Homework 1 Solution

9-22-2018

Warm up:

1.

If f_1 and f_2 are convex functions then their *pointwise maximum* f, defined by

$$f(x) = \max\{f_1(x), f_2(x)\},$$

with $\mathbf{dom}\, f = \mathbf{dom}\, f_1 \cap \mathbf{dom}\, f_2$, is also convex. This property is easily verified: if $0 \leq \theta \leq 1$ and $x,\ y \in \mathbf{dom}\, f$, then

$$\begin{aligned} f(\theta x + (1-\theta)y) &= \max\{f_1(\theta x + (1-\theta)y), f_2(\theta x + (1-\theta)y)\} \\ &\leq \max\{\theta f_1(x) + (1-\theta)f_1(y), \theta f_2(x) + (1-\theta)f_2(y)\} \\ &\leq \theta \max\{f_1(x), f_2(x)\} + (1-\theta)\max\{f_1(y), f_2(y)\} \\ &= \theta f(x) + (1-\theta)f(y), \end{aligned}$$

which establishes convexity of f. It is easily shown that if $f_1, \ldots, f_m$ are convex, then their pointwise maximum

$$f(x) = \max\{f_1(x), \ldots, f_m(x)\}$$

is also convex.

Note: the domain of f: we denote **dom** f.

2.

Subgradient of a convex function $f(x)$ at point x_0 is any line tangent to $f(x)$ that passes through x_0 and underestimates $f(x), \forall x \in R$

(a)

$$f(x) = max\{x^2 - 2x, |x|\}$$

Let us open up $f(x)$.

1. if $x < 0$

   $$f(x) = max\{x^2 - 2x, -x\}$$

   Since,

   $$x^2 - 2x \geq -x$$
   $$\Rightarrow x(x-1) \geq 0$$

   Therefore,

   $$\forall x < 0, f(x) = x^2 - 2x$$

2. if $x \geq 0$

   $$f(x) = max\{x^2 - 2x, x\}$$

   Since,

   $$x^2 - 2x \geq x$$
   $$\Rightarrow x(x-3) \geq 0$$

   Therefore,

   $$f(x) = \begin{cases} x & 0 \leq x < 3 \\ x^2 - 2x & x \geq 3 \end{cases}$$

$$\Rightarrow f(x) = \begin{cases} x^2 - 2x & x < 0 \\ x & 0 \leq x < 3 \\ x^2 - 2x & x \geq 3 \end{cases}$$

$$\text{subgradient of } f(x) = \begin{cases} \frac{\partial(x^2-2x)}{\partial x} & x < 0 \\ [\frac{\partial(x^2-2x)}{\partial x}, \frac{\partial x}{\partial x}] & x = 0 \\ \frac{\partial x}{\partial x} & 0 < x < 3 \\ [\frac{\partial x}{\partial x}, \frac{\partial(x^2-2x)}{\partial x}] & x = 3 \\ \frac{\partial(x^2-2x)}{\partial x} & x \geq 3 \end{cases}$$

At $x = 0$, subgradient of $f(x) \in [-2, 1]$
$\Rightarrow \vec{0}$ is a subgradient of $f(x)$ at $x = 0$

At $x = -2$, subgradient of $f(x) = 2x - 2 = 2(-2) - 2 = -6$

(b)

$$g(x) = max\{(x-1)^2, (x-2)^2\}$$

Since,

$$(x-1)^2 \geq (x-2)^2$$
$$\Rightarrow x \geq 1.5$$

Therefore,

$$\Rightarrow g(x) = \begin{cases} (x-2)^2 & x < 1.5 \\ (x-1)^2 & x \geq 1.5 \end{cases}$$

$$\text{subgradient of} g(x) = \begin{cases} \frac{\partial(x-2)^2}{\partial x} & x < 1.5 \\ [\frac{\partial(x-2)^2}{\partial x}, \frac{\partial(x-1)^2}{\partial x}] & x = 1.5 \\ \frac{\partial(x-1)^2}{\partial x} & x > 1.5 \end{cases}$$

At $x = 1.5$, subgradient of $g(x) \in [2(x-2), 2(x-1)]$
subgradient of $g(x) \in [-1, 1]$
One of the subgradients of $g(x)$ at $x = 1.5$ is $\vec{0}$

At $x = 0$, subgradient of $g(x) = 2(x-2) = 2(0-2) = -4$

Problem 1: Perceptron Learning

1.

