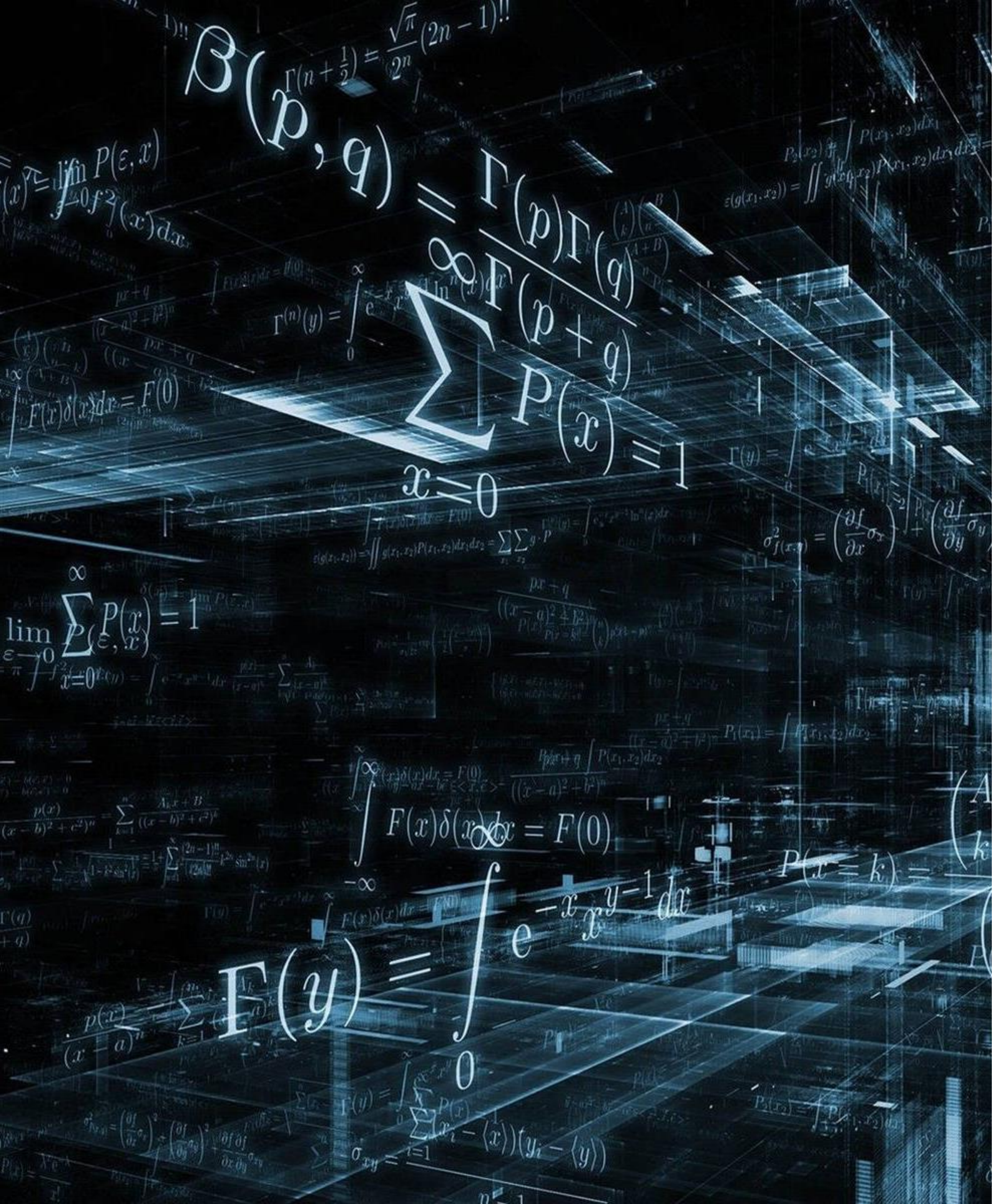


Adjusting the Output of a Classifier to New a Priori Probabilities

Minh Duc Pham, Lorenzo Lattanzi

Sep 2024

Overview

In machine learning, classifiers are often trained on data with class distributions that do not reflect the real-world data. This mismatch can lead to suboptimal classification performance.

