6.867: Exam 1, Fall 2016

Solutions

These are not the **only** acceptable answers. Some other answers also received credit.

Answer the questions in the spaces provided. Show your work neatly. **We will only grade answers that appear in the answer boxes.**

If a question seems vague or under-specified to you, make an assumption, write it down, and solve the problem given your assumption.

You may use electronic devices (laptop, calculator, phone) *only if they are in "airplane mode" with all cell and network connections disabled.*

Write your name on every page.

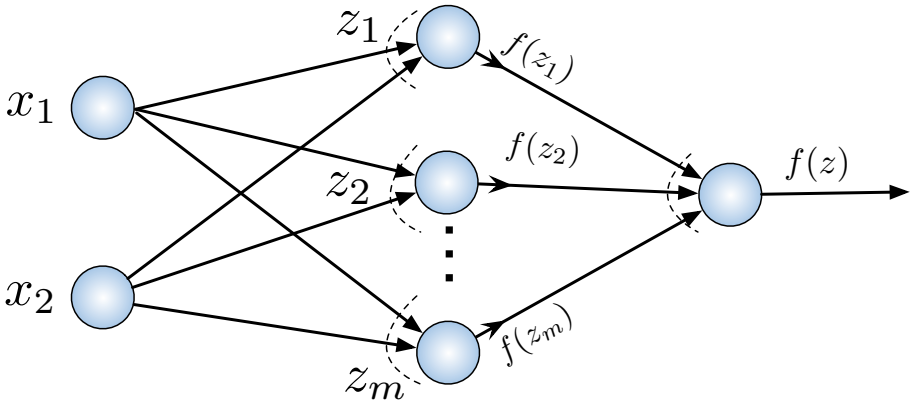
Come to the front if you need to ask a question.

Name: _____ MIT Email: _____

Question	Points	Score
1	8	
2	16	
3	12	
4	18	
5	18	
6	10	
7	18	
Total:	100	

Name: _____

Shallow neural network



1. (8 points)

We consider classifying points on the plane using a neural net with one hidden layer as shown above. All hidden units are ReLUs.

(a) Suppose that the y values in the training data are elements of $\{-1, 1\}$ and we wish to make predictions that minimize the hinge-loss. Mark all activation functions that are appropriate for the output layer.

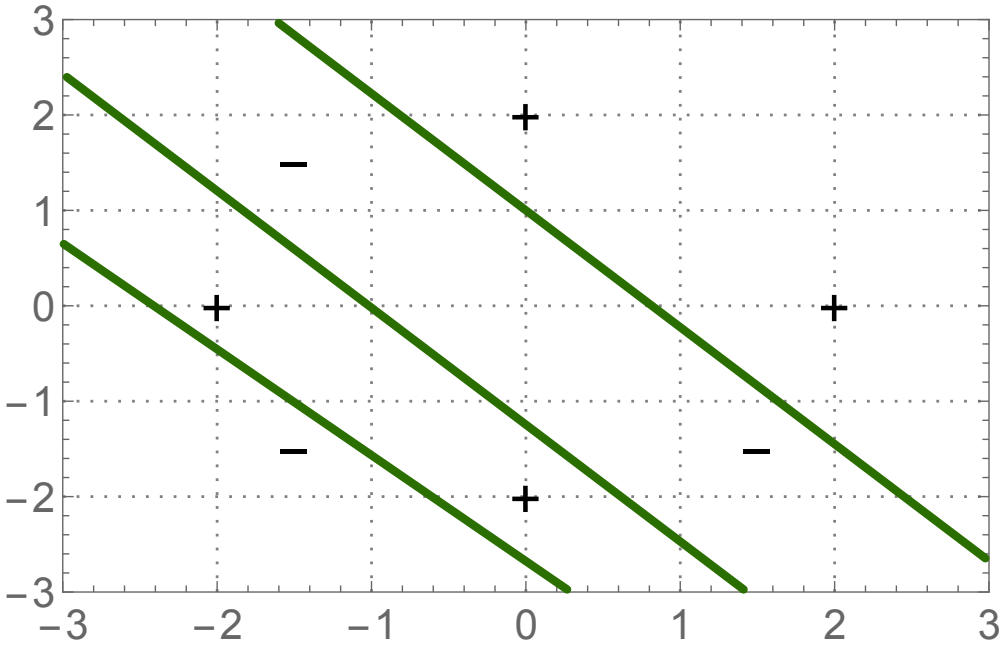
◯ sigmoid √ **tanh** ◯ ReLU √ **linear**