Number of iterations to find the classifier: 47

2.

Number of iterations to find the classifier: 1091000

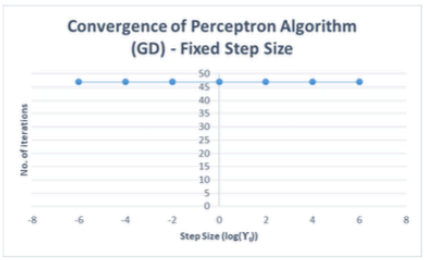
i	w_1	w_2	w_3	w_4	b
1	1278.99646108	460.06125801	-108.55851404	-1672.31572948	-354.0
2	1307.29472974	432.74778799	-27.55191988	-1523.78895446	-493.0
3	1255.18981362	425.50402882	18.7965404	-1434.66754197	-625.0
47(Final)	685.79932892	243.89947473	8.24199193	-797.62505314	-1485.0

Table 1: Gradient Descent, $\gamma_t = 1$: w, b values at the end of i number of iterations

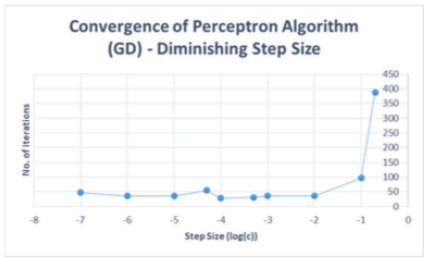
i	w_1	w_2	w_3	w_4	b
1	4.61754424	2.46967938	1.96766079	-1.81335551	-1.0
2	4.61754424	2.46967938	1.96766079	-1.81335551	-1.0
3	3.45322288	0.16943482	2.62801595	-4.64709851	-2.0
1091000(Final)	149.27714019	52.53347317	1.67167265	-172.89194014	-322.0

Table 2: Stochastic Gradient Descent, $\gamma_t = 1$: w, b values at the end of i number of iterations

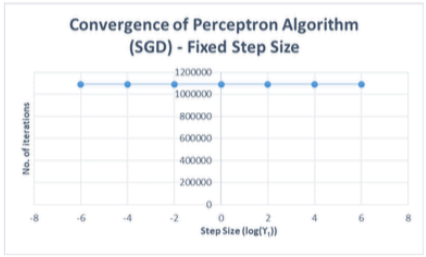
3.



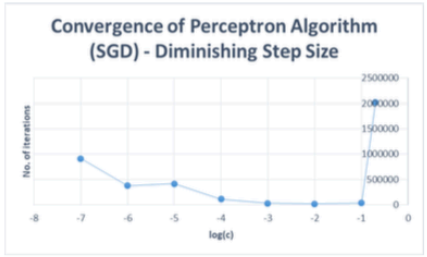
(a) Gradient Descent with Fixed Step Size



(b) Gradient Descent with Diminishing Step Size



(c) Stochastic Gradient Descent with Fixed Step Size



(d) Stochastic Gradient Descent with Diminishing Step Size

Figure 1: Perceptron: Convergence Analysis using Gradient Descent and Stochastic Gradient Descent with different step sizes

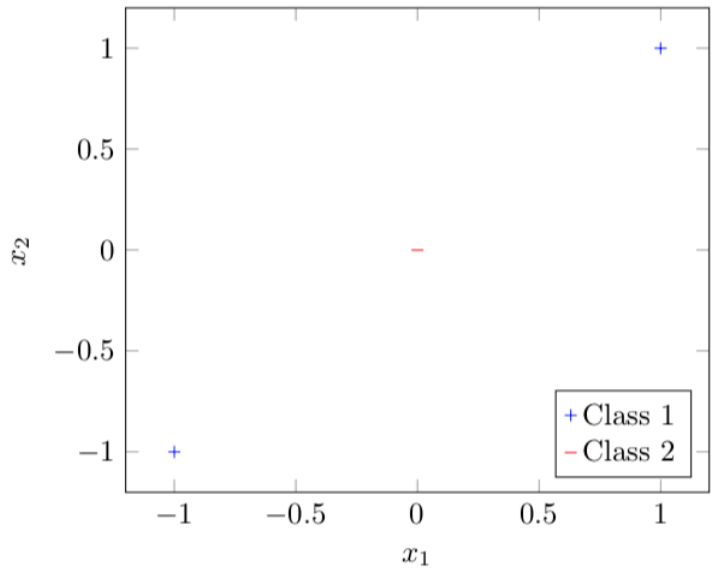
On the dataset, different step sizes were tried in order to understand how it affects the convergence of the perceptron algorithm.

