ML and PR Project Report – Part 6

The theme of the current laboratory is logistic regression, with weighted and quadratic variants.

# Non weighted

The first method we use to classify using logistic regression is using its most basic variant (which still includes normalization through the parameter), in which the objective function does not include weights.

The data is trained without weights, but regardless we are interested in analyzing the performance of the classifier over an application represented by an effectife prior of 0.1.

Our data size is decently large, so we don’t necessarily risk overfitting even for small values of .

Immagine che contiene testo, schermata, Diagramma, linea

Descrizione generata automaticamente

Our data does not seem to need normalization, quite the opposite: as we start normalizing more and more, we deviate completely from the minDCF values, which instead remain constant, and provide better results even for a normalization with .

As we predicted, our dataset is good enough to not risk overfitting even without normalization.

If we try to reduce the size of our data by a factor of 1:50 we may start to see some benefits.

Immagine che contiene testo, diagramma, schermata, linea

Descrizione generata automaticamente

As expected, on a smaller dataset is easier to find overfitting, which is effectively reduced by our normalization. Just like the other model though, too much normalization is detrimental.

As for the minDCF instead, we see a slight decrease in cost when the normalization is particularly strong, which is weird, since finding a more optimal threshold when the data is less scattered should prove harder.

# Prior weighted

We want to analyze the same application as before, but this time modifying the objective function, in order to include the information about the actual application. The way we do it is, during the training phase, we weigh the score of every sample based on the label.

The results are the following.

Immagine che contiene testo, schermata, diagramma, Diagramma

Descrizione generata automaticamente

I expected this model to perform better than the non-weighted version, since it’s training and evaluation priors vastly differed, but it seems to do just as good.

# Quadratic

Another possible variant of the logistic regression classifier involves looking for nonlinear decision boundaries, and thus mapping the nonlinear score, to a linear one.

Such a model proves very useful for data that is not clearly separatable through a linear decision, meaning that an increase in performance using this model, could suggest such a distribution in the data.

Immagine che contiene testo, schermata, Diagramma, diagramma

Descrizione generata automaticamente

The model seems to have the same relation with normalization as the other models had, meaning that it doesn’t help much, and degrades completely before the mark.

The actual model reaches DCF values much closer to the minDCF that the other models seen so far (if we stay in the low normalization zone that is), and in terms of the latter, performs better than the other models by a whole 0.1 points. For greater values of though it gains a few points.

# Regularization (centering)

For the regularized linear regression model, affine transformations of the data don’t give the same results, so we want to test that. We will simply use centering of the data using the training data mean.

Immagine che contiene testo, schermata, linea, Diagramma

Descrizione generata automaticamente

The results seem to be the same as the standard non-weighted logistic regressions model, hinting at the fact that our data was already regularized in terms of centering.

# Comparison

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **model ()** | **QLogistic** | **Naïve** | **MVG** | **Logistic** | **PWLogistic** | **Tied** |
| minDCF | 0.243 (=0.03) – 0.259 (=0.001) | 0.257 | 0.263 | 0.361 | 0.362 | 0.363 |
| ranking | 1~ | 2 | 3 | 4 | 4 | 4 |

The first thing that meets the eye when comparing the results from above is that the non quadratic logistic models and the tied model have pretty much the same performance. The common point of linear logistic regression and tied gaussian is that they both generate linear decision boundaries. The fact that these three take last place suggests that our dataset isn’t really well classifiable just through linear models.

Naïve and MVG have similar results, as we have observed many times in the previous laboratories, suggesting that the features actually are uncorrelated.

The quadratic logistic regression model doesn’t fall that far from Naïve and MVG (and meets their performance if we see smaller or greater values of normalization), meaning that the gaussian distribution is effective at generating predictions

# Conclusions

Logistic regression is decent, although definitely not best, for classifying our dataset. The dataset doesn’t seem to need much normalization (, nor regularization, for this process. The regression doesn’t seem to improve through the weighted variant, but does incredibly better when using the quadratic one. The linear logistic regression models match the performance of the tied gaussian model, while the quadratic one meets and bests that of the MVG and Naïve Bayes models.