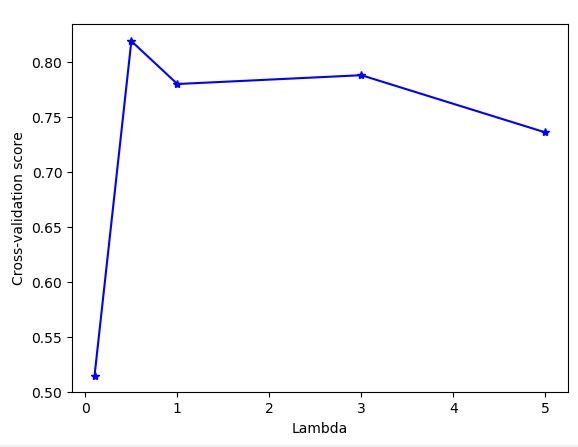
# 1

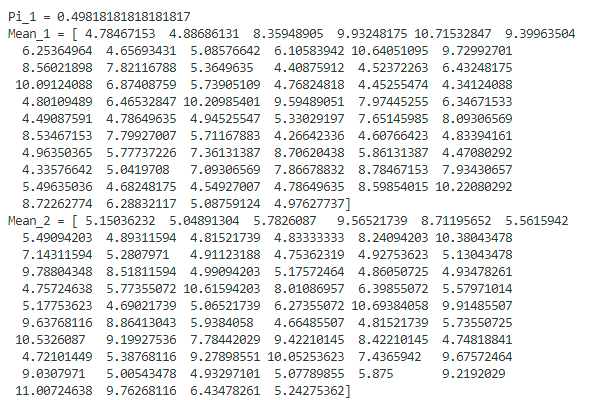
1. Cross-validation accuracy of logistic regression as λ varies

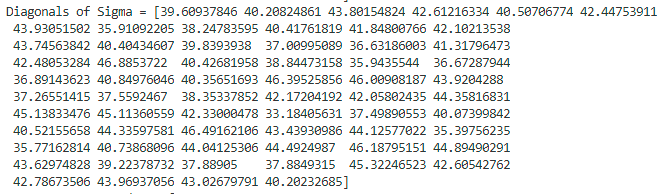


1. Report the accuracy of mixtures of Gaussians and logistic regression (with the best λ for regularization) on the test set.

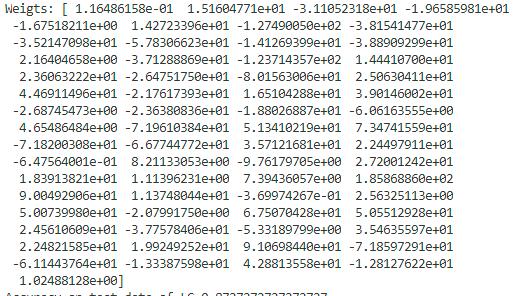
|  |  |  |
| --- | --- | --- |
| Dataset | Gaussian Mixture | Logistic Regression |
| Test Data | 0.89 | 0.87 |

1. Print the parameters π, µ1, µ2, Σ found for mixtures of Gaussian. Since the covariance Σ is quite big, print only the diagonal of Σ. Print also the parameters w, w0 found for logistic regression.





**W +**wo



1. Briefly discuss the results:

* In Gaussian Mixture model, we have π, µ1, µ2, Σ, the number of parameters of which is basically 1+2N + N^2 (N is number of features). In Logistic Regression, we have only **W** and wo, the number of parameters of which is basically N+1. The accuracies of both models on test dataset are comparable.
* In case of Mixture of Gaussians and logistic regression, the separators are line (2D), plane (3D) and hyperplane ( n-dim) while KNN seeks to find curve, or surfaces to separate the data. Mixture of Gaussians and logistic regression are applicable when data are linearly separable while KNN can work well non-linear data.
* Then KNN algorithm in Assignment 1 only accomplished an 81% accuracy. Thus, we can conclude that a linear separator was more suited for this data because both methods that found a linear separator achieved a higher accuracy. We could say that the data was more linearly separable that “K Nearest Neighbour separable”.

# 2

Let ‘s assume a case of two inputs (x1, x2) and one output (y) for the sake of simplicity. So we have w0, w1, and w2

* And : we can encode And function as a threshold perceptron with wo = 1.5 , w1 = 1, w2 = 1
* OR: we can encode OR function as a threshold perceptron with wo = 0.5 , w1 = 1, and w2 = 1
* Exclusive-OR (XOR): we cannot encode XOR function as a threshold perceptron since it is not linearly separable. However, we can use feature mappings to make data linearly separable by creating an extra variable which is product of x1 and x2 (x1\*x2).
* Iff (or XNOR). we cannot encode XOR function as a threshold perceptron since it is not linearly separable. XNOR is the opposite of XOR. So we can use feature mappings to make data linearly separable by creating an extra variable which is product of x1 and x2 (x1\*x2).

1. To answer whether the train set is linearly separable. We use the weight vector from the model with ‘best lambda’ to identify the data which falls above or below the hyper-plane given by **wTx.** if **wTx** > 0, we assign 1 and otherwise 0 to a result vector. Please note that **w0** is augmented inside **w**. Then we can compare the result vector with train labels by computing confusion matrix. If train data is linearly separable, there will not be any misclassification.

Here is the resulting confusion matrix:



There are misclassifications, so we can conclude that the train data is not separable.