- **Fixed Step Size:** Different fixed step sizes ranging from 10^{-6} to 10^{6} were tried out. The convergence of the algorithms did not get affected by the step size on this dataset. This was perhaps due to the smaller size of the dataset.
- **Diminishing Step Size:** The following step size was used:

$$\gamma_t = \frac{1}{1 + c.t}$$

where c was varied from 0.2 to 10^{-7}. For higher values of c, the algorithm converges very slowly since the step size $\propto \frac{1}{ct}$. On the other hand, for very low values of c, the algorithm is equivalent to using a fixed step size. For intermediary values, the algorithm converges faster than at either extremes. This can be attributed to the fact that the step size changes with each iteration (unlike with fixed step size) but does not become too small too early either (unlike with higher c values). So, a diminishing step size allows faster conversions if the parameter c is tuned properly.

4.



The perceptron algorithm will not converge on the dataset of size = 3 shown above since this dataset is not linearly separable.
If it were linearly separable, we would be able to find a classifier $f(x) = \vec{a}^T\vec{x} + b$ such that it fits to the data
$D(\vec{x}, y) = \{([-1, -1], 1), ([1, 1], 1), ([0, 0], -1)\}$
Or,

$$\forall \vec{x}, y; y = a_1x_1 + a_2x_2 + b$$

Therefore, the following system of linear equations will be consistent:

$$(-1)a_1 + (-1)a_2 + b = 1 \tag{1}$$

$$(1)a_1 + (1)a_2 + b = 1 \tag{2}$$

$$(0)a_1 + (0)a_2 + b = -1 \tag{3}$$

Add equations (1) & (2)

$$\Rightarrow b = 1$$

However, this is inconsistent with equation (3) Therefore, this system of linear equations has no solution.
$\Rightarrow$ There exists no linear separator of the form $f(x) = \vec{a}^T\vec{x} + b$ on the dataset D.
It can be inferred from this example that even more generally, the perceptron algorithm will not converge on datasets that are not linearly separable. Moreover, the way the algorithm has been defined, it also does not output a best-fit solution by allowing misclassifications on a few data points.

Note:

1.4 The smallest set of data points for which the perceptron algorithm will fail to converge is 2. If there are two identical data points with different labels, the algorithm will not converge. Full credit would also be given to the answer three (all points along a line with alternating +'s and –'s) If the student explicitly ruled out the case that the two identical data points could have different labels. Students should receive partial credit for providing a data set for which the perceptron algorithm does not converge even if it is not the smallest.

Problem 2: Separability & Feature Vectors

1.

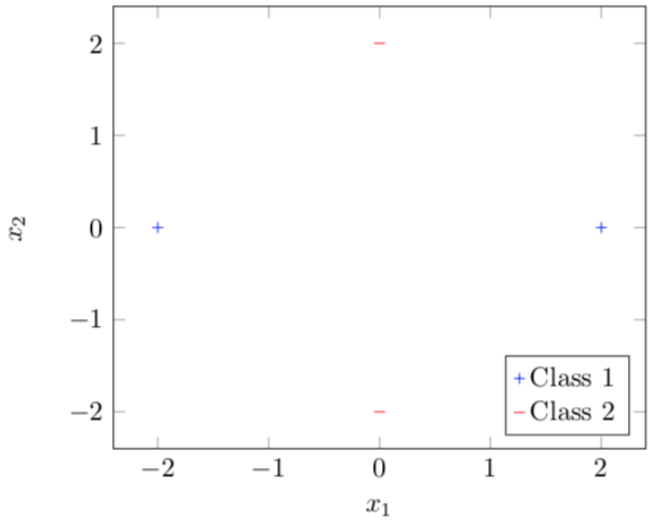
$D(\vec{x}, y) = \{([-1,-1],1), ([1,-1],-1), ([1,1],1), ([-1,1],-1)\}$