Main scenario and solutions

1 **Known the new A Priori Probabilities:**

- Direct Adjustment with Bayes' Theorem

2 **Unknown the new A Priori Probabilities:**

- Confusion Matrix Method
- Expectation-Maximization Algorithm
- Likelihood Ratio Test

Bayes Formula for Known a Priori

By the Bayes Theorem, we have the within-class probabilities:

$$\hat{p}_t(x \mid \omega_i) = \frac{\hat{p}_t(\omega_i \mid x)\hat{p}_t(x)}{\hat{p}_t(\omega_i)} \qquad (2.1)$$

$$\hat{p}(x \mid \omega_i) = \frac{\hat{p}(\omega_i \mid x)\hat{p}(x)}{\hat{p}(\omega_i)} \qquad (2.2)$$

Considering that the $\widehat{p_t}(x \mid \omega_i) = \hat{p}(x \mid \omega_\mathrm{i})$, we define $f(x) = \widehat{p_t}(x)/\hat{p}(x)$, and we find :

$$\hat{p}(\omega_i \mid x) = f(x) \cdot \left(\frac{\hat{p}(\omega_i)}{\hat{p}_t(\omega_i)} \right) \cdot \hat{p}_t(\omega_i \mid x) \qquad (2.3)$$

Since $\sum_{i=1}^{n} \hat{p}(\omega_i \mid x) = 1$ we define a normalization factor $f(x) = \left[\sum_{j=1}^{n} \left(\frac{\hat{p}(\omega_j)}{\hat{p}_t(\omega_j)} \right) \hat{p}_t(\omega_j \mid x) \right]^{-1}$

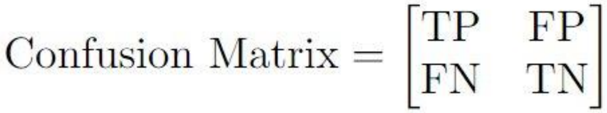
$$\hat{p}(\omega_i \mid x) = \frac{\left(\frac{\hat{p}(\omega_i)}{\hat{p}_t(\omega_i)} \right) \hat{p}_t(\omega_i \mid x)}{\sum_{j=1}^{n} \left(\frac{\hat{p}(\omega_j)}{\hat{p}_t(\omega_j)} \right) \hat{p}_t(\omega_j \mid x)} \qquad (2.4)$$

Unknown a Priori: Confusion Matrix Method

Equation for estimating the a priori $\hat{p}(\omega_j)$

$$\hat{p}(\delta_i) = \sum_{j=1}^{n} \hat{p}_t(\delta_i \mid \omega_j)\hat{p}(\omega_j)$$

Confusion matrix overview



$$\text{Confusion Matrix} = \begin{bmatrix} \text{TP} & \text{FP} \\ \text{FN} & \text{TN} \end{bmatrix}$$

Set up the System

Left Side×Priors=Right Side

$\hat{p}_t(\delta_i \mid \omega_j)$

$$Left\ Side = \left[\begin{matrix} \dfrac{TP_{train}}{TP_{train} + FP_{train}} & \dfrac{FP_{train}}{TP_{train} + FP_{train}} \\ \dfrac{FN_{train}}{FN_{train} + TN_{train}} & \dfrac{TN_{train}}{FN_{train} + TN_{train}} \end{matrix}\right]$$

$\hat{p}(\delta_i)$

$$Right\ Side = \left[\begin{matrix} \dfrac{TP_{test} + FN_{test}}{TP_{test} + FN_{test} + FP_{test} + TN_{test}} \\ \dfrac{FP_{test} + TN_{test}}{TP_{test} + FN_{test} + FP_{test} + TN_{test}} \end{matrix}\right]$$

