\ &= \sigma(W^T x + W_0)(1 - \sigma(W^T x + W_0))W \\ &= h(x; W, W_0)(1 - h(x; W, W_0)W\end{aligned}$$

Thus, we have

$$\nabla_x J(x) = 2\alpha(x - x^*) + \begin{cases} 0 & \text{if } h(x; W, W_0) \leq 0.5 \\ h(x; W, W_0)(1 - h(x; W, W_0))W & \text{otherwise.} \end{cases}$$

(d) If Betty sets α to a very *small* value and finds $x_A = \arg\min_x J(x)$, is it likely that she will have succeeded in finding a plant similar to x^* that is classified as negative? Explain why or why not.

Solution: Yes, surprisingly. (Almost all of the staff got this one wrong).
When α is positive but very small, then Betty will find the plant that is closest to x^*, but classified as negative. This is because, for any negative example, the second term is 0. Then, within the space of x's that are negative, the first term will push the solution as close as possible to x^*.

(e) If Betty sets α to a very *large* value and finds $x_A = \arg\min_x J(x)$, is it likely that she will have succeeded in finding a plant similar to x^* that is classified as negative? Explain why or why not.

Solution: No. Here, the stress of the objective function shifts to having x^A be close to x^*, at the expense of having x^A get misclassified. Thus, she'll likely get something very similar to x^* that does not receive a negative classification.

Name: _____

Trigonometric basis

7. (14 points) We can define a non-linear feature transformation using trigonometric functions. In this question, we will consider only the case in which our original $x^{(i)}$ are in $\mathbb{R}$ (that is, the input dimension $d = 1$.)

   Define the kth order trigonometric basis feature transformation to be

   $$\phi(x) = (1, \sin(x), \cos(x), \sin(2x), \cos(2x), \sin(x/2), \cos(x/2), \ldots, \sin(kx), \cos(kx), \sin(x/k), \cos(x/k)) \quad .$$

   (a) Sindy thinks that this basis is missing an important aspect, and suggests that it would be useful to add to the feature vector components of the form

   $$j \sin(x), j \cos(x)$$

   for values of j from 2 to k.

   Cosima thinks that Sindy's suggestion won't add any expressive power (that is, that any function that could be represented using Sindy's basis can also be represented using the original one.)

   Who is right? √ **Cosima** ◯ Sindy

   Let h be a hypothesis written in terms of Sindy's basis of order 2:

   $$\begin{aligned} h(x) =& \theta_0 + \theta_1 \sin(x) + \theta_2 \cos(x) + \\ & \theta_3 \sin(2x) + \theta_4 \cos(2x) + \theta_5 \sin(x/2) + \theta_6 \cos(x/2) + \theta_7 2 \sin(x) + \theta_8 2 \cos(x) \quad . \end{aligned}$$

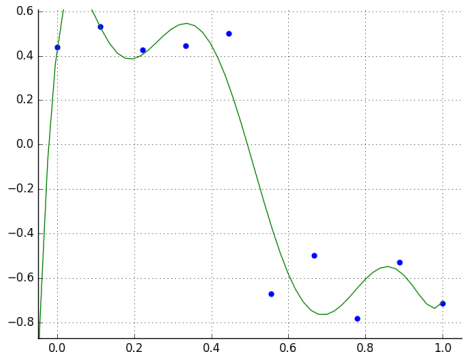
   If Cosima is right, show how to describe this hypothesis in terms of the original trigonometric basis of order 2, using parameters $\theta_0, \ldots, \theta_8$.

   Solution:

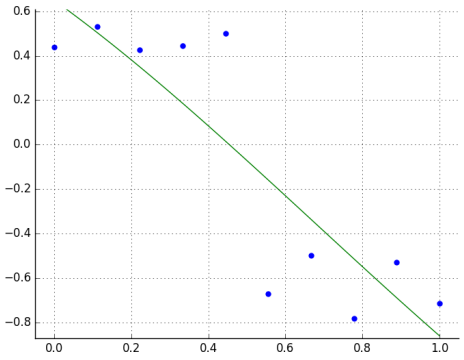
   $$\begin{aligned} h(x) =& \theta_0 + (\theta_1 + 2\theta_7) \sin(x) + (\theta_2 + 2\theta_8) \cos(x) + \\ & \theta_3 \sin(2x) + \theta_4 \cos(2x) + \theta_5 \sin(x/2) + \theta_6 \cos(x/2) \end{aligned}$$

   (b) We used the trigonometric basis, for several different values of k to transform the input to a new feature space, performed linear regression, and obtained the following plots of $h(x)$ versus x. For each plot, provide the correct k value, chosen from the set $0, 1, 2, 3, 7$.