(a)

On applying the feature transformation

$$\phi(x_1, x_2) = \begin{bmatrix} x_1 + x_2 \\ x_1 - x_2 \end{bmatrix}$$

The transformed dataset looks as follows:



$D' = D(\phi(\vec{x}), y) = \{([-2, 0], 1), ([0, 2], -1), ([2, 0], 1), ([0, -2], -1)\}$
Let $\vec{x'} = \phi(\vec{x})$
If D' is linearly separable, then $\exists \vec{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$ and b where, $a_1, a_2, b \in R$ such that

$$\forall \vec{x'}, y \in D', y = \vec{a}^T x + b$$

Or,

$$a_1 {x'}_1 + a_2 {x'}_2 + b = y$$

$\Rightarrow$ The following system of equations must be consistent.

$$-2a_1 + b = 1 \tag{1}$$

$$2a_2 + b = -1 \tag{2}$$

$$2a_1 + b = 1 \tag{3}$$

$$-2a_2 + b = -1 \tag{4}$$

Add equations (1) and (3)

$$\Rightarrow b = 1 \tag{5}$$

Add equations (2) and (4)

$$\Rightarrow b = -1 \tag{6}$$

Equations (5) and (6) are inconsistent. Therefore, this system of linear equations has no solution and no linear separator exists for the dataset $D(\phi(\vec{x}), y)$.

(b)

On applying the feature transformation

$$\phi(x_1, x_2) = \begin{bmatrix} {x_1}^2 \\ {x_2}^2 \\ x_1 x_2 \end{bmatrix}$$

$D' = D(\phi(\vec{x}), y) = \{([1,1,1],1),([1,1,-1],-1),([1,1,1],1),([1,1,-1],-1)\}$
Let $\vec{x'} = \phi(\vec{x})$

If D' is linearly separable, then $\exists \vec{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$ and b where, $a_1, a_2, a_3, b \in R$ such that

$$\forall \vec{x'}, y \in D', y = \vec{a}^T x + b$$

Or,

$$a_1 {x'}_1 + a_2 {x'}_2 + a_3 {x'}_3 + b = y$$

$\Rightarrow$ The following system of equations must be consistent.

$$a_1 + a_2 + a_3 + b = 1 \tag{1}$$

$$a_1 + a_2 - a_3 + b = -1 \tag{2}$$

Subtract equations (1) and (2)

$$\Rightarrow a_3 = 1 \tag{3}$$

Substituting equation (3) in (1)

$$\Rightarrow a_1 + a_2 + b = 0 \tag{4}$$

This system of equations is consistent and has infinitely many solutions.
Any hyperplane of the form $\vec{a}^T \vec{x} + b$ where the constraints given by equations (3) and (4) are satisfied will be a linear separator on the dataset D'.

One such separator is $f(x'_1, x'_2, x'_3) = [1,1,1] \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} + (-2) = 0$

On D',

$$f(\vec{x'}^{(1)}) = f(\vec{x'}^{(3)}) = f(1,1,1) = 1 > 0$$

$$f(\vec{x'}^{(2)}) = f(\vec{x'}^{(4)}) = f(1,1,-1) = -1 < 0$$

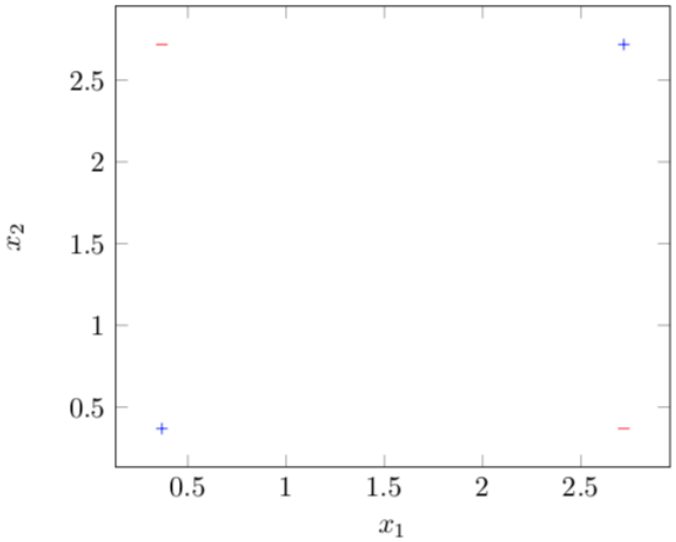
Therefore, the hyperplane $f(x'_1, x'_2, x'_3) = [1,1,1] \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} + (-2) = 0$ separates the dataset $D(\phi(\vec{x}), y)$.

