HW4 – Gal Kaptsenel 209404409

Q1

1. Yes.
2. $g(x) = \begin{cases} 2x, & x < 0 \\ 2, & x \geq 0 \end{cases}$

   Let's split the proof according to the value of $u \in \mathbb{R}$

   - $u < 0$,
     $g(u) = 2u$
     $f(u) = u^2$
     And indeed, $\forall v \in \mathbb{R} = V$
     - If $v < 0$
       $$f(u) + g(u)(v-u) = u^2 + 2u(v-u) = u^2 + 2uv - 2u^2 = -u^2 + 2uv =$$
       $$= v^2\left(-\left(\frac{u}{v}\right)^2 + 2\frac{u}{v}\right) \overset{(1)}{\leq} v^2 = f(v)$$
       (1) $\left(\frac{u}{v}\right)^2 + 2\frac{u}{v} \overset{define\ t=\frac{u}{v}}{=} -t^2 + 2t$ has a maximum value in respect to $t = \frac{u}{v}$ at point $(1,1)$, and therefore the value of $-t^2 + 2t = -\left(\frac{u}{v}\right)^2 + 2\frac{u}{v}$ bounded by a maximum of 1.

       Therefore $g(u) \in \partial f(u)$

     - Otherwise, $v \geq 0$,
       $$f(u) + g(u)(v-u) = u^2 + 2u(v-u) = -u^2 + 2uv \overset{(1)}{\leq} 2uv \overset{(2)}{\leq} 0 \overset{(3)}{\leq} 2v = f(v)$$
       (1) $-u^2 \leq 0$
       (2) $u < 0, v \geq 0$
       (3) $v \geq 0$

       Therefore $g(u) \in \partial f(u)$

   - $u \geq 0$,
     $g(u) = 2$
     $f(u) = 2u$
     And indeed, $\forall v \in \mathbb{R} = V$
     $f(u) + g(u)(v-u) = 2u + 2(v-u) = 2u + 2v - 2u = 2v$
     - If $v < 0$
       $f(u) + g(u)(v-u) = 2v \leq 0 \leq v^2 = f(v)$

       Therefore $g(u) \in \partial f(u)$

     - If $v \geq 0$
       $f(u) + g(u)(v-u) = 2v \leq 2v = f(v)$
       Therefore $g(u) \in \partial f(u)$

   Therefore, at call cases we conclude that $\forall u \in \mathbb{R}, g(u) \in \partial f(u)$

3.

Yes, the algorithm will converge to a minimum at $x^* = 0$ with value $f(x^*) = 0$.

Lets prove that x_i is a series of points which obeys $x_i = -\left(\frac{1}{2}\right)^i$

Proof by induction over the iteration number,

<u>Iteration 0</u>

$$x_0 = -1 = -\left(\frac{1}{2}\right)^0$$

$\underline{assume\ for\ iteration\ n, prove\ for\ n+1}$

$$x_n = -\left(\frac{1}{2}\right)^n$$

At iteration $n+1$, $x_n < 0$, and therefore $g(x_n) = 2x_n = -\left(\frac{1}{2}\right)^{n-1}$, and then we will get that,

$$x_{n+1} = x_n - 0.25 * (g(x_n)) = x_n - 0.25 * \left(-\left(\frac{1}{2}\right)^{n-1}\right) = x_n + \left(\frac{1}{2}\right)^2 \left(\frac{1}{2}\right)^{n-1} =$$

$$= x_n + \left(\frac{1}{2}\right)^{n+1} = -\left(\frac{1}{2}\right)^n + \left(\frac{1}{2}\right)^{n+1} = -2 * \left(\frac{1}{2}\right)^{n+1} + \left(\frac{1}{2}\right)^{n+1} =$$

$$= -\left(\frac{1}{2}\right)^{n+1}$$

Therefore,

$$\lim_{i\to\infty} x_i = \lim_{i\to\infty} -\left(\frac{1}{2}\right)^i = 0$$

And therefore, the minimized function will converge to a value of $f(x^*) = 0$.

Indeed f is a non-negative function, which gets a value of 0 at point $x = 0$, and therefore the gradient decent algorithm indeed converges to the minimum.

I	x_i	$f(x_i)$	$\frac{\partial}{\partial x} f(x_i) = g(x_i)$
0	-1	1	-2
1	$-1 - 0.25 * (-2) = -\frac{1}{2}$	$\frac{1}{4}$	-1
2	$-\frac{1}{2} - 0.25 * (-1) = -\frac{1}{4}$	$\frac{1}{16}$	$-\frac{1}{2}$
3	$-\frac{1}{4} - 0.25 * \left(-\frac{1}{2}\right) = -\frac{1}{8}$	$\frac{1}{64}$	$-\frac{1}{4}$
4	$-\frac{1}{8} - 0.25 * \left(-\frac{1}{4}\right) = -\frac{1}{16}$	$\frac{1}{16^2}$	$-\frac{1}{8}$
…	…	…	…

4.