Provides conditional probabilities for the training confusion matrix

Provides observed classification frequencies for the test confusion matrix

Priors=solve(Left Side, Right Side)

Unknown a Priori: Expectation - Maximization Algorithm

We have some realizations of X: $X_1^N = (x_1, x_2, \dots, x_N)$

Define the Likelihood:

$$L(x_1, x_2, \ldots, x_N) = \prod_{k=1}^{N} p(x_k) = \prod_{k=1}^{N} \sum_{i=1}^{n} p(x_k, \omega_i) = \prod_{k=1}^{N} \sum_{i=1}^{n} p(x_k \mid \omega_i) p(\omega_i)$$

Initialization:

Posterior probability of the model

$$\hat{p}_t(\omega_i \mid x_k) = g_i(x_k)$$

Training prior probability

$$\hat{p}_t(\omega_i) = \frac{N_t^i}{N_t}$$

New a priori

$$\hat{p}^{(0)}(\omega_i) = \hat{p}_t(\omega_i)$$

Expectation - Maximization steps

Expectation Step:

$$\hat{p}^{(s)}(\omega_i \mid x_k) = \frac{\left(\frac{\hat{p}^{(s)}(\omega_i)}{\hat{p}_t(\omega_i)}\right)\hat{p}_t(\omega_i \mid x_k)}{\sum_{j=1}^{n}\left(\frac{\hat{p}^{(s)}(\omega_j)}{\hat{p}_t(\omega_j)}\right)\hat{p}_t(\omega_j \mid x_k)}$$

Re-estimates the posterior probabilities using the current estimates of the priors

Maximization Step:

$$\hat{p}^{(s+1)}(\omega_i) = \frac{1}{N}\sum_{k=1}^{N}\hat{p}^{(s)}(\omega_i \mid x_k)$$

Updates the prior probabilities by averaging the posterior probabilities across all observations

Repeat this steps until convergence

Likelihood Ratio Test

Likelihood based on the **original a priori** probabilities:

$$
\begin{aligned}
L(x_1, x_2, \ldots, x_N) &= \prod_{k=1}^{N} \hat{p}_t(x_k) \\
&= \prod_{k=1}^{N} \left[\frac{\hat{p}(x_k \mid \omega_i)\hat{p}_t(\omega_i)}{\hat{p}_t(\omega_i \mid x_k)} \right]
\end{aligned}
\tag{3.1}
$$

Likelihood based on the **new a priori** probabilities:

$$
\begin{aligned}
L(x_1, x_2, \ldots, x_N) &= \prod_{k=1}^{N} \hat{p}(x_k) \\
&= \prod_{k=1}^{N} \left[\frac{\hat{p}(x_k \mid \omega_i)\hat{p}(\omega_i)}{\hat{p}(\omega_i \mid x_k)} \right]
\end{aligned}
\tag{3.2}
$$

Likelihood ratio:

$$
\begin{aligned}
\frac{L(x_1, x_2, \ldots, x_N)}{L_t(x_1, x_2, \ldots, x_N)} &= \frac{\prod_{k=1}^{N} \left[\frac{\hat{p}(x_k|\omega_i)\hat{p}(\omega_i)}{\hat{p}(\omega_i|x_k)} \right]}{\prod_{k=1}^{N} \left[\frac{\hat{p}(x_k|\omega_i)\hat{p}_t(\omega_i)}{\hat{p}_t(\omega_i|x_k)} \right]} \\
&= \frac{\prod_{k=1}^{N} \frac{\hat{p}(\omega_i)}{\hat{p}(\omega_i|x_k)}}{\prod_{k=1}^{N} \frac{\hat{p}_t(\omega_i)}{\hat{p}_t(\omega_i|x_k)}}
\end{aligned}
\tag{3.3}
$$