(c)

On applying the feature transformation

$$\phi(x_1, x_2) = \begin{bmatrix} exp(x_1) \\ exp(x_2) \end{bmatrix}$$

The transformed dataset looks as follows:



$D' = D(\phi(\vec{x}), y) = \{([e^{-1}, e^{-1}], 1), ([e^1, e^{-1}], -1), ([e^1, e^1], 1), ([e^{-1}, e^1], -1)\}$
Let $\vec{x'} = \phi(\vec{x})$
If D' is linearly separable, then $\exists \vec{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$ and b where, $a_1, a_2, b \in R$ such that

$$\forall \vec{x'}, y \in D', y = \vec{a}^T x + b$$

Or,

$$a_1 x'_1 + a_2 x'_2 + b = y$$

$\Rightarrow$ The following system of equations must be consistent.

$$e^{-1}a_1 + e^{-1}a_2 + b = 1 \tag{1}$$

$$ea_1 + e^{-1}a_2 + b = -1 \tag{2}$$

$$ea_1 + ea_2 + b = 1 \tag{3}$$

$$e^{-1}a_1 + ea_2 + b = -1 \tag{4}$$

We have 4 equations and 3 unknowns. Therefore, we will solve the first 3 equations and check if their solution (if any) is consistent with equation ((4)) From equation (1), we get

$$\Rightarrow a_1 + a_2 = e(1 - b) \tag{5}$$

From equation (3), we get

$$\Rightarrow a_1 + a_2 = (1 - b)e^{-1} \tag{6}$$

Therefore,

$$e(1 - b) = e^{-1}(1 - b) \Rightarrow b = 1 \tag{7}$$

$$\Rightarrow a_1 + a_2 = 0 \tag{8}$$

Substitute equation (7) into (2)

$$ea_1 + e^{-1}a_2 = -2 \Rightarrow e^2a_1 + a_2 = -2e \tag{9}$$

Solving equations (8) and (9)

$$a_1 = \frac{-2e}{e^2 - 1}, a_2 = \frac{2e}{e^2 - 1} \tag{10}$$

Substitute the values of a_1, a_2, b into equation (4)

$$e^{-1}\frac{-2e}{e^2 - 1} + e\frac{2e}{e^2 - 1} + 1 = 3 \neq 1 \tag{11}$$

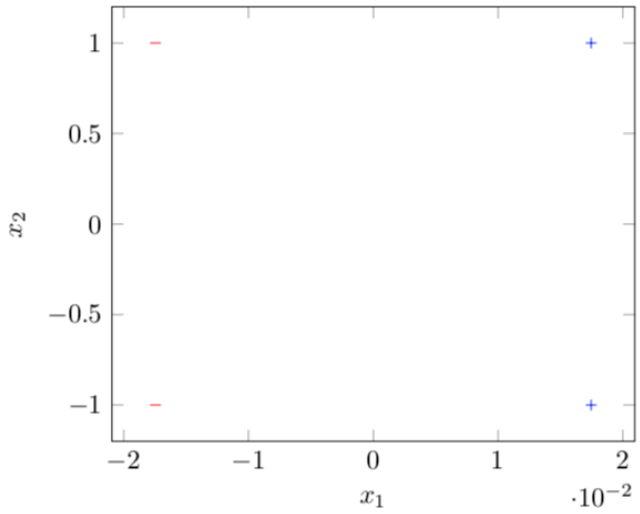
$\Rightarrow$ This system of equations is inconsistent and has no solution. Therefore, no linear separator exists for the dataset $D(\phi(\vec{x}), y)$.