No,

The series of x_i will be $-1, 1, -1, 1, \ldots$, that is, the algorithm will alternate between -1 and 1, and will never converge to the minimum which, as stated at 1.3 above, is at $x^* = 0$.

Lets prove it by showing that each iteration of the algorithm, $x_i = 1$ or $x_i = -1$.

Proof by induction over the iteration number,

<u>Iteration 0</u>

$x_0 = -1$ and therefore the statement holds.

$\underline{assume\ for\ iteration\ n, prove\ for\ n+1}$

- If $x_n = 1$
  $g(x_n) = 2$, and therefore,
  $x_{n+1} = x_n - 1 * g(x_n) = 1 - 2 = -1$
- If $x_n = -1$
  $g(x_n) = -2$, and therefore,
  $x_{n+1} = x_n - 1 * g(x_n) = -1 + 2 = 1$

Therefore, it holds that at all cases, x_{n+1} equals to 1 or -1, and therefore the statement holds.

And indeed, as can be seen from the first three iterations,

I	x_i	$f(x_i)$	$\frac{\partial}{\partial x} f(x_i) = g(x_i)$
0	-1	1	-2
1	$-1 - (-2) = 1$	2	2
2	$1 - 2 = -1$	1	-2
…	…	…	…

The algorithm will alternate between $x_i = 1$ and $x_i = -1$, and will not converge.

Q2

Denote a random variable,

$Y_i - indicates\ the\ label\ the\ linear\ model\ obtains\ for\ sample\ x_i$

And it holds that given $\boldsymbol{w}$ and $\boldsymbol{x_i}$, $Y_i = \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle + \epsilon_i$. Note that,

$$P(Y_i = y_{i_i} | \boldsymbol{w}, \boldsymbol{x_i}) =$$

$$\overset{above}{=} P(\langle \boldsymbol{w}, \boldsymbol{x_i} \rangle + \epsilon_i = y_{i_i} | \boldsymbol{w}, \boldsymbol{x_i}) = P(\epsilon_i = y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle | \boldsymbol{w}, \boldsymbol{x_i}) \overset{(1)}{=} P(\epsilon_i = \epsilon | \boldsymbol{w}, \boldsymbol{x_i}) =$$

$$\overset{(2)}{=} \frac{1}{2b} \exp\left\{-\frac{|\epsilon - 0|}{b}\right\} = \frac{1}{2b} \exp\left\{-\frac{|y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|}{b}\right\}$$

(1) define $\epsilon = y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle$
(2) it is given that $\epsilon_i \sim laplace(0, b)$

Therefore, $Y_i | \boldsymbol{w}, \boldsymbol{x_i} \sim laplace(\langle \boldsymbol{w}, \boldsymbol{x_i} \rangle, b)$

Therefore,

$$\underset{\mathbf{w}}{\operatorname{argmax}} \Pi_{i=1}^{m} P(y_i, \boldsymbol{x_i} | \boldsymbol{w}) =$$

$$\overset{(1)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \Pi_{i=1}^{m} P(Y_i = y_i | \boldsymbol{x_i}, \boldsymbol{w}) \cdot P(\boldsymbol{x_i} | \boldsymbol{w}) \overset{(2)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \Pi_{i=1}^{m} \frac{1}{2b} \exp\left\{-\frac{|y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|}{b}\right\} \cdot P(\boldsymbol{x_i}) =$$

$$\overset{(3)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \Pi_{i=1}^{m} \frac{1}{2b} \exp\left\{-\frac{|y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|}{b}\right\} \overset{(4)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \frac{1}{2b} \exp\left\{-\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|\right\} =$$

$$\overset{(5)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \ln \frac{1}{2b} \exp\left\{-\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|\right\} \overset{(6)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \ln \frac{1}{2b} + \ln \exp\left\{-\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|\right\} =$$

$$\overset{(7)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} \ln \exp\left\{-\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|\right\} \overset{(8)}{=} \underset{\mathbf{w}}{\operatorname{argmax}} -\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle| =$$