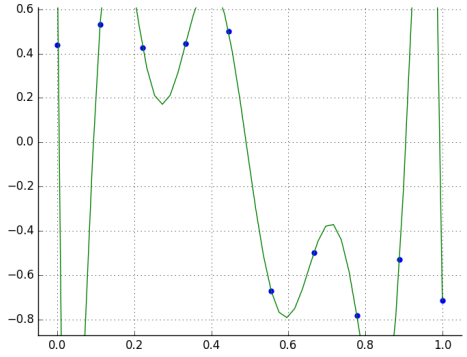
Name: _____



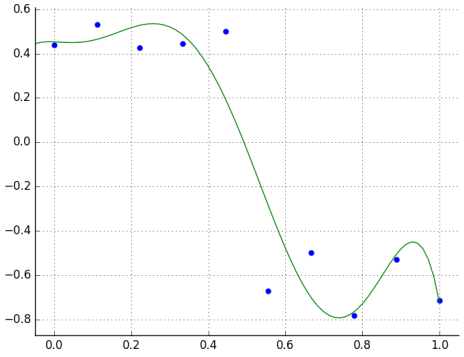
(A) k: **3**



(B) k: **1**



(C) k: **7**



(D) k: **2**

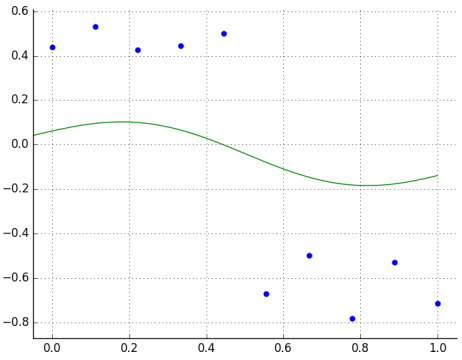
(c) Which hypotheses from the plots above do you think would lead to the best test-set performance on the range of x values between 0 and 1 (the range that is plotted)?

√ **A** ◯ B ◯ C √ **D**

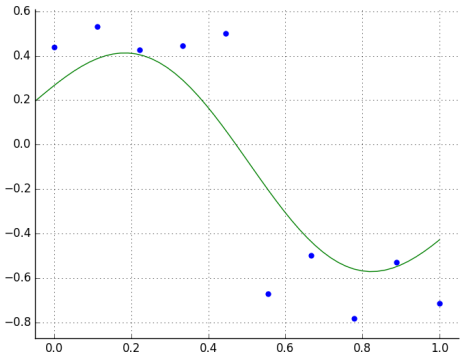
We accepted A, D, or both.

(d) We also used the trigonometric basis on the same data with a large fixed value of k, but performed ridge regression with various values of regularization parameter λ. For each plot, provide the correct λ value, chosen from the set $(0.0, 10^{-7}, 1e1, 1e2)$