(d)

On applying the feature transformation

$$\phi(x_1, x_2) = \begin{bmatrix} x_1 sin(x_2) \\ x_1 \end{bmatrix}$$

The transformed dataset looks as follows:



$D' = D(\phi(\vec{x}), y) = \{([-sin(-1), -1], 1), ([sin(-1), 1], -1), ([sin(1), 1], 1), ([-sin(1), -1], -1)\}$
Let $\vec{x'} = \phi(\vec{x})$
If D' is linearly separable, then $\exists \vec{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$ and b where, $a_1, a_2, b \in R$ such that

$$\forall \vec{x'}, y \in D', y = \vec{a}^T x + b$$

Or,

$$a_1 x'_1 + a_2 x'_2 + b = y$$

$\Rightarrow$ The following system of equations must be consistent.

$$-sin(-1)a_1 - a_2 + b = 1 \Rightarrow sin(1)a_1 - a_2 + b = 1 \tag{1}$$

$$sin(-1)a_1 + a_2 + b = -1 \Rightarrow -sin(1)a_1 + a_2 + b = -1 \tag{2}$$

$$sin(1)a_1 + a_2 + b = 1 \tag{3}$$

$$-sin(1)a_1 - a_2 + b = -1 \tag{4}$$

We have 4 equations and 3 unknowns. Therefore, we will solve the first 3 equations and check if their solution (if any) is consistent with equation ((4)) Add equations (1) and (2)

$$\Rightarrow b = 0 \tag{5}$$

Substitute equation (5) into (2) and (3) and add them

$$\Rightarrow a_2 = 0 \tag{6}$$

Substitute equations (5) and (6) into (3)

$$\Rightarrow a_1 = \frac{1}{sin(1)} \tag{7}$$

Check if the values of a_1, a_2, b are consistent with equation (4)

$$-sin(1)\frac{1}{sin(1)} - 0 + 0 = -1 = -1$$

Therefore, these equations are consistent.
One such linear separator is $f(\vec{x'}) = [\frac{1}{sin(1)}, 0]\vec{x'} = 0$
On D',

$$f(\vec{x'}^{(1)}) = f(-sin(-1), -1) = f(sin(1), -1) = \frac{1}{sin(1)} sin(1) = 1 > 0$$

$$f(\vec{x'}^{(2)}) = f(sin(-1), 1) = f(-sin(1), 1) = \frac{1}{sin(1)}(-1)sin(1) = -1 < 0$$

$$f(\vec{x'}^{(3)}) = f(sin(1), 1) = \frac{1}{sin(1)} sin(1) = 1 > 0$$

$$f(\vec{x'}^{(4)}) = f(-sin(1), -1) = \frac{1}{sin(1)}(-1)sin(1) = -1 < 0$$

Therefore, the hyperplane $f(x'_1, x'_2) = [\frac{1}{sin(1)}, 0] \begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} = 0$ separates the dataset $D(\phi(\vec{x}), y)$.

2.

Dataset:

$$D(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, y)$$

Linear regression on two-dimensional data:

$$f(x_1, x_2) = a_0 + a_{10}x_1 + a_{11}x_2$$

Quadratic regression on two-dimensional data:

$$f(x_1, x_2) = a_0 + a_{10}x_1 + a_{11}x_2 + a_{20}{x_1}^2 + a_{21}x_1x_2 + a_{22}{x_2}^2$$

Polynomial regression with a polynomial of degree k on two-dimensional data:

$$f(x_1, x_2) = a_0 + \sum_{i=1}^{k}\sum_{j=0}^{i} a_{ij}{x_1}^{i-j}{x_2}^j$$

It should be noted that though the data has been transformed to a k-degree polynomial feature space, the model stays linear.