(b) Consider the positively and negatively labeled points shown below: Suppose we use our neural net specified above to classify these points (that are labeled $\{-1, 1\}$). What is the smallest number of hidden ReLUs that suffice to correctly separate the positive from the negative points?

Solution: 3

(c) For each of the m (where m is your answer to the previous question) hidden ReLUs in a network that correctly classifies the data, draw the boundary at which its output transitions from 0 to positive. (Note that these boundaries are not unique).

Name: _____

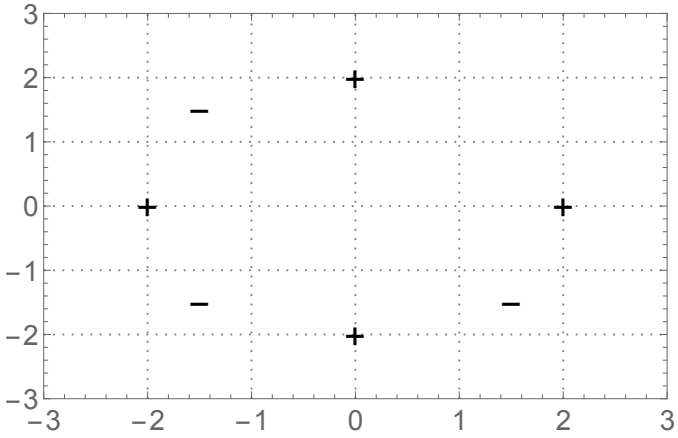


Name: _____

SVM

2. (16 points) We consider the behavior of the support vector machine (SVM) classifier.

   (a) Consider the 7 points shown below (same as the figure in the previous question):



   Which of the following kernels can correctly separate the training data (mark all that apply):

   ◯ $k(x, x') = (1 + x \cdot x')$
   √ $k(x, x') = (1 + x \cdot x')^2$
   √ $k(x, x') = \exp(-\|x - x'\|^2)$

   (b) The Hard-SVM constraint is $y^{(i)}(w^T x^{(i)} + b) \geqslant 1$. If we replace the 1 by an arbitrary number $\gamma > 0$, does it change the optimal margin hyperplane?
   ◯ Yes √ **No**

   (c) For the C-SVM, if $\alpha_j = C$, is the corresponding training data point always a support vector that lies on the margin.
   ◯ Yes √ **No**

   (d) If we run a C-SVM on linearly separable data, does it yield the same hyperplane as the Hard-SVM.
   ◯ Yes √ **No**

(e) Consider the soft-margin SVM with primal objective $\frac{1}{2}\|w\|^2 + C\sum_i \xi_i^2$, where we have penalized slacks via $C\sum_i \xi_i^2$ (instead of the usual $C\sum_i \xi_i$). In this formulation of the soft-margin SVM, can we omit the constraint $\xi_i \geqslant 0$?

$\surd$ **Yes** ◯ No

Provide a brief explanation.

> **Solution:** The constraint $\xi_i \geqslant 0$ can be dropped. Without going through KKT conditions, this can be seen directly. We have constraints of the form $y^{(i)}(w^T x^{(i)} + b) \geqslant 1 - \xi_i$. If any ξ_i is negative, this constraint gets even harder to fulfill. If any x_i is negative, by merely setting it to zero we still retain feasibility and lower the objective, hence no negative ξ_i values can ever be optimal for the problem, and we can safely drop the $\xi_i \geqslant 0$ constraint.

