Introduction to Machine Learning
with Applications in Information Security

Errata

November 15, 2018
1. Pages 15, 16, 17, 20, and 21: Change $O_0$ to $𝒪_0$.

2. Page 17, last line of Section 2.5.2: The description just above the displayed equation is correct, but the expression
\[ \tilde{X}_t = \max_i \gamma_t(i) \]
is not right, as $\tilde{X}_t$ is the state at which this maximum occurs, not the maximum value of $\gamma_t(i)$.

3. Page 26, number 5: Since we are using the scaled $\alpha$ and $\beta$, there is no need to compute $\text{denom}$ and normalize by it, as we will always have $\text{denom} = 1$. This is also true in the “special case” of $\gamma_{T-1}(i)$.

4. Page 27, number 6: The given pseudo-code will work, but when re-estimating $A$ and $B$, the $\text{denom}$ does not depend on $j$, so there is no need to re-compute it for each $j$. Hence, the following code should be more efficient for re-estimating $A$, especially in cases where $N$ is large.

```plaintext
// re-estimate A
for i = 0 to N - 1
    denom = 0
    for t = 0 to T - 2
        denom = denom + $\gamma_t(i)$
    next t
    for j = 0 to N - 1
        numer = 0
        for t = 0 to T - 2
            numer = numer + $\gamma_t(i, j)$
        next t
        $a_{ij} = \frac{\text{numer}}{\text{denom}}$
    next j
next i
```

An analogous modification should be made when re-estimating $B$. Some minor additional savings (again, in terms of the $\text{denom}$ calculation) might be obtained by re-estimating $A$ and $B$ within the same loop.

5. Page 32, problem 11: Although not an error, it is worth noting that an alternative approach for part c) would be to train an HMM on English text with $N = 26$ and use the resulting $A$ matrix to solve part d). Then the key in part d) could be determined by matching the rows of the $B^T$ determined in part c) with the rows of the $B^T$ matrix computed from the ciphertext in part d). Similar comments apply to problem 14.
6. Page 41, first paragraph: Change “... insert-to-match transitions $a_{I,M}$, delete-to-insert transitions $a_{D,I+1}$, and so on” to “... insert-to-match transitions $a_{I,M+1}$, delete-to-insert transitions $a_{D,I}$, and so on”.

7. Page 46, equation (3.4): The pairwise alignment dynamic program is incorrect. For the case of a linear gap penalty function $g(x) = ax$, the dynamic program is particularly simple. In this case, there is no need to keep track of the number of gaps, since each gap carries a uniform penalty of $a$, and hence we can eliminate the $G(i,j)$ matrix. Then the initialization is simply

$$F(0,j) = aj \quad \text{and} \quad F(i,0) = ai$$

and the dynamic program recurrence is

$$F(i,j) = \max \begin{cases} F(i-1,j-1) + s(X_i,Y_j) \\ F(i-1,j) + a \\ F(i,j-1) + a \end{cases}$$

The case of an affine gap penalty is slightly more complex—see the article at http://pages.cs.wisc.edu/~bsettles/ibs08/lectures/02-alignment.pdf for a detailed explanation.

8. Page 59, problem 3: Assume that the symbols that actually appear are the only possible symbols.

9. Page 70, 4th line from bottom: Change “… means, then $x_i - \mu_x$ and $y_i - \mu_y$ will both be of the same sign …” to “… means, then $x_i - \mu_x$ and $y_i - \mu_y$ will both be of the same sign …”.

10. Page 85, equation (4.11): The last row of the $\Delta$ matrix in (4.11) is incorrect. The correct version of the last row is

$$\begin{array}{cccc} -0.7523 & 1.3229 & 0.2109 & -0.7815 \end{array}$$

11. Page 85, line below the displayed equation $Y = (y_1 \ y_2 \ y_3 \ y_4 \ y_5 \ y_6)^T$: Change “We first determine $\tilde{Y} - \mu$,...” to “We first determine $\tilde{Y} = Y - \mu$,...”

12. Page 86, middle of the page: The second $W$ and the associated $\varepsilon_{1i}$ are incorrect. Change

$$W = \begin{pmatrix} \tilde{Y} \cdot u_1 \\ \tilde{Y} \cdot u_2 \end{pmatrix} = \begin{pmatrix} -0.2256 \\ -0.8712 \end{pmatrix}$$

to

$$W = \begin{pmatrix} \tilde{Y} \cdot u_1 \\ \tilde{Y} \cdot u_2 \end{pmatrix} = \begin{pmatrix} -3.5122 \\ 0.4033 \end{pmatrix}$$
and change

\[ \varepsilon_1 = 2.57, \varepsilon_2 = 2.07, \varepsilon_3 = 2.48, \varepsilon_4 = 2.73 \]

to

\[ \varepsilon_1 = 2.66, \varepsilon_2 = 4.79, \varepsilon_3 = 1.81, \varepsilon_4 = 6.16 \]

and change “… score(\(Y\)) = 2.07 …” to “… score(\(Y\)) = 1.81 …”.