Log-likelihood ratio:

$$
\begin{aligned}
\log\left(\frac{L(x_1, x_2, \ldots, x_N)}{L_t(x_1, x_2, \ldots, x_N)}\right) &= \sum_{k=1}^{N} \log\left(\hat{p}_t(\omega_i \mid x_k)\right) - \sum_{k=1}^{N} \log\left(\hat{p}(\omega_i \mid x_k)\right) \\
&+ N \log\left(\hat{p}(\omega_i)\right) - N \log\left(\hat{p}_t(\omega_i)\right).
\end{aligned}
\tag{3.4}
$$

Statistical Inference

Null Hypothesis H0:
The a priori probabilities have not changed. There is no significant difference between the original and updated a priori probabilities.

Alternative Hypothesis H1:
The a priori probabilities have changed. There is a significant difference between the original and updated a priori probabilities.

Compute the test statistic $2 * \log(\frac{L}{L_t})$ is distributed as a chi-squared distribution with **n-1 degrees of freedom**

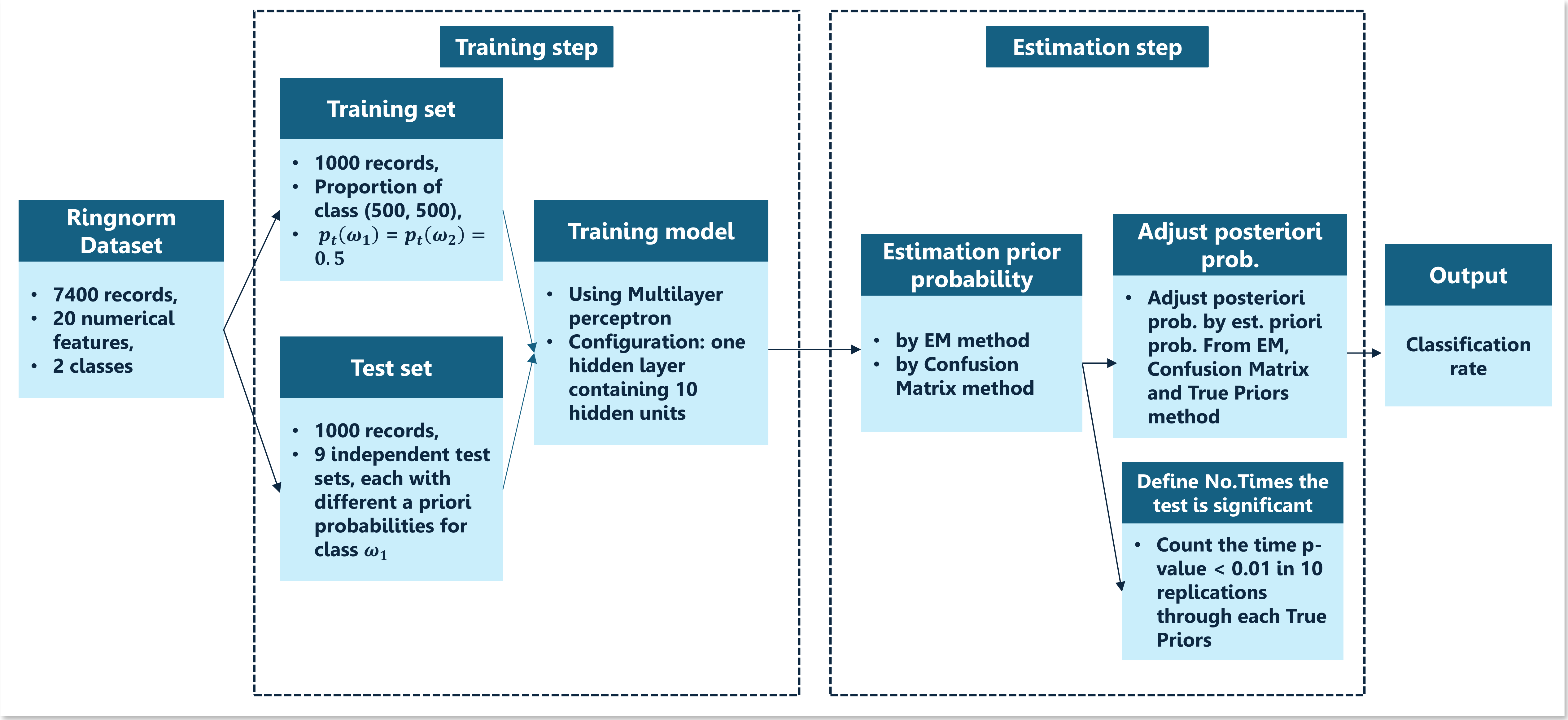
Decision rule:

•If **p-value<significance level:** Reject H0, the evidence suggests that the a priori probabilities have significantly changed.

•If **p-value≥significance level:** Fail to reject H0, there is not enough evidence to suggest that the a priori probabilities have changed

Simulation on Artificial Data: Experimental Design

The experiment was replied 10 times to ensure the reliability



Ringnorm Dataset

- 7400 records,
- 20 numerical features,
- 2 classes

Training step

Training set

- 1000 records,
- Proportion of class (500, 500),
- $p_t(\omega_1) = p_t(\omega_2) = 0.5$

Test set

- 1000 records,
- 9 independent test sets, each with different a priori probabilities for class ω_1

Training model

- Using Multilayer perceptron
- Configuration: one hidden layer containing 10 hidden units

Estimation step

Estimation prior probability

- by EM method
- by Confusion Matrix method

Adjust posteriori prob.

- Adjust posteriori prob. by est. priori prob. From EM, Confusion Matrix and True Priors method

Define No.Times the test is significant

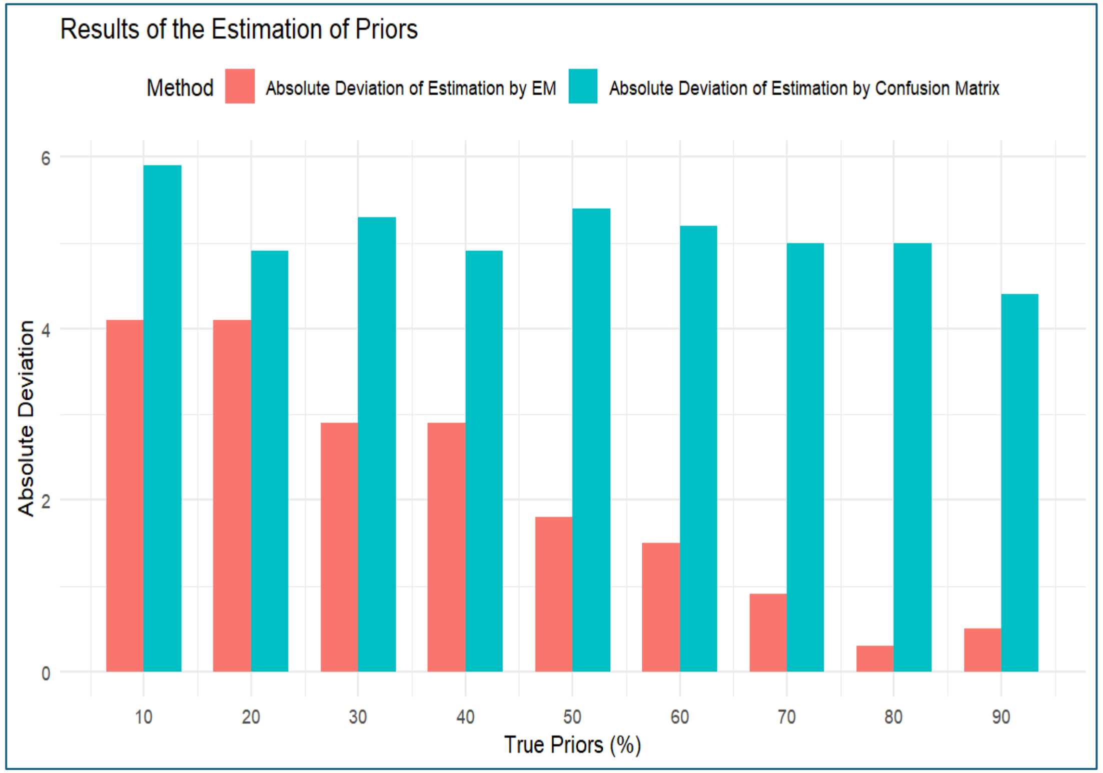
- Count the time p-value < 0.01 in 10 replications through each True Priors