$$\phi(\vec{x_1, x_2})^T = [1, x_1, x_2, ..., {x_1}^k, ..., {x_2}^k]$$

There are $(i+1)$ elements in the feature vector with degree i : $[{x_1}^i, {x_1}^{i-1}x_2, ..., x_1{x_2}^{i-1}, {x_2}^i]$
Size of the feature space $(size_{fs}) = \sum_{i=0}^{k}(i+1) = \frac{(k+1)(k+2)}{2}$

$$f(x_1, x_2) = \vec{a}^T\phi(\vec{x_1, x_2})$$

where,

$$\vec{a}^T = [a_0a_{10}a_{11}, ..., a_{k0}, ..., a_{kk}]$$

Assuming that there are total M data points, the loss function can be defined as follows if we use average squared loss function

$$L(f) = \frac{1}{M}\sum_{m=1}^{M}(f(\vec{x}^{(m)}) - y^{(m)})^2$$

$$L(f) = \frac{1}{M}\sum_{m=1}^{M}((a_0 + \sum_{i=1}^{k}\sum_{j=0}^{i} a_{ij}({x_1}^{i-j}{x_2}^j)^{(m)}) - y^{(m)})^2$$

On using standard gradient descent to minimize the loss:

$$\nabla L(f)_{a_0} = \frac{2}{M}\sum_{m=1}^{M}((a_0 + \sum_{i=1}^{k}\sum_{j=0}^{i} a_{ij}({x_1}^{i-j}{x_2}^j)^{(m)}) - y^{(m)})$$

$$\nabla L(f)_{a_{ij}} = \frac{2}{M}\sum_{m=1}^{M}((a_0 + \sum_{i=1}^{k}\sum_{j=0}^{i} a_{ij}({x_1}^{i-j}{x_2}^j)^{(m)}) - y^{(m)})({x_1}^{i-j}{x_2}^j)^{(m)}$$

The coefficients would be updated as follows:

$$a_0 = a_0 - \gamma_t \nabla L(f)_{a_0}$$

$$a_{ij} = a_i j - \gamma_t \nabla L(f)_{a_{ij}}$$

Assuming that all the coefficients a_{ij} are updated in parallel, the complexity of a single iteration of this algorithm can be evaluated as follows:
Complexity = O(number of data points x size of feature space) = $O(M \times size_{fs})$

Problem 3: Piecewise Linear Regression

$f(x) = max(a_1 x + b_1, a_2 x + b_2)$
Or,

$$f(x_1, x_2) = max(a_{11} x_1 + a_{12} x_2 + b_1, a_{21} x_1 + a_{22} x_2 + b_2)$$

where, $x = [x_1, x_2]$ is a 2-dimensional vector.

1.

$$D(x^{(m)}, y^{(m)})$$

Given M data points, the regression problem can be defined as optimizing the following squared loss function:

$$L(f) = \frac{1}{M} \sum_{m=1}^{M} \left(f(x^{(m)}) - y^{(m)}\right)^2$$

With the given hypothesis class, it can be refined to the following function:

$$L(f) = \frac{1}{M} \sum_{m=1}^{M} \left(max(a_1 x^{(m)} + b_1, a_2 x^{(m)} + b_2)) - y^{(m)}\right)^2$$

$$f(x) = \begin{cases} a_2 x + b_2 & \text{if } a_1 x + b_1 < a_2 x + b_2 \\ a_1 x + b_1 & \text{if } a_1 x + b_1 \geq a_2 x + b_2 \end{cases}$$

$$\Rightarrow L(f) = \sum_{m=1}^{M} (a_2 x^{(m)} + b_2 - y^{(m)})^2 {}_{a_1 x^{(m)} + b_1 < a_2 x^{(m)} + b_2} + (a_1 x^{(m)} + b_1 - y^{(m)})^2 {}_{a_1 x^{(m)} + b_1 \geq a_2 x^{(m)} + b_2}$$

The problem can then be defined as finding the optimum values of a_1, a_2, b_1, b_2 such that the loss $L(f)$ is minimum. Once such a_1, a_2, b_1, b_2 have been determined, for any new x, the function $f(x) = max(a_1 x + b_1, a_2 x + b_2)$ can be used to obtain its corresponding value y.

2.

