CMPE 257 HW1
Bo An
010695035

1. Prove that selecting the hypothesis h that maximizes the likelihood $\prod_{n=1}^{N} P(y_n|x_n)$ is equivalent to minimizing the cross-entropy error $E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N} ln(1 + e^{-y_n w^T x_n})$

   Likelihood f(x) = $P(y_n|x_n)$ $P(y_n|x_n) = \{f(x)\, for\, y = +1\, or\, 1 - f(x)\, for\, y = -1$
   If $h \approx f$ then the likelihood(h)≈probability using f
   g = argmax likelihood(h)

   $$max_h\ likelihood(logistic\ h)\ \alpha \prod_{n=1}^{N} h(y_n x_n)$$
   $$max_h\ likelihood(logistic\ h)\ \alpha \prod_{n=1}^{N} \theta(y_n w^T x_n)$$
   $$max_w\ ln \prod_{n=1}^{N} \theta(y_n w^T x_n)$$
   $$min_w\ \frac{1}{N} \prod_{n=1}^{N} - ln\ \theta(y_n w^T x_n)$$
   $$\theta(s) = \frac{1}{1+e^{(-s)}}\ :\ min_w\ \frac{1}{N} \prod_{n=1}^{N} ln\ (1 - e^{(-y_n w^T x_n)})$$
   $$E_{in}(w) = \frac{1}{N} \prod_{n=1}^{N} ln\ (1 - e^{(-y_n w^T x_n)})$$

2. Derive the gradient of the in-sample error used in the gradient descent algorithm

   From problem 1, we know that $E_{in}(w) = \frac{1}{N} \prod_{n=1}^{N} ln\ (1 - e^{(-y_n w^T x_n)})$
   We can consider $1 - e^{(-y_n w^T x_n)}$ as x and $-y_n w^T x_n$ as y
   To derive the gradient of the in-sample error we get:

   $$\frac{dE_{in}(w)}{dw_i} = \frac{1}{N} \sum_{n=1}^{N} (\frac{d\ ln\ (x)}{dx})(\frac{d\ ((1+e^y)}{dy})(\frac{d - y_n w^T x_n}{dw_i})$$
   $$= \frac{1}{N} \sum_{n=1}^{N} (\frac{1}{x})(e^y)(-y_n x_{n,i})$$
   $$= \frac{1}{N} \sum_{n=1}^{N} (\frac{1}{x})(e^y)(-y_n x_{n,i})$$
   $$= \frac{1}{N} \sum_{n=1}^{N} (\frac{e^y}{1+e^y})(-y_n x_{n,i})$$

   Because $\theta(s) = \frac{1}{1+e^{(-s)}}$
   $$= \frac{1}{N} \sum_{n=1}^{N} \theta(-y_n w^T x_n)(-y_n x_{n,i})$$

3.

a. $(x_1 - 3)^2 + x_2 = 1$

$(x_1 - 3)^2 + x_2 - 1 = 0$

$x_1^2 - 6x_1 + 9 + x_2 - 1 = 0$

$x_1^2 - 6x_1 + x_2 + 8 = 0$

And from the problem description, I get that $\Phi_2(x) = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)$

I can get that $\widehat{w}\ in\ Z = (8, -6, 1, 1, 0, 0)^T$

b.

$(x_1 - 3)^2 + (x_2 - 4)^2 = 1$

$x_1^2 - 6x_1 + 9 + x_2^2 - 8x_2 + 16 - 1 = 0$

$x_1^2 - 6x_1 + x_2^2 - 8x_2 + 24 = 0$

And from the problem description, I get that $\Phi_2(x) = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)$

I can get that $\widehat{w}\ in\ Z = (24, -6, -8, 1, 0, 1)^T$

c.

$2(x_1 - 3)^2 + (x_2 - 4)^2 = 1$

$2x_1^2 - 12x_1 + 18 + x_2^2 - 8x_2 + 16 - 1 = 0$

$2x_1^2 - 12x_1 + x_2^2 - 8x_2 + 33 = 0$

And from the problem description, I get that $\Phi_2(x) = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)$

I can get that $\widehat{w}\ in\ Z = (33, -12, -8, 2, 0, 1)^T$

Reference:

- https://github.com/kgourgou/Linear-Perceptron/blob/master/Perceptron-Algorithm.ipynb
- https://www.youtube.com/watch?v=OVHc-7GYRo4
- https://www.youtube.com/watch?v=4Gac5I64LM4
- https://www.youtube.com/watch?v=WW5k_3JhxAY
- https://www.youtube.com/watch?v=4J1ccdYRhmc
- https://en.wikipedia.org/wiki/Perceptron
- https://github.com/kgourgou/Linear-Perceptron
- https://github.com/topics/perceptron-learning-algorithm
- https://www.youtube.com/watch?v=kft1AJ9WVDk
- http://book.caltech.edu/bookforum/
- https://medium.com/@thomascountz/19-line-line-by-line-python-perceptron-b6f113b161f3
- https://medium.com/@thomascountz/perceptron-implementing-and-part-2-84bfb1f46597
- https://medium.com/@thomascountz/perceptron-implementing-and-part-3-84bfb1f46597
- https://medium.com/@thomascountz/perceptron-implementing-and-part-1-84bfb1f46597
- https://www.youtube.com/watch?v=t2ym2a3pb_Y
- https://www.youtube.com/watch?v=5g0TPrxKK6o
- https://medium.com/@thomascountz/tdd-ing-a-perceptron-in-ruby-by-implementing-and-part-1-234c9527ec66
- https://maviccprp.github.io/a-perceptron-in-just-a-few-lines-of-python-code/
- https://livebook.manning.com/book/grokking-machine-learning/4-3-how-to-find-a-good-classifier-the-perceptron-algorithm/v-4/18
- https://www.csie.ntu.edu.tw/~htlin/course/ml13fall/doc/08_handout.pdf
- https://www.csie.ntu.edu.tw/~htlin/course/
- http://www.vynguyen.net/2016/03/23/learning-from-data-a-short-course-exercise-1-3/
- http://book.caltech.edu/bookforum/
- https://github.com/aroques/perceptron-learning
- https://github.com/VitaminDhruv/Pocket-Algorithm
- https://github.com/anupamish/Perceptron-Pocket-Algorithm
- https://github.com/mujiezha/Regression_Algorithms_4
- https://github.com/anupamish/Perceptron-Pocket-Algorithm/projects

- https://github.com/simrantinani/Pocket-Learning-Algorithm-from-a-scratch-in-R
- https://github.com/thiagodnf/perceptron
- https://github.com/saikirankoppu/MachineLearningProj
- https://gist.github.com/GGYIMAH1031/8b7b114f1167789077f83a1d09a83353
- https://github.com/kylerlittle/perceptron

- *And I discussed Hw with my teammates.*