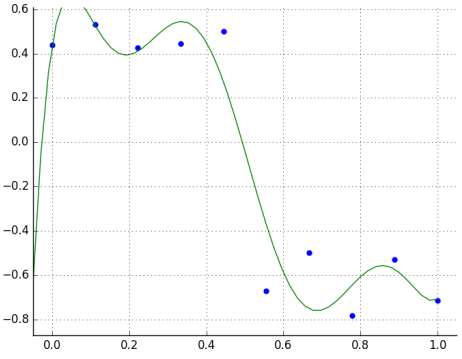
Name: _____



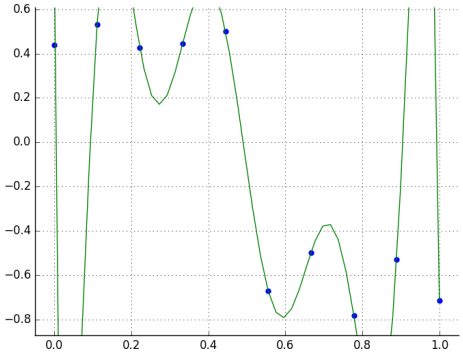
λ: **1e2**



λ: **1e1**



λ: **1e-7**



λ: **0.0**

Name: _____

Costis and Sinclair

8. (12 points) Costis thinks that using $f(x) = \cos(x)$ as an activation function for hidden layers in a neural network would be a good idea. Sinclair thinks using $\cos(x)$ as an output-layer activation function could be useful, depending on what the target values in the training data and the loss function are.

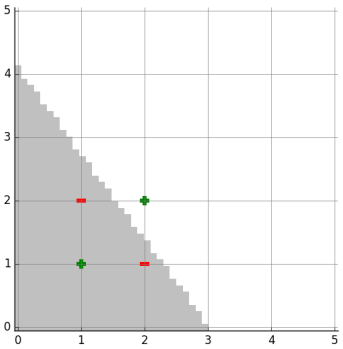
   Sinclair and Costis trained up 4 different neural networks using a combination of structures. In each case the inputs were points in two dimensions, with no feature transformation. In every case, they used NLL as the loss function and the y values in the data set were either 0 or 1. Assume that in each case the training arrived at a globally optimal solution.

   Here are hypotheses generated by some of their networks. *The darker areas correspond to network outputs less than 0.5.*

   Provide a selection of each of the attributes of a network with the shown optimal solution. Do not provide number of units and hidden layer activation if you select 0 as the number of hidden units. *There may be multiple correct answers; you just need to provide one.*

   - Number of hidden layers: 0 or 1
   - Number of units in hidden layer: 1 or 2 (if hidden layer)
   - Hidden-layer activation function: linear, cos, relu
   - Output activation function: linear, cos, sigmoid

Name: _____

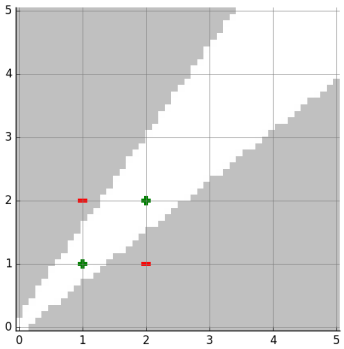


n hidden layers: **0**

n units: **-**

f hidden: **-**

f output: **sig or relu or linear**

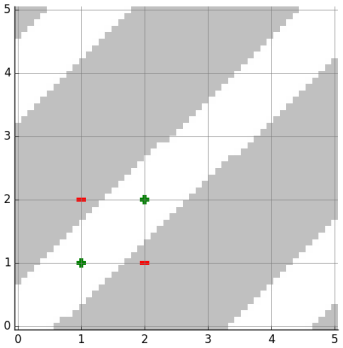


n hidden layers: **1**

n units: **2**

f hidden: **relu or sig**

f output: **sig or relu or linear**

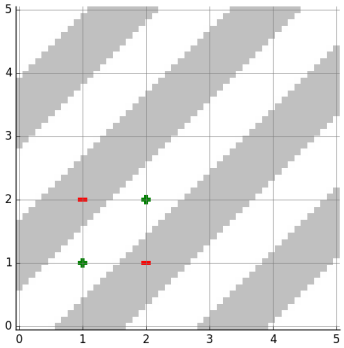


n hidden layers: **0**

n units: **-**

f hidden: **-**

f output: **cos**



n hidden layers: **1**

n units: **2**

f hidden: **cos**

f output: **sig or relu or linear**

The most disappointing error of all was including a hidden layer with linear units.

Note that with a cos on the output, we'll get asymmetric stripes because we're cutting the cos off at 0.5.

Some people tried to do the last parts with a large number of relu units the hidden layer—-that's

possible, but we only asked for 1 or 2 hidden units.

In fact, the only really sensible output unit is sigmoid, since we're training with targets of 0 and 1.

You need two hidden units to do the second one.