Aprendizagem

LEIC IST-UL

RELATÓRIO - MERIT PRIZE CHALLENGE

Grupo 10:

Gabriel Ferreira	107030
Irell Zane	107161

2024/2025 – 1st Semester, P1

1. *Perform logistic regression and indicate the accuracy.*

	Testing Accuracy	Training Accuracy
StandardScaler	98.25%	98.74%
Unscaled	97.66%	95.98%

2. *Perform EM clustering on the training data set with different number k of clusters. Evaluate the quality of the clusterings using Silhouette. Is the number of clusters correlated with the quality of clustering? Which is the optimal k?*

   We used `scikit-learn`'s `GaussianMixture` model to perform EM clustering on the breast cancer dataset features. An evaluation of the clustering quality was made using each solution's silhouette for numbers of clusters ranging from 2 to 10, see Figure 1.

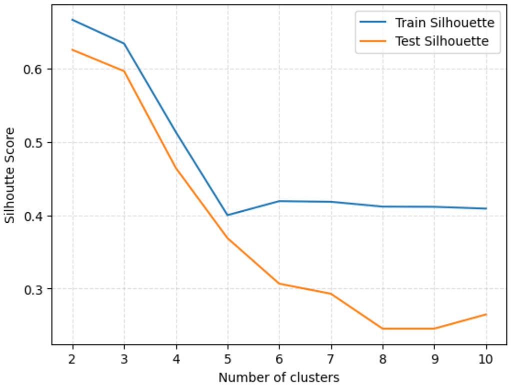


Figure 1: Silhouette of different EM clustering solutions.

   According to the Silhouette score, there is a trend of declining clustering quality as the number of clusters increases in the EM clustering solution. We can also see an increase in the gap in the Silhouette scores between the classifications of the testing and training samples as the number of clusters increases, suggesting a deterioration of the generalization capacity of the clustering solution.

   According to these results, the optimal k is $k = 2$ with a silhouette score of 0.626 for the testing sample, and 0.667 for the training sample.

3. *Map the test set into probability values of the k-clusters. If you have a data point represented by a vector of dimension d, you will map it into a vector of dimension:*

   `prob=em_model.predict_proba(X)`

4. *Perform logistic regression on the mapped data set with the labels of the original test set. Indicate now the accuracy. Is there a relation between the number of clusters, the cluster evaluation and the accuracy of the logistic regression model?*

   We mapped the dataset to the cluster probabilities of the different clustering solutions and performed logistic regression on these transformed features.

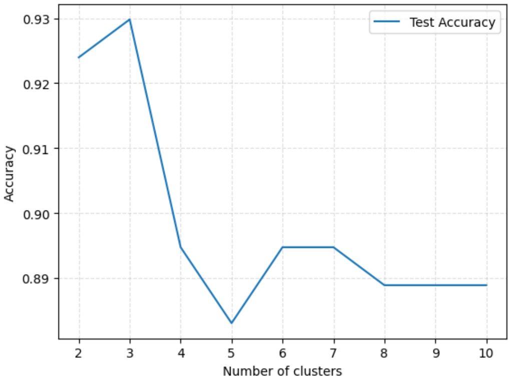


Figure 2: Silhouette of different EM clustering solutions.

Upon analyzing the relationship between the number of clusters (k) and both performance metrics - the Silhouette scores and Logistic Regression accuracy on the mapped dataset - we can see that there is a notable relation: For low numbers of clusters, such as 2 and 3, both metrics show high values, and as the k increases, there is a sharp decline in both the silhouette score and the classification accuracy of the logistic model, which seem to follow the same trajectory, suggesting that a higher quality of clustering, as measured by the Silhouette score, is conducive to a better logistic regression model.

Using $k = 2$ or $k = 3$ the EM algorithm effectively serves as a dimensionality reduction technique, transforming the original high-dimensional feature space (30 features) into a lower dimensional feature space, while maintaining a high accuracy on the logistic regression model.

The accuracy for the model that used clustering with $k = 2$ was 92.4%.