13. Page 91, problem 13: This problem is not clearly stated. To (hopefully) clarify, for part a), use the unit eigenvectors \(u_1\) and \(u_2\) in (4.10) and compute the weight vector for each of the malware samples \(M_i\), and collect these into a \(\Delta\)-like matrix, and call it, say, \(\tilde{\Delta}\). For part b), do the same as part a), except using the benign samples provided, and call the resulting matrix \(\nabla\). For part c), compute each weight vector for \(Y_i\) as in parts a) and b). Then classify \(Y_i\) as malware if it is closer to a column of \(\tilde{\Delta}\) than to any column of \(\nabla\); otherwise classify it as benign. Note that in every case (for all three parts), you subtract the mean vector (4.8) and you use the unit eigenvectors in (4.10). Hence, there is no need to compute any additional eigenvectors or eigenvalues to solve this problem.

14. Page 97, second sentence: Remove “the” from this sentence so that it reads, “Consequently, we must use preprocessed training data that includes labels.”

15. Page 114, equation (5.20): For some unknown reason, \LaTeX{} decided to label two equations as (5.20) on this page. The reference to (5.20) in the last full paragraph on this page refers to the second equation (5.20).

16. Page 125, Problem 1: Remove the phrase “subject to the constraint \(\lambda \geq 0\)” from the last sentence of the problem statement.

17. Page 125, Problem 4: This problem refers to equation (5.20), but for some bizarre reason there are two equations labeled as (5.20) on p. 114 (see errata number 15). This problem is referring to the first equation (5.20), which is near the top of p. 114.


19. Page 130, Problem 15, part c): Change the first sentence from “Graph the separating hyperplane corresponding to your solution in part b) and on a separate graph, plot your solution to part c).” to read “Graph the separating hyperplane corresponding to your solution in part a) and on a separate graph, plot your solution to part b).”.

20. Page 132, last sentence: Change “… solve the unknowns” to “… solve for the unknowns”.

21. Page 149, the summation for \(E\): Change \(E = \sum_{i=1}^{K} M_j E_j\) to \(E = \sum_{j=1}^{K} M_j E_j\).

22. Page 150, Figure 6.9: Note that these entropy calculations use the natural logarithm.
23. Page 155, first line: Change “Since $\mathcal{L}$ is an increasing function...” to “Since $\log$ is an increasing function...”.

24. Page 179, Section 7.3, Neural Networks: I’ve expanded this into a chapter-length tutorial on deep learning, which can be found at https://www.cs.sjsu.edu/~stamp/ML/files/ann.pdf. I highly recommend this tutorial, as it not only provides a detailed introduction to the all-important topic of deep learning, but it also highlights many interesting connections to other topics in the book.

25. Page 183, Section 7.4.2, AdaBoost: Regrettably, this section was never properly proofread. I’d recommend that in place of this section (and problem 3 at the end of the chapter), you should read my tutorial on AdaBoost at https://www.cs.sjsu.edu/~stamp/ML/files/ada.pdf, which is a corrected and expanded version of Section 7.4.2. If you insist on reading Section 7.4.2, fix the following issues.

- Page 184: The notation in Table 7.1 is not consistent with that used in problem 3 at the end of the chapter. To make this table consistent with problem 3, we will interpret the +1 or −1 in row $i$ and column $j$ as being the classification assigned by $c_j$ to data point $X_i$. The wording in the paragraph below Table 7.1 needs to be modified accordingly.
- Page 184: In the paragraph below Figure 7.1, change “row” to “column” in the first and second sentences.
- Page 184: To be consistent with the notation on the next two pages, change each “a” in the two displayed equations at the bottom of the page to “$\alpha$”.
- Page 184: The sign of the classifier $C_m(X_i)$ determines the classification, i.e., if $C_m(X_i) < 0$ then $X_i$ is classified as −1; otherwise $X_i$ is classified as +1.
- Page 185: At the initial step of the algorithm, set each weight to 1, that is, $w_i = 1$ for $i = 1, 2, \ldots, n$.
- Page 186: The summary of the algorithm (numbered as 1, 2, and 3 in the middle of the page) mentions the “number of misses” and “number of hits.” These are actually weighted misses and weighted hits, where the weights are the $w_i$.

26. Page 198, near the bottom of the page: Change

Minimize: $J(w) = -\frac{1}{2} w^T S_B w$

Subject to: $\frac{1}{2} w^T S_W w = 1$

to

Minimize: $-\frac{1}{2} w^T S_B w$

Subject to: $\frac{1}{2} w^T S_W w = \frac{1}{2}$
Then the $w$ that solves this minimization problem will be the same as the $w$ that solves the maximization problem in (7.11).

27. Page 199, equation (7.15): There is no need to consider the dual problem. Instead, we observe that

$$S_W^{-1}S_B w = \lambda w$$

directly implies that we want to select the dominant eigenvalue. This follows, since our goal is to maximize the between-class scatter (which implies that the effect of matrix $S_B$ must be “large”), and minimize the within-class scatter (which implies that the effect of $S_W^{-1}$ is also “large”). Since $w$ is a unit eigenvector, the dominant eigenvalue $\lambda$ gives us the optimal result.

28. Page 215, problem 7: Delete parts b) and c) of this problem and replace with the following:

b) Explain why we must select the dominant eigenvalue in (7.14).