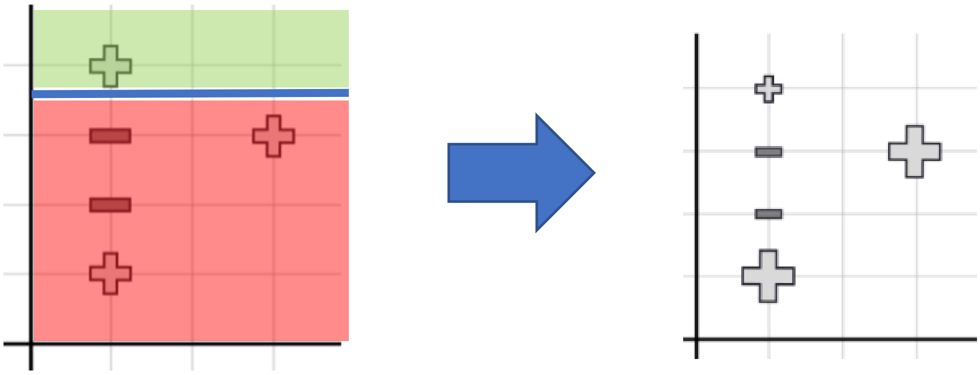
$$\overset{(9)}{=} \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle| \overset{(10)}{=} \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{m} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle| \overset{(11)}{=} \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{m} \Sigma_{i=1}^{m} |\boldsymbol{w}^T \boldsymbol{x_i} - y_i|$$

(1) conditional probability + definition of Y_i
(2) from above, $Y_i | \boldsymbol{w}, \boldsymbol{x_i} \sim laplace(\langle \boldsymbol{w}, \boldsymbol{x_i} \rangle, b)$, and $P(\boldsymbol{x_i} | \boldsymbol{w}) = P(\boldsymbol{x_i})$ because the samples are independent of the chosen vector of weights $\boldsymbol{w}$
(3) $\forall i \in [m], P(\boldsymbol{x_i}) \geq 0$ and therefore it does not affect the $\boldsymbol{w}$ which maximizes the expression, and therefore we can omit it from the expression.
(4) $\frac{1}{2b}$, constant value, could be extracted from the multiplication operator Π + exponent rules + the constant value $-\frac{1}{b}$ could be extracted from the summation operator Σ.
(5) $\ln x$ is a monophonic ascending function, and therefore it doesn't affect the $\boldsymbol{w}$ which maximizes the expression.
(6) $\ln rules$
(7) $\ln \frac{1}{2b}$ is a constant value, therefore it doesn't affect the $\boldsymbol{w}$ which maximizes the expression.
(8) $\ln e^x = x$
(9) maximizing the expression $-\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|$ is the same as minimizing the expression $\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|$
(10) Multiplying the expression by $\frac{b}{m}$, which is a constant value, doesn't affect the $\boldsymbol{w}$ which Minimizes $\frac{1}{b} \Sigma_{i=1}^{m} |y_i - \langle \boldsymbol{w}, \boldsymbol{x_i} \rangle|$.
(11) $\langle \boldsymbol{w}, \boldsymbol{x_i} \rangle = \boldsymbol{w}^T \boldsymbol{x_i}$

Q3

Figure **(a)**.

Figure **(a)** could be accomplished using the following weak classifier,



This weak classifier, which separates using the y-axis value, could be chosen because it succeeds in classifying 3 out of 5 samples. There is no classifier which separates using only a single y-value or x-value, which succeeds in separating 4 or more samples, and therefore this classifier could be chosen.

Any classifier which separates using only a single x-value, will yield the same classification for all the left samples, and therefore will be mistaken for at least 2 samples (and therefore correct for at most 3 samples).

Any classifier which separates only using a single y-value, will be mistaken on one of the '+' and '–' samples with the same y-value, and in addition, because there is two '-' samples in between two '+' samples (in respect to the y-values), any y-value weak classifier will be mistaken over at least (another) one sample. Therefore, any y-value weak classifier will be mistaken over at least <u>two</u> samples (correct for at most 3 samples).

Any of the other figures could not be the result of AdaBoost with a weak classifier,

- **(b)** – as described above, there is no weak classifier for the given samples and features, which will accomplish less then 2 incorrect classifications, but according to figure **(b)**, the classifier obtained at the first iteration only classifies incorrectly a single sample, therefore it is impossible to achieve this figure after a single iteration.
- **(c)** –The classifier must successfully separate <u>most</u> of the left samples, and because all of them got the same x-value, the classifier must separate them using the y-value.
  The classifier must chose a y value that causes the two middle left "-" samples to be classified differently, therefore the chosen y value must be in between them, and indicate that all points beneath it are "+".
  On the other hand, the upper left "+" is classified currently, and also the "-" beneath it, and therefor the chosen y value should be in between those two samples, and indicate that all points above it are "+".
  Therefore, The chosen classifier must indicate that all points above and beneath it are "+", which is impossible according to the stump classifiers class.
- **(d)** – the weak classifier that is chosen will be mistaken over 3 out of the 5 samples, but there exists a classifier which successfully classifies 3 out of the 5 samples correctly, for example the one we described above, or for example a classifier which returns '+' for any positive y-value. Therefore, the AdaBoost algorithm, which is a greedy algorithm, will not chose the classifier which resulted in figure **(d)**.