5. *Train an RBF network using the clustering with optimal k from 2).*

   A typical RBF Network looks like this:

   (a) Input layer with the dimension of an observation

   (b) Hidden layer with the dimension of the number of clusters

   (c) Output layer with a single neuron

   Before training an RBF network we need to choose an Radial Basis Function. A Radial Basis Function is any function that depends on the distance, the most commonly used is the Gaussian, which should be appropriate for EM clustering:

$$\phi_k(x) = \phi_k(d(x, c_k)) = exp(-\gamma \cdot d(x, c_k)^2) \tag{1}$$

   Since we are dealing with EM Clusters, we want to measure the distance between the point of the observation and the distribution of cluster 1 and 2, the distance between a point and a distribution is the Mahalanobis distance:

$$d(x, c_k)^2 = (x - \mu)^T \Sigma_k^{-1} (x - \mu) \tag{2}$$

So our Radial Basis Function ends up like so:

$$\phi_k(x) = exp(-\gamma \cdot (x-\mu)^T \Sigma_k^{-1}(x-\mu)) \tag{3}$$

As in 'Machine Learning: A Journey to Deep Learning'[1], we thought the γ should be $\frac{1}{2}$; However when this was tried, the network would always make the same prediction regardless of the input. This was quite unexpected, we speculated that this was a float underflowing issue, arising from the negative exponent being too large resulting into loss of information, but it seems more plausible that the model was unable to learn when the activation values were of very different scales and mostly near zero. Thus we plotted the Accuracy against the Gamma to decrease the exponent and, after tuning this hyperparameter, settled on $\gamma = 0.015$ (maximizing training accuracy), see Figure 3, resulting in a test accuracy of 92.4%.

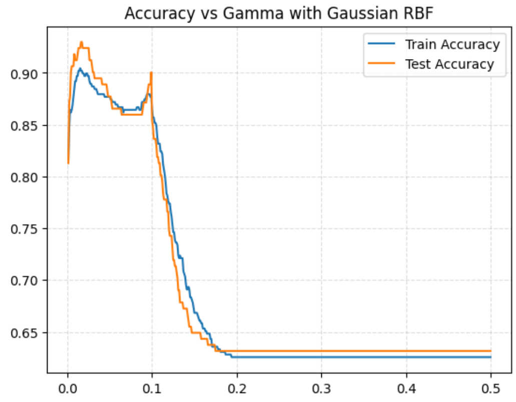
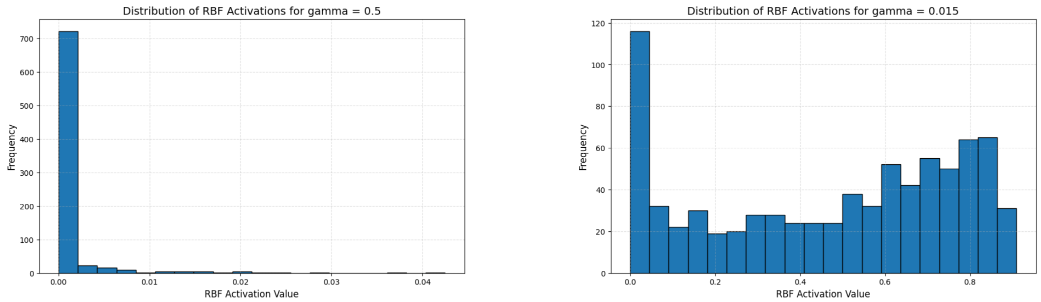


Figure 3: RBF Gamma tuning

Comparing the distribution of the values of the hidden layers in both solutions, we can see that for $\gamma = 0.015$ there is a wider range of values, which have a less concentrated distribution, see Figure 4.



(a) $\gamma = 0.5$

(b) $\gamma = 0.015$

Figure 4: Hidden layer activation value distribution

The distribution seen in Figure 4b will allow for a much more diverse activation for the output layer, but it is still a skewed distribution, and the model is not able to learn anything from observations that have RBF values very close to zero for *both* clusters, tuning the gamma value is also expensive and something we want to avoid. Thus we considered normalizing the values of the hidden layer before the logistic regression. Making a Normalized Radial Basis Function Network.

