Meg Schroeder

DS 4400

* 1. Random Forest and AdaBoost. Random forest uses boostrap sample of data and averages the predictions of all the trees. Adaboost is an ensemble of many weak learners (ex. Decision stump) and weights each training example by how incorrect it was as opposed to taking the average
  2. Ensemble learning reduces the variance whereas decision trees have 0 bias resulting in a naturally high variance. This means that Random Forest and AdaBoost have a better bias/variance tradeoff than base decision trees.
  3. Random forest uses bagging or random input vectors which use a bootstrap sample of the data chosen at random for each tree. AdaBoosting uses many separate weak learners that are not associated with one another.
  4. Convolution neural networks have a convolution layer(linear operation that uses local information) for feature reduction and is good for data with natural topology, whereas FFNNs just use nonlinear activation functions.
  5. Hidden layers help with feature reduction.
  7. Odor

|  |
| --- |
| No  yes  D  C  yes  No  No  G  U  No  Shape  W  B  yes  3  2  1  Color  Odor |

* 1. TP=2  
     FP=1  
     TN=3  
     FN=1  
     Confusion Matrix:

|  |  |
| --- | --- |
| 2/7 | 1/7 |
| 2/7 | 3/7 |

* 1. ,
  2. ,
  4. so classify as YES
  5. Laplace smoothing:
  7. .
  8. Num params = (d+1)num layers  
     There are 1503 total parameters, because there are 500 inputs, so (500+1)\*3 layers = 1503.   
     The input layer has 501 parameters. The hidden layer has 21(3) = 63 parameters. The Output layer has (11)3 = 33 parameters.
     1. The input is a vector representation of the images
     2. The output layer has 6 nodes, 1 for each class
     3. A possible NN architecture would be  
        Convolution(64 (3,3))  
        MaxPool  
        Convolution(64 (3,3))  
        MaxPool  
        Dense  
          
        Convolution(128 (3,3))  
        MaxPool  
        Convolution(128 (3,3))  
        MaxPool  
        Dense  
          
        Convolution(256 (3,3))  
        MaxPool  
        Convolution(256 (3,3))  
        MaxPool  
        Dense  
        Softmax  
        This is similar to AlexNet, and the neural network I am using for my final project.
  10. The output size of the matrix is 2x2. There are 4 parameters
  11. . This layer has 2x2 = 4 parameters
  12. The output matrix will be a 2x2x10 matrix with 40 parameters
  13. You cannot create a FFNN with no hidden layers to compute this function. The bias must be a positive number for . must be a negative number greater than the bias in order for with a positive bias. must also be a negative number greater than the bias in order for with a positive bias. However, this means that is impossible. If and , then will always be negative.
  14. A hidden layer could be used to compute this function. . The hidden layer takes the output and reverses the sign if it is over a certain threshold.