(f) Using the same primal objective as in (e), is the optimal predictor w also of the form $w = \sum_i \alpha_i y^{(i)} \phi(x^{(i)})$?

$\surd$ **Yes** ◯ No

Provide a brief explanation.

> **Solution:** The optimal predictor is of the claimed form. This follows from the representer theorem. It also follows upon setting the derivative of the Lagrangian wrt w to zero and solving for w.

Solution:

(a) Only 3 really works. If you use the coordinates on the plot literally, then 2 works, but only because the aspect ratio distorts the coordinates. The quadratic kernel will not separate them if they had been drawn to lie on a circle.

(b) No it does not change the hyperplane (just rescales w and b)

(c) No. While the data point is a support vector, it can be a margin error and need not lie on the supporting hyperplane.

(d) No, the C-SVM may still have a different solution (outliers hurt the hard-margin case more)

(e) The constraint $\xi_i \geqslant 0$ can be dropped. Without going through KKT conditions, this can be seen directly. We have constraints of the form $y^{(i)}(w^T x^{(i)} + b) \geqslant 1 - \xi_i$. If any ξ_i is negative, this constraint gets even harder to fulfill. If any x_i is negative, by merely setting it to zero we still retain feasibility and lower the objective, hence no negative ξ_i values can ever be optimal for the problem, and we can safely drop the $\xi_i \geqslant 0$ constraint.

(f) The optimal predictor is of the claimed form. This follows from the representer theorem. It also follows upon setting the derivative of the Lagrangian wrt w to zero and solving for w.

Name: _____

Respecting boundaries

3. (12 points) *Question credit: Tommi Jaakkola*

We estimate an SVM using training examples. After trying a few choices of SVMs, we obtained 6 different plots of the decision boundary. We have visualized the corresponding hyperplanes in feature space as nonlinear separators in the original space. The solid line corresponds to the separating hyperplane (i.e., points for which $\sum_i \alpha_i y^{(i)} k(x^{(i)}, x) + b = 0$), while the dashed lines correspond to the supporting hyperplanes (i.e., points for which $\sum_i \alpha_i y^{(i)} k(x^{(i)}, x) + b \in \{\pm 1\}$). Each of the plots was obtained by solving the dual formulation of the SVM subject to different constraints on the dual variables α and different choice of kernel. Each resulting SVM classifier corresponds to a different choices of kernel and constraints:

(K1) $k_1(x, x') = (1 + x \cdot x')$,

(K2) $k_2(x, x') = (1 + x \cdot x')^2$,

(K3) $k_3(x, x') = (1 + x \cdot x')^3$,

(Kg) $k_g(x, x') = \exp(-\|x - x'\|^2)$

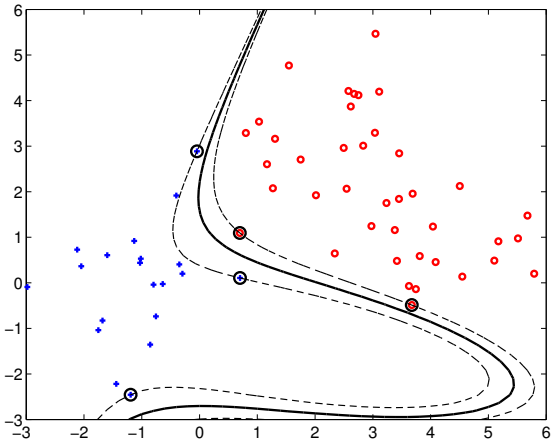
(C1) $0 \leqslant \alpha_i \leqslant 0.1,\ i = 1, \ldots, n$

(C2) $\alpha_i \geqslant 0,\ i = 1, \ldots, n.$

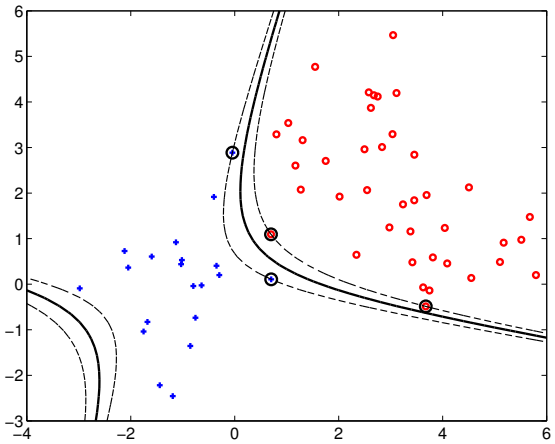
Under each figure, select **one** kernel and **one** constraint specifying which combination most likely yielded the plot. Each (kernel, constraint) pair can be assigned to **at most one plot** (the support vectors are circled).

