University of Salerno

.DIEM
Department of Information and Electrical Engineering and Applied Mathematics

Master’s degree in computer engineering

Data Science 30/01/2025 A-H

Project Work
Classification

Group n. **07 – AH**

Student	ID number	e-mail
Apicella Antonio	0622702531	a.apicella97@studenti.unisa.it
Celano Benedetta Pia	0622702558	b.celano1@studenti.unisa.it
Cirillo Francesco Pio	0622702466	f.cirillo36@studenti.unisa.it
Fasolino Alessandra	0622702465	a.fasolino35@studenti.unisa.it

Table of Contents

1. Case 1: perfectly known statistical model

1.1. Computation of the Posterior PMF

$$p(y|x) = \frac{\pi(y)\cdot \ell(x|y)}{\sum \pi(y')\,\ell(x|y')} = \frac{\pi(y)\,\ell(x|y)}{\pi(1)\,\ell(x|y=1) + \pi(-1)\,\ell(x|y=-1)} =$$

$$= \frac{\pi(y)\,\ell(x|y)}{\frac{1}{2}\left(\ell(x|y=-1) + \ell(x|y=1)\right)} \overset{(2)}{=} \frac{\ell(x|y)}{\ell(x|y=-1) + \ell(x|y=1)} =$$

$$= \frac{\ell(x|y)}{\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{(x+1)^2}{2\sigma^2}\right\} + \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{(x-1)^2}{2\sigma^2}\right\}} =$$

$$\overset{(1)}{=} \frac{\exp\left\{-\frac{(x-y)^2}{2\sigma^2}\right\}}{\exp\left\{-\frac{(x+1)^2}{2\sigma^2}\right\} + \exp\left\{-\frac{(x-1)^2}{2\sigma^2}\right\}}$$

(1) $$\ell(x|y=y) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{(x-y)^2}{2\sigma^2}\right\}$$

(2) $$\pi(y) = \pi(-1) = \pi(1) = \frac{1}{2}$$

Figure 1: Computation of the Posterior PMF

The calculation shown in the figure is simply the computation of the posterior using Bayes' rule.

1.2. Graphical Representation of $\boldsymbol{p(y = +1 \mid x)}$ in Two Cases with Different Variance

The lower the variance, the more we expect the classifier to be accurate, as the data dispersion decreases. To avoid choosing extreme values, as requested, the values variance_good = 1 and variance_bad = 2 were assumed.

