Machine Learning From Data Assignment 1

Greg Stewart

September 10, 2018

Exercise 1.3

The weight rule in (1.3) has the nice interpretation that it moves in the direction of classifying $x(t)$ correctly.

(a) *Show that $y(t)\mathbf{w}^T(t)\mathbf{x}(t) < 0$.*

When $\mathbf{x}$ is misclassified, by definition it must be the case that

$$\mathbf{w}^T\mathbf{x} < 0.$$

And also due to misclassification, we know that

$$y(t) \neq \text{sign}(\mathbf{w}^T\mathbf{x})$$

which necessarily means $y(t) = +1$. Thus we have that

$$y(t)\mathbf{w}^T\mathbf{x} < 0$$

(b) *Show that $y(t)\mathbf{w}^T(t+1)\mathbf{x}(t) > y(t)\mathbf{w}^T(t)\mathbf{x}(t)$.*

We start by noting that

$$\mathbf{w}(t+1) = \mathbf{w}(t) + y(t)\mathbf{x}(t).$$

So we can rewrite the LHS of the expression as follows, and the rest is obvious.

$$\begin{aligned} y(t)[\mathbf{w}(t) + y(t)\mathbf{x}(t)]^T\mathbf{x}(t) &> y(t)\mathbf{w}^T(t)\mathbf{x}(t) \\ y(t)\mathbf{w}^T(t)\mathbf{x}(t) + y(t)[y(t)\mathbf{x}(t)]^T\mathbf{x}(t) &> y(t)\mathbf{w}^T(t)\mathbf{x}(t) \\ y(t)[y(t)\mathbf{x}(t)]^T\mathbf{x}(t) &> 0 \end{aligned}$$

This final result is guaranteed to be true, so the original inequality is true.

(c) *As far as classifying $\mathbf{x}(t)$ is concerned, argue that the move from $\mathbf{w}(t)$ to $\mathbf{w}(t+1)$ is a move 'in the right direction'.*

From the perspective of making the resulting $\mathbf{w}$ "look better," moving to $\mathbf{w}(t+1)$ is a step in the right direction precisely because of the inequality in the previous part. Essentially this move corrects for the misclassification of $\mathbf{x}(t)$, with the hope of making the overall classification of all of $\mathbf{X}$ better. Eventually this method should result in every member of that set being classified in the right way (if there is in fact such a classification), so the move is 'in the right direction.'

Exercise 1.5

Which of the following problems are more suited for the learning approach and which are more suited for the design approach?

(a) *Determining the age at which a particular medical test should be performed*

Learning. There is likely a pattern here, but it may not be obvious or known.

(b) *Classifying numbers into primes and non-primes*

Design. We know the definition of a prime and many ways to calculate them already.

(c) *Detecting potential fraud in credit card charges*

Learning. Fraud is already a guessing game for credit card companies - perhaps a pattern could be better detected this way.

(d) *Determining the time it would take a falling object to hit the ground*

Design. This has been solvable for a long time thanks to physics. You can just plug in some numbers and calculate already.

(e) *Determining the optimal cycle for traffic lights in a busy intersection*

Learning. There are patterns in traffic but the calculations necessary for this particular problem are (I imagine) quite difficult. Using learning to come up with a model for optimal cycles could be useful.

Exercise 1.6

For each of the following tasks, identify which type of learning is involved and the training data to be used. If a task can fit more than one type, explain how and describe the training data for each type.

(a) *Recommending a book to a user in an online bookstore*

Supervised learning. Data could include dimensions like books owned by people, whether they finish those books, the length of the books, and the ratings given to those books, and correspond to the other books which a person buys & reads.

(b) *Playing tic tac toe*

Reinforcement learning. Every time a set of moves is made that results in losing, avoid that particular set of moves.

(c) *Categorizing movies into different types*

Supervised / Unsupervised.

Supervised: Training data would include qualities of a movie (studio; director; writers; length; ubiquity at release; etc) with $\mathbf{y}$ being the genre / other category type.

Unsupervised: Training data could include all the same things as before, save for the genre, and the algorithm would attempt to classify movies based on these. They may not end up being split into genres, but some other type of categorization.

(d) *Learning to play music*

Reinforcement / Supervised learning.

Reinforcement: Whenever a ”wrong” note / rhythm / cadence is played, note its wrongness.

Supervised: Data could include an input of real music, or MIDI input, and output of its correctness. There is of course a lot more that goes into learning music so this would be quite basic.

(e) *Credit limit: Deciding the maximum allowed debt for each bank customer*

Supervised learning. Training data would include customer credit info (e.g. debt, credit score, income, etc) and the credit limit that they were given.

Exercise 1.7

For each of the following learning scenarios in the above problem, evaluate the performance of g on the three points in $\mathscr{X}$ outside $\mathscr{D}$. To measure the performance, compute how many of the 8 possible target functions agree with g on all three points, on two of them, on one of them, and on none of them.

(a) *$\mathscr{H}$ has only two hypotheses, one that always returns '●' and one that always returns '○', The learning algorithm picks the hypothesis that matches the data set the most.*

The algorithm will pick one that returns all ● because that output makes up the majority of $\mathbf{y}$.

all three. 1/8. g only agrees with either f_8.

only two. $4/8 = 1/2$. g will still agree with f_8, along with two points of f_4, f_6, f_7.

only one. 7/8. g will agree with one point of all f_i's except f_1.

none. 1/8. g only disagrees with all the points of 1 f, which is f_1.

(b) *The same $\mathscr{H}$, but the learning algorithm now picks the hypothesis that matches the data set the least.*

The algorithm will pick the g that returns all ○ since that matches the known data the least.

all three. 1/8. f_1 matches g.

only two. $4/8 = 1/2$. f_1, f_2, f_3, f_5 match g on two points.

only one. 7/8. All except f_8 match g on one point.

none. 1/8. f_8 matches g on no points.

(c) *$\mathscr{H} = \{XOR\}$ (only one hypothesis which is always picked), where XOR is degined by $XOR(\mathbf{x}) = \bullet$ if the number of 1's in $\mathbf{x}$ is odd and $XOR(\mathbf{x}) = \circ$ if the number is even.*

all three. 1/8. Only f_2 matches the outcome for g.

only two. $4/8 = 1/2$. f_1, f_2, f_4, f_6 all match g on two points.

only one. 7/8. $f_1, f_2, f_3, f_4, f_5, f_6, f_8$ match g on one point.

none. 1/8. Only f_7 matches g on no points.

(d) *$\mathscr{H}$ contains all possible hypotheses (all Boolean functions on three variables), and the learning algorithm picks the hypothesis that agrees with all training examples, but otherwise disagrees the most with the XOR.*

all three. 1/8. Only agrees with f_7.

only two. 4/8. f_3, f_5, f_7, f_8 agree with two points from g.

only one. 7/8. Only f_2 does not have one agreement with g.

none. 1/8. Only f_2 agrees with none of g.