Figures are on the next page

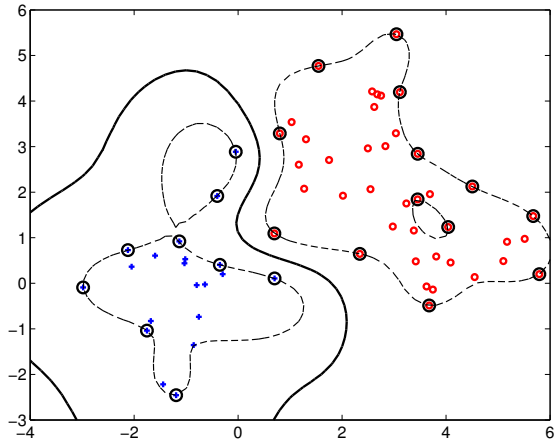
Name: _____



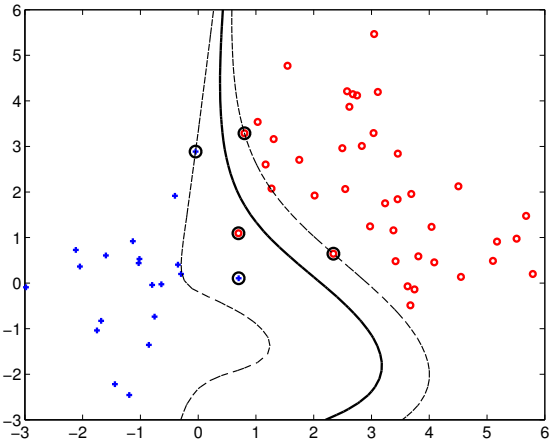
○ K1 ○ K2 √ **K3** ○ Kg
○ C1 √ **C2**



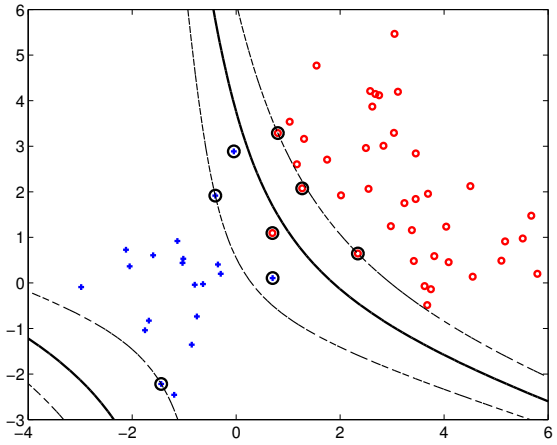
○ K1 √ **K2** ○ K3 ○ Kg
○ C1 √ **C2**



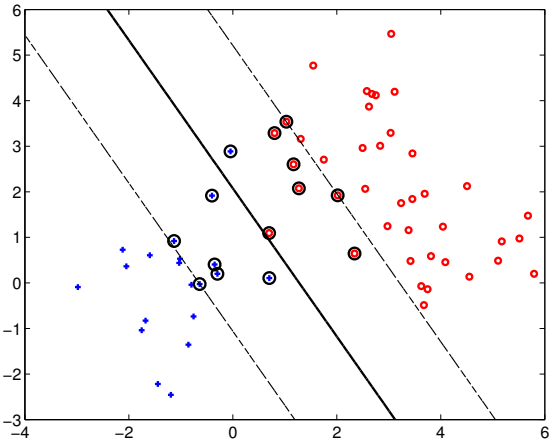
○ K1 ○ K2 ○ K3 √ **Kg**
○ C1 √ **C2**



○ K1 ○ K2 √ **K3** ○ Kg
√ **C1** ○ C2



○ K1 √ **K2** ○ K3 ○ Kg
√ **C1** ○ C2



√ **K1** ○ K2 ○ K3 ○ Kg
√ **C1** ○ C2

Name: _____

IPUC Kernel

4. (18 points) We are interested in doing regression, in which the input to our regressor will be subsets of some fixed finite set U, and the output will be a real number. We plan to apply kernel ridge regression.

   (a) First, we need to pick a kernel on sets. Here are three possible alternatives. For each one, either show that it is not a valid kernel or supply a feature transformation ϕ such that $k(x, z) = \phi(x) \cdot \phi(z)$. The notation $|S|$ stands for the cardinality of set S (for a finite set, that is the number of elements it contains.)

      (a) Intersection (I):

$$k_I(x, z) = |x \cap z|$$

> **Solution:** $\phi_I(x) =$ a bit vector of length $|U|$, where bit $b_i = 1$ iff element i is in x and 0 otherwise.

      (b) Intersection minus union (IMU):

$$k_{imu}(x, z) = |x \cap z| - |x \cup z|$$

> **Solution:** Consider the sets $x_1 = \{a\}$ and $x_2 = \{b\}$. Then the kernel matrix is $K = ((0, -2), (-2, 0))$.
> Consider an arbitrary vector $v = (v_1, v_2)$.
>
> $$v^T K v = -2(v_1^2 + v_2^2)$$
>
> which is negative, so K is not positive definite.

      (c) Intersection plus union complement (IPUC):

$$k_{ipuc}(x, z) = |x \cap z| + (|U| - |x \cup z|)$$