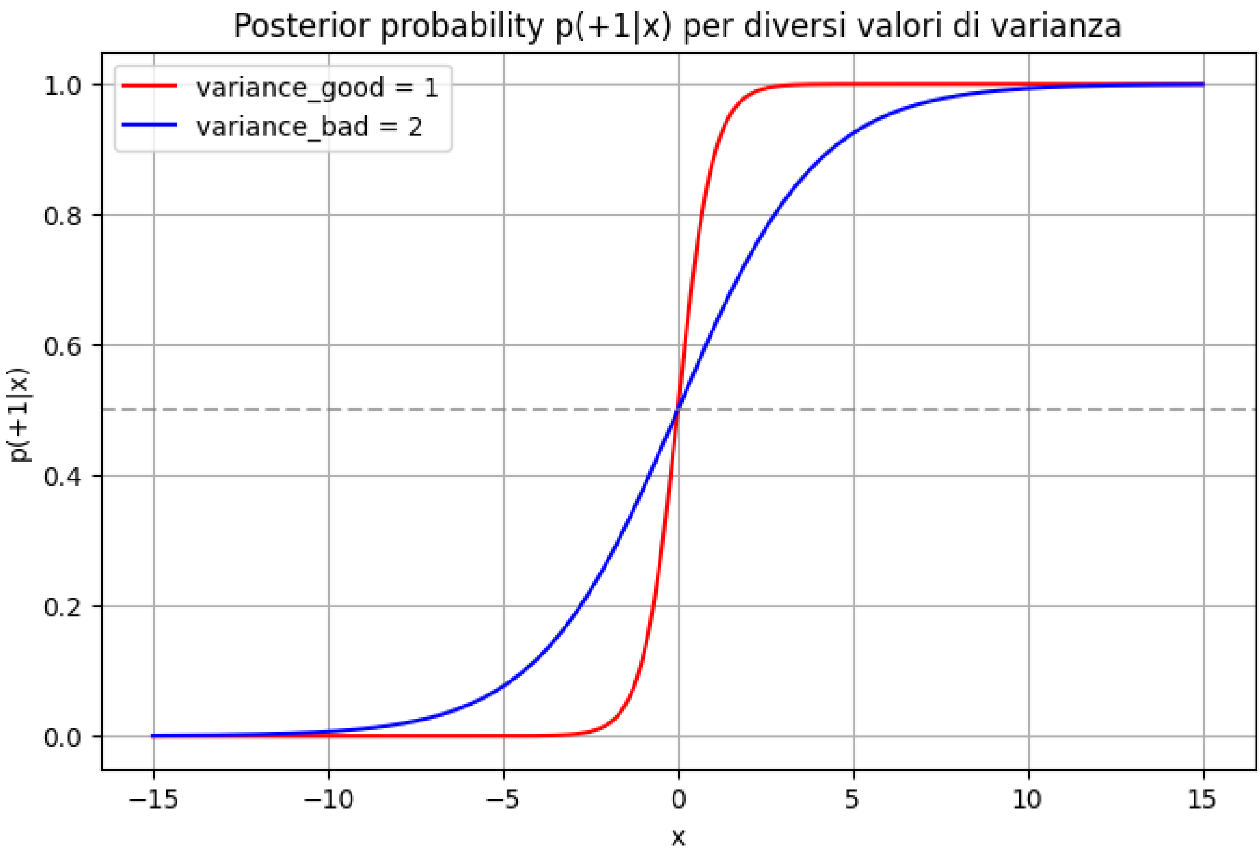

Figure 2: Plot showing the posterior in the case where y = +1

1.2.1. Conclusions Drawn from the Observation of the Graph

We can interpret the plot of $p(y = 1 \mid x)$ (Figure 2) as the graph representing the probability that y is equal to 1, given the specific observation of x that we have.

What we observe is that when the variance is higher (blue plot), and therefore the Gaussian realizations are more spread out, it becomes harder to determine whether a given x belongs to class -1 or class 1 — especially when the values of x are close to the midpoint between the two data-generating distributions.

In fact, for the red plot, the probability that y corresponding to x is 1 quickly shifts from 0 to 1, leaving only a small range of x values (between -1 and 1) where there is uncertainty. On the other hand, in the blue plot, there's a wider range of x values — approximately between $x = -5$ and $x = 5$ — for which the class membership is uncertain.

1.3. Computation of the Empirical Error Through Monte Carlo Simulations

We observe that in the case of higher variance, the estimated error probability is approximately 0.31, whereas in the case of lower variance, the estimated error probability is around 0.16 (roughly half).

This result is in line with expectations — it was indeed predictable that when the realizations of the two Gaussians are less dispersed, it becomes easier to determine which distribution generated the data, thus enabling easier classification with fewer errors.

2. Case 2: supervised classification

2.1. Supervised Classification Using Stochastic Gradient Descent

The beta values are computed both without the aid of a Monte Carlo technique (Figure 3) and with the use of a Monte Carlo technique (Figure 4), in order to compare the quality of the results in these two cases.

In the implementation where the beta values are calculated without using a Monte Carlo technique, we simply generate a dataset and use it for training. In this case, we expect less accurate results than those obtained in the following implementation.