To minimize the loss function, we would use subgradient descent as follows:
Compute the subgradients,

$$\nabla L(f)_{a_1} = \frac{2}{M}\sum_{m=1}^{M}(a_1x^{(m)} + b_1 - y^{(m)})x^{(m)}{}_{a_1x^{(m)}+b_1 \geq a_2x^{(m)}+b_2}$$

$$\nabla L(f)_{a_2} = \frac{2}{M}\sum_{m=1}^{M}(a_2x^{(m)} + b_2 - y^{(m)})x^{(m)}{}_{a_1x^{(m)}+b_1 < a_2x^{(m)}+b_2}$$

$$\nabla L(f)_{b_1} = \frac{2}{M}\sum_{m=1}^{M}(a_1x^{(m)} + b_1 - y^{(m)})_{a_1x^{(m)}+b_1 \geq a_2x^{(m)}+b_2}$$

$$\nabla L(f)_{b_2} = \frac{2}{M}\sum_{m=1}^{M}(a_2x^{(m)} + b_2 - y^{(m)})_{a_1x^{(m)}+b_1 < a_2x^{(m)}+b_2}$$

Use the subgradients to update the parameters a_1, a_2, b_1, b_2

$$a_1 = a_1 - \gamma_t \nabla L(f)_{a_1}$$

$$a_2 = a_2 - \gamma_t \nabla L(f)_{a_2}$$

$$b_1 = b_1 - \gamma_t \nabla L(f)_{b_1}$$

$$b_2 = b_2 - \gamma_t \nabla L(f)_{b_2}$$

3.

$$\text{minimize} \quad J(\alpha) = \sum_{i=1}^{m}\left(\max_{j=1,\ldots,k}(a_j^T u_i + b_j) - y_i\right)^2, \tag{3}$$

with variables $a_1, \ldots, a_k \in \mathbf{R}^n$, $b_1, \ldots, b_k \in \mathbf{R}$. The function J is a piecewise-quadratic function of α. Indeed, for each i, $f(u_i) - y_i$ is piecewise-linear, and J is the sum of squares of these functions, so J is convex quadratic on the (polyhedral) regions on which $f(u_i)$ is affine. But J is not globally convex, so the fitting problem (3) is not convex.

Problem 4: Support Vector Machines

The original data is not linearly separable. Therefore, a feature transformation was applied on it to make it linearly separable.
The data was found to become linearly separable on fitting a polynomial of degree 4 on it. Note that doing so homogenizes the dataset as well, eliminating b from the standard SVM problem.

$$\phi(x_1, x_2, x_3, x_4)^T = [1, x_1, x_2, x_3, x_4, {x_1}^2, x_1x_2, ..., x_3{x_4}^3, {x_4}^4]$$

This is a vector of length 70. We need to find a hyperplane $w^T\phi(\vec{(x)}) = 0$ such that the following problem is solved:

$$min_w \frac{1}{2}\|w\|^2$$

subject to,

$$\forall i, -y_i(w^T(\phi(\vec{x}^{(i)}))) \leq -1$$

This is a quadratic programming problem with $M =$ size of dataset $= 1000$ linear constraints.
A general quadratic optimization problem has the form:

$$minimize_x \frac{1}{2}x^TPx + q^Tx$$

subject to

$$Gx \leq h$$

$$Ax = b$$

Let us convert our optimization problem into this form:

$$min_w \frac{1}{2} w^T w$$

subject to,

$$-y_i(w^T(\phi(\vec{x}^{(i)}))) \leq -1$$

$$\Rightarrow$$

$$P = I_{len(w) \times len(w)}$$

$$q = 0_{len(w) \times 1}$$

$$G = [-y_i \phi(x_i')]_{M \times len(w)}$$

$$h = [-1]_{M \times 1}$$

$$A = 0, b = 0$$

On solving it using a convex optimization solver, the following results were obtained:

1. w : A file containing the weights called 'weights_prob4.data' has been attached with the submission which has this information.
2. b : Since the dataset has been homogenized, w_0 will be considered as b. Therefore, $b = 7.29514847$
3. Optimal margin $= \frac{1}{\|w\|} = 0.07705720525729502$
4. Support vectors = The following data was obtained related to the support vectors taking into account the floating point rounding error. Please note that the index starts from 0.

Data point index	$[x_1 x_2 x_3 x_4]$	Class Label	$(w^T \phi(\vec{x})$
931	[0.21489345 0.89688165 0.85845297 0.5659933]	-1	0.9999999999992388
519	[0.02286581 0.23878615 0.23228752 0.87798567]	1	0.9999999999996447

Table 3: Support Vectors Data

Students do not need to produce my exact embedding, but should provide a perfect separator in some feature space.