> **Solution:** $\phi_{ipuc}(x) = \phi_I(x)$ concatenated with $\phi_I\bar{(}x)$ where $\bar{(}v)$ is the bit-wise complement of vector v.

(d) Using the IPUC kernel, and assuming $|U| = 10$, we will perform a kernelized regression, finding parameters α_i, so that the predictions are of the form:

$$\hat{y} = \sum_{i=1}^{N} \alpha_i k_{\text{ipuc}}(x^{(i)}, x).$$

Assume you have the following training data:

i	$x^{(i)}$	$y^{(i)}$
1	$\{\square, \clubsuit, \heartsuit, \spadesuit\}$	4
2	$\{\heartsuit, \triangle, \square, \flat, \natural, \sharp\}$	-2
3	$\{\heartsuit\}$	1

Given a new input $x = \{\heartsuit, \spadesuit\}$, provide an expression for the predicted y value in terms of the α_i parameters.

Solution:

$$\alpha_1 k(x^{(1)}, x) + \alpha_2 k(x^{(2)}, x) + \alpha_3 k(x^{(3)}, x)$$
$$= \alpha_1(2 + 10 - 4) + \alpha_2(1 + 10 - 7) + \alpha_3(1 + 10 - 2)$$
$$= 8\alpha_1 + 4\alpha_2 + 9\alpha_3$$

Name: _____

Uniformly naive

5. (18 points) Consider a generative approach to classification, in which we estimate $P(Y)$ and $P(X|Y)$ from data. There are two classes, 0 and 1. We will make the same independence assumption as in Naive Bayes, that the features X_j are independent of each other given the class Y, but the features are d real-valued random variables, with independent uniform distributions. So:

$$Y \sim \text{Bernoulli}(q_1) \tag{1}$$

$$X_j \mid Y = c \sim \text{Uniform}(a_{cj}, b_{cj}) \quad \text{for } 1 \leqslant j \leqslant d \tag{2}$$

where $c \in \{0, 1\}$ and $q_0 = 1 - q_1$.
So, the parameter vector $\theta = q_1, a_{01}, b_{01}, a_{11}, b_{11}, \ldots, a_{0d}, b_{0d}, a_{1d}, b_{1d}$.