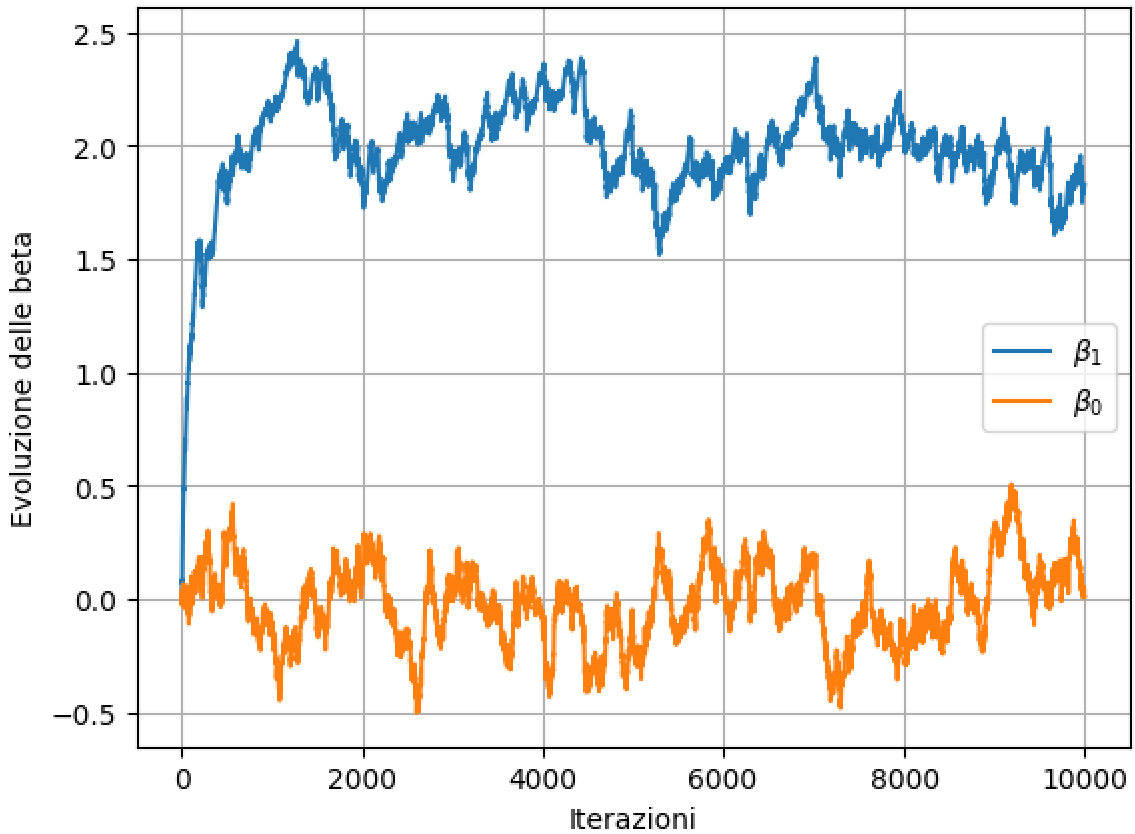

Figure 3: Evolution of the beta values at each iteration not using the Monte Carlo technique

In the following implementation, the beta values are computed using a Monte Carlo technique: at each iteration, a dataset is generated according to the known distributions, and the resulting betas are the averages of those obtained from the different dataset realizations.