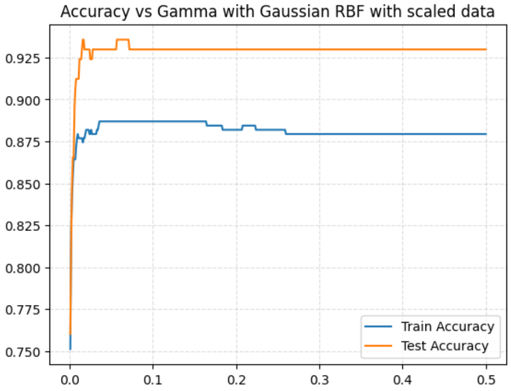
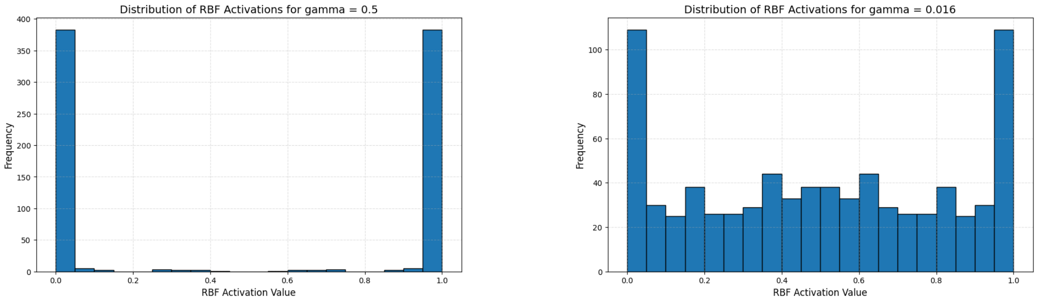


Figure 5: NRBF Gamma tuning

This resulted on a much better testing accuracy, for $\gamma = 0.5$: 92.96%, which is greater than the 92.4% obtained in ex 4. and additionally tuning the gamma stopped being a necessity, or effective at all. If we do maximize train accuracy, we obtain the same 92.96% test accuracy.

Looking at the distributions we can see a much more balanced distribution, since it is normalized, there are more activations, and the model can learn more effectively.

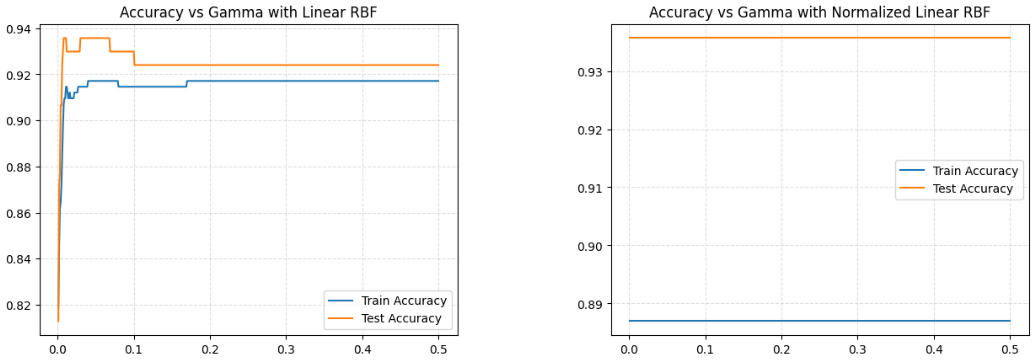


(a) $\gamma = 0.5$

(b) $\gamma = 0.036$

Figure 6: Hidden layer activation value distribution

However Gaussian is not the only effective function, Linear actually seems to perform generally better, see Figure 7.



(a) Linear RBF

(b) Normalized Linear RBF

Figure 7: Linear RBF Accuracy

6. *Discuss your findings.*

References

[1] Luis Sa-Couto and Andreas Miroslaus Wichert, *Machine Learning - A Journey To Deep Learning: With Exercises And Answers*, International series of monographs on physics, World Scientific Pub Co Inc, 2021, ISBN: 9789811234057.