Machine Learning Binary classifier home work

Due: Start of class Feb

**DATA SET 1**

|  |  |
| --- | --- |
| Value | Type |
| 1 | Red |
| 1 | Red |
| 2 | Red |
| 4 | Blue |
| 5 | Red |
| 7 | Blue |
| 10 | Blue |
| 14 | Blue |
| 15 | Blue |
| 16 | Blue |

1. Consider a binary classifier with one parameter. Prediction is Blue if the value is greater than 3.
   1. Create the confusion matrix for data set 1.

|  |  |
| --- | --- |
| 6 | 0 |
| 1 | 3 |

* 1. Compute accuracy, precision, sensitivity, and specificity.

9/10

1

6/7

1

* 1. Compute the F and G score

.923

.9258

1. Consider a binary classifier with one parameter. Prediction is Blue if the value is greater than 6.
   1. Create the confusion matrix for data set 1.

|  |  |
| --- | --- |
| 5 | 0 |
| 4 | 1 |

* 1. Compute accuracy, precision, sensitivity, and specificity.

3/5

1

5/9

1

* 1. Compute the F and G score

.714

.745

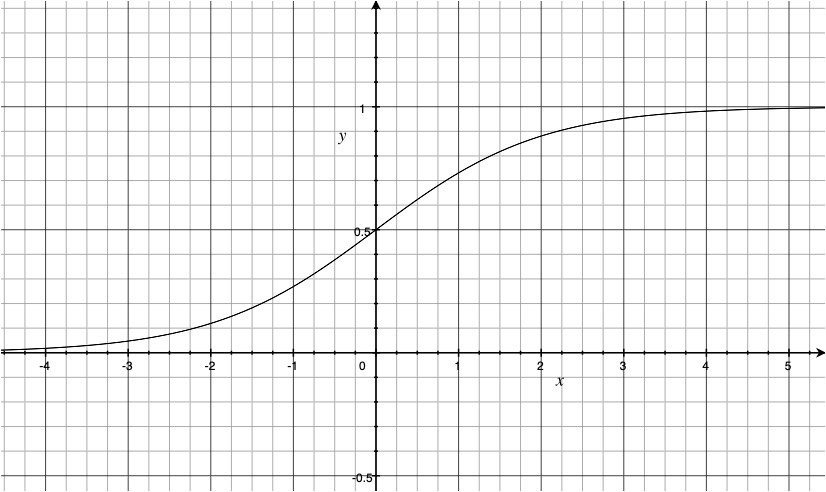
1. Which of the previous classifiers would you prefer for the data set 1? Explain your choice.

I would prefer the first classifier because it had higher g and f scores. These scores seem to determine the overall score of the classifiers and the best one would have the highest score because you would want the highest sub scores in a given f and g score.

1. The sigmoid function is defined as



It has an S shape and limits to 0 and 1.



This makes it suitable to use as a classifier with two parameters that returns a probability. W specifies the steepness of the transition. (Larger W is steeper). K specifies the middle of sigmoid. (Where the probability is 0.5)



a) Fill in the following table using . If p is > ½ predict Blue

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | Type | probability | prediction | Log-Loss |
| 1 | Red | 0.017986209962091562 | Red | 0.01814992791780973 |
| 1 | Red | 0.017986209962091562 | Red | 0.01814992791780973 |
| 2 | Red | 0.11920292202211755 | Red | 0.12692801104297252 |
| 4 | Blue | 0.8807970779778825 | blue | 0.12692801104297238 |
| 5 | Red | 0.9820137900379085 | blue | 0.01814992791780973 |
| 7 | Blue | 0.9996646498695335 | blue | 0.00033540637289577373 |
| 10 | Blue | 0.9999991684719723 | blue | 8.315283733756919e-07 |
| 14 | Blue | 0.9999999997210532 | blue | 2.7894675462274766e-10 |
| 15 | Blue | 9999999999622486 | blue | 3.775135759625163e-11 |
| 16 | Blue | 0.9999999999948909 | blue | 5.1091353370355594e-12 |

b) Compute the average log-loss

0.03429356045138339

c) Fill in the following table using . If p is > ½ predict Blue

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | Type | probability | prediction | Log-Loss |
| 1 | Red | 4.5397868702434415e-05 | Red | 4.539889921682063e-05 |
| 1 | Red | 4.5397868702434415e-05 | Red | 4.539889921682063e-05 |
| 2 | Red | 0.00033535013046647827 | Red | 0.00033540637289577373 |
| 4 | Blue | 0.017986209962091562 | Red | 0.01814992791780973 |
| 5 | Red | 0.11920292202211755 | Red | 0.12692801104297252 |
| 7 | Blue | 0.8807970779778825 | Blue | 0.12692801104297238 |
| 10 | Blue | 0.9996646498695335 | Blue | 0.00033540637289577373 |
| 14 | Blue | 0.9999998874648379 | Blue | 1.1253516840995767e-07 |
| 15 | Blue | 0.9999999847700205 | Blue | 1.5229979615740706e-08 |
| 16 | Blue | 0.9999999979388464 | Blue | 2.0611535832696244e-09 |

d) Compute the average log-loss

0.03030752115269793

e) Based on the average log-loss, which classifier do you prefer?

Considering that less log loss is better I would pick the second classifier because it is a tiny bit less than the other.

**DATA SET 2**

|  |  |
| --- | --- |
| Value | Type |
| 1 | Red |
| 1 | Red |
| 2 | Red |
| 3 | Blue |
| 4 | Red |
| 5 | Blue |
| 7 | Blue |
| 10 | Blue |
| 14 | Red |
| 15 | Blue |
| 16 | Red |
| 19 | Red |
| 20 | Red |

1. Consider a binary classifier with two parameters. Prediction is Red if the value is less than 3 or greater than 15. Blue, otherwise.
   1. Create the confusion matrix for data set 2.

|  |  |
| --- | --- |
| 6 | 1 |
| 4 | 2 |

* 1. Compute accuracy, precision, sensitivity, and specificity.

8/13

.8571

.6

2/3

* 1. Compute the F and G score

.7057

.7171192