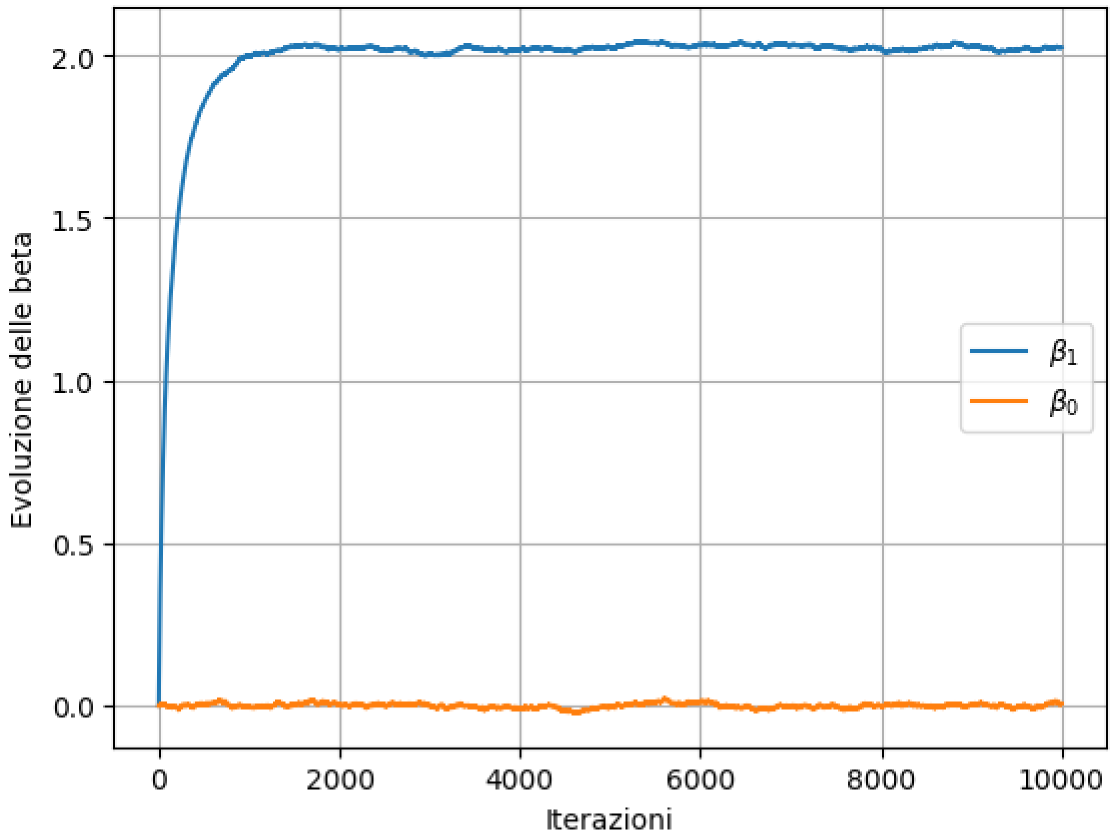

Figure 4: Evolution of the beta values at each iteration using the Monte Carlo technique

We note from the observation of the graphs showing the evolution of the Betas that, thanks to the Monte Carlo iterations, it was possible to obtain much less noisy estimates of the beta values.

2.2. Empirical Estimation of Classifier Performance

The test error was calculated using Monte Carlo iterations for both classifiers obtained, in order to ensure a more accurate estimate.
Below, the results obtained with the betas computed without and with the aid of Monte Carlo iterations during training are compared.