(a) For a data set $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^n$, write a formula for the log likelihood $P(D; \theta)$ in terms of x and y values in the data set and parameter values in θ.

Solution:

$$\begin{aligned}\log P(D;\theta) \quad = \quad & \sum_{i=1}^{n} y^{(i)} \log q_1 \left(\begin{cases} \frac{1}{\prod_{j=1}^{d}(b_{1j}-a_{1j})} & \text{if } a_{1j} \leqslant x_j \leqslant b_{1j} \text{ for all } 1 \leqslant j \leqslant d \\ 0 & \text{otherwise} \end{cases} \right) \\ & + (1 - y^{(i)}) \log q_0 \left(\begin{cases} \frac{1}{\prod_{j=1}^{d}(b_{0j}-a_{0j})} & \text{if } a_{0j} \leqslant x_j \leqslant b_{0j} \text{ for all } 1 \leqslant j \leqslant d \\ 0 & \text{otherwise} \end{cases} \right)\end{aligned}$$

(b) Given parameters θ and a new example x, such that for all feature indices j, $a_{1j} \leqslant x_j \leqslant b_{1j}$ and $a_{0j} \leqslant x_j \leqslant b_{0j}$, under what conditions would you predict that it belongs to class 1? Express your answer in terms of elements of x and θ.

Solution:

$$\begin{aligned} P(Y = 1 \mid X) &> P(Y = 0 \mid X) \\ P(X \mid Y = 1)P(Y = 1) &> P(X \mid Y = 0)P(Y = 0) \\ P(X \mid Y = 1)P(Y = 1) &> P(X \mid Y = 0)P(Y = 0) \\ \frac{q_1}{\prod_{j=1}^{d}(b_{1j} - a_{1j})} &> \frac{q_0}{\prod_{j=1}^{d}(b_{0j} - a_{0j})} \end{aligned}$$

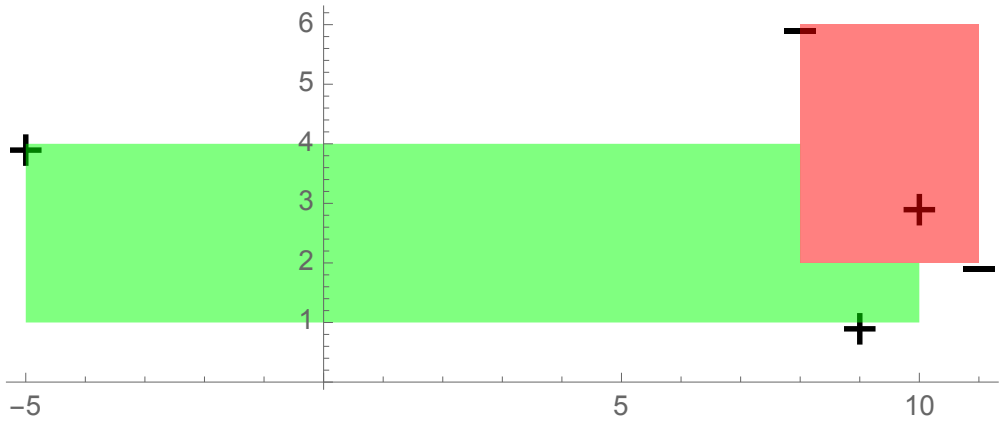
(c) Given training data

x	y
$(10, 3)$	1
$(9, 1)$	1
$(-5, 4)$	1
$(11, 2)$	0
$(8, 6)$	0

What are the maximum-likelihood parameter estimates?

Solution: $q_1 = 3/5; q_0 = 2/5; a_{11} = -5; b_{11} = 10; a_{01} = 8; b_{01} = 11; a_{12} = 1; b_{12} = 4; a_{02} = 2; b_{02} = 6$

(d) Given the same training data (plotted below), and using the maximum-likelihood parameter estimates, label very clearly all regions of the space that would be classified as positive and those that would be classified as negative.



Solution: The positive box has dimensions 15 x 3 and probability 3/5. The negative box has dimensions 3 x 4 and probability 2/5.