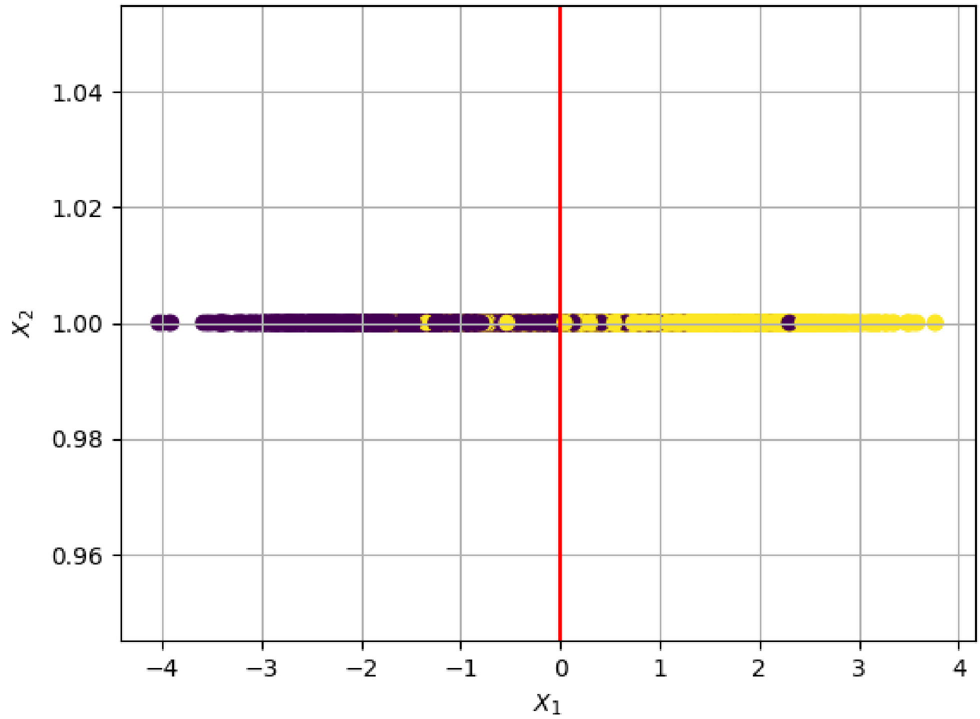

Figure 5: Scatter plot of the model obtained without Monte Carlo iterations

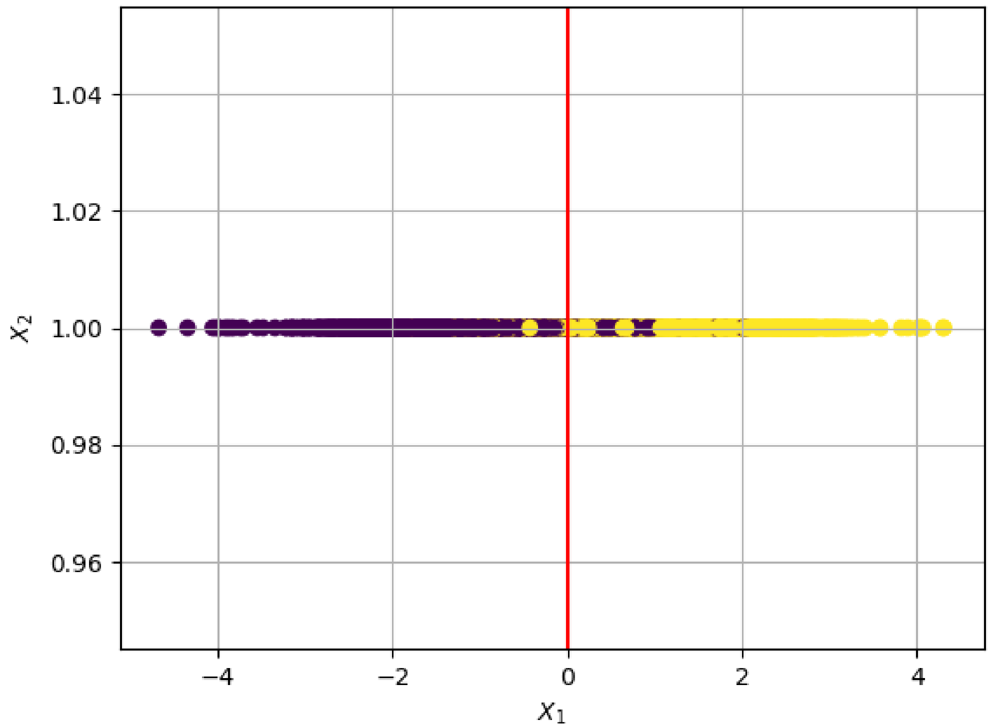

Figure 6: Scatter plot of the model obtained with Monte Carlo iterations

From the scatter plots (Figures 5–6), it is not possible to appreciate a noticeable difference between the two approaches — this is certainly due to the data-rich scenario we have created. However, it is still appreciable that the beta estimates obtained with the aid of Monte Carlo are significantly less noisy.

2.2.1. Comparisons Between the Error Probabilities of the Different Approaches

All the following comparisons are relative, as required by the assignment, to the case with the lowest variance — specifically, a variance equal to 1 was chosen.

Error probability calculated for the MAP classifier with known model	0.15843
Error probability calculated for the classifier obtained with SGD using constant step-size, without Monte Carlo iterations	0.15866
Error probability calculated for the classifier obtained with SGD using constant step-size, with Monte Carlo iterations	0.15847

We note that, although only slightly, the error probability obtained with MAP is lower than those obtained with SGD. Surely, the fact that the difference is small is due to the large number of samples that make up the datasets used for SGD. In a situation that is not "data-rich," the difference would have been more pronounced. MAP, by relying on models, is the optimal criterion and is free from the compromises that arise from dependence on a dataset.

We also note that, between the two error probabilities obtained with SGD, the lower one is the result of using the betas calculated through Monte Carlo iterations. This result is also in line with expectations. By averaging over multiple datasets, results are less affected by the specific characteristics that a certain dataset may have, and thus are improved.

List of Figures