Compare (1/75) with (1/30). That means that, inside the negative box, a negative label is most likely.

Name: _____

Off to the races

6. (10 points) You are trying to decide what fraction, g, of your wealth to bet on the next horse race. You can observe a vector x of features of the horse. This particular horse will either win ($y = 1$) or lose ($y = 0$) the race. Your loss function is, for some fixed positive constant $c > 0$,

$$L(g, y) = \begin{cases} -cg & \text{if } y = 1 \\ g & \text{if } y = 0 \end{cases}$$

That is, if the horse wins and you bet fraction g of your money, then you win cg in profit (that is, your loss is $-cg$); if the horse loses, then you lose your bet g.

You have a data set $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{n}$, representing previous examples of horses labeled with whether or not they have won their races.

We would like to use the principle of empirical risk minimization to estimate parameters w to fit a model of the form $g = \sigma(w \cdot x)$ to the data, where σ is the sigmoid function.

(a) Write an expression for the empirical risk as a function of w, in terms of σ, w, c, and elements of D.

Solution:

$$ER(w) = \sum_{i=1}^{n} \sigma(w \cdot x^{(i)})(-c)^{y^{(i)}}$$

(b) What would the update rule for a stochastic gradient optimizer be? Please write it in terms of σ, w, c, $x^{(j)}$, and $y^{(j)}$, where $(x^{(j)}, y^{(j)})$ is a new training example.

Solution:

$$w = w - \eta\sigma(w \cdot x^{(j)})(1 - \sigma(w \cdot x^{(j)})) \cdot x^{(j)} \cdot (-c)^{y^{(i)}}$$

Name: _____

Regression with variances

7. (18 points) Your friend Dana is an astronomer who is trying to predict the degree of sunspot activity, Y, as a function of a vector of observable parameters X. Dana believes the observations of X are very reliable, but the Y observations are corrupted by Gaussian noise. Furthermore, the noise depends on the atmospheric conditions and may be different on every observation. Luckily, last year, the astronomers developed a good way of predicting the level of noise, and so Dana has a data set consisting of triples $D = \{(x^{(i)}, y^{(i)}, v^{(i)})\}_{i=1}^{n}$. We make the modeling assumption that, for some weight vector w, and for all i,

$$Y^{(i)} \mid X^{(i)} = x^{(i)} \sim \text{Normal}(w \cdot x^{(i)}, v^{(i)})$$

Note: $v^{(i)}$ is a variance.

(a) Write an expression for the log likelihood of the data in terms of parameters w and elements of D.

Solution:

$$\log P(D; w) = \sum_{i=1}^{n} -\frac{1}{2}\log(2v^{(i)}) - \frac{(w \cdot x^{(i)} - y^{(i)})^2}{2v^{(i)}}$$

(b) Derive a stochastic gradient descent update rule for w. Please write it in terms of w, $x^{(j)}$, $y^{(j)}$, and $v^{(j)}$ where $(x^{(j)}, y^{(j)}, v^{(j)})$ is a new training example.

Solution:

$$w := w - \eta \frac{x^{(i)}}{v^{(i)}}(y^{(i)} - w \cdot x^{(i)})$$

(c) If all the $v^{(i)}$ are equal is this the same as ordinary least squares? Explain briefly.

Solution: Yes. All samples have the same variance, which is the standard OLS assumption.

(d) Is there a value of $v^{(i)}$ that would cause the maximum likelihood weight estimates to be independent of $x^{(i)}$ and $y^{(i)}$? Explain briefly.

Solution: Infinity. We're dividing by the variance in the update, so the bigger the variance, the less effect it has on the result.

(e) Is there a value of $v^{(i)}$ that would cause the maximum likelihood weight estimates to be independent of $x^{(j)}$ and $y^{(j)}$, for $j \neq i$, irrelevant? Explain briefly.

Solution: 0. We're dividing by the variance in the update, so the smaller the variance, the more effect it has on the result (and the less the other observations have, in comparison).