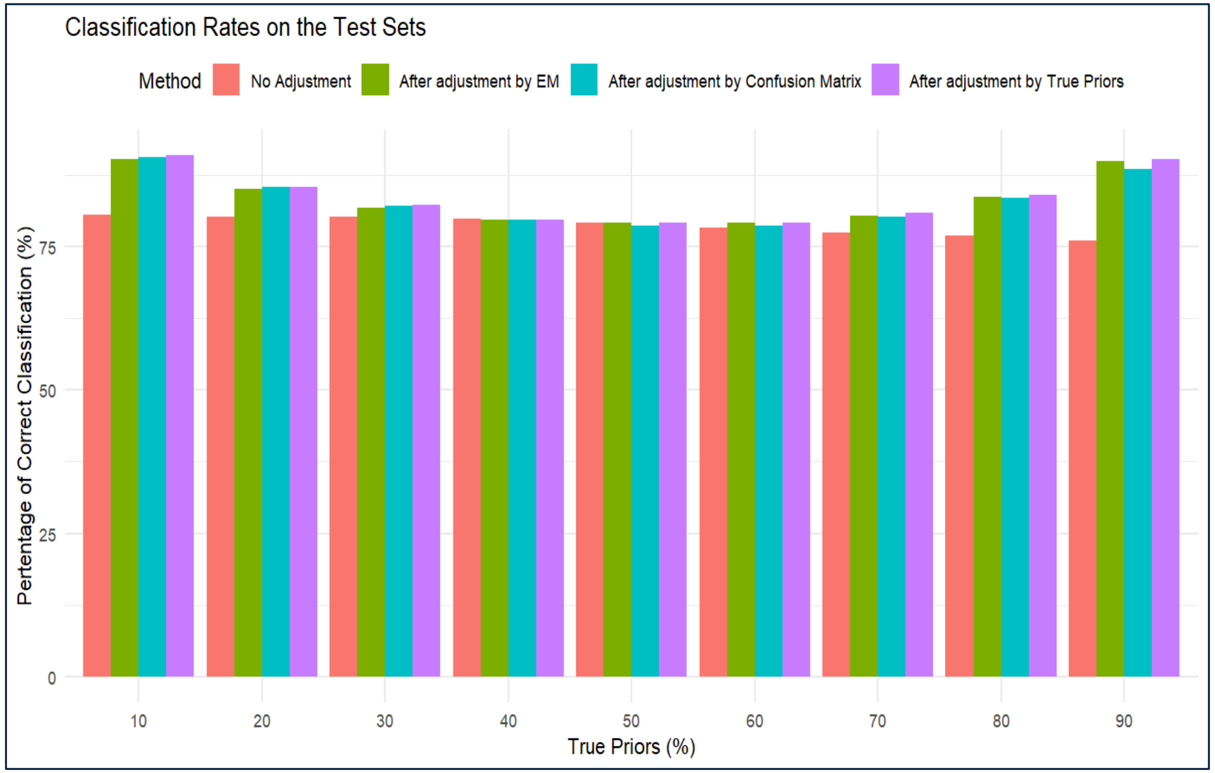
Output

Classification rate

Results of the Estimation Priors on the Artificial Dataset

Results of the Estimation of Priors on the Test Sets



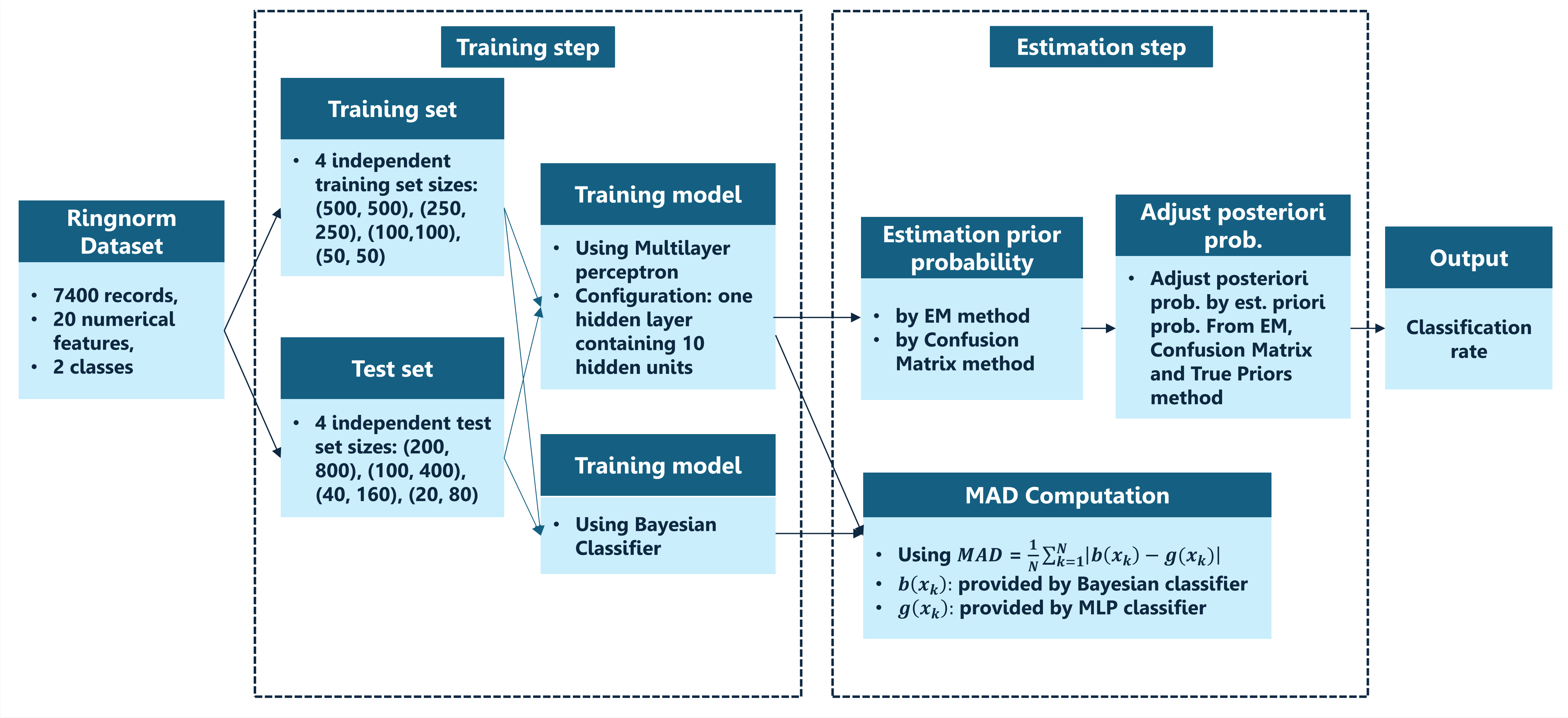


Key Assessments

- Absolute Deviation = | $Estimated\ Priori - True\ Priors$ |
- The classification rates after adjustment by EM algorithm and the Confusion Matrix method were very close
- The Classification Rates after adjustment by EM, Confusion Matrix, True Priors always gave better results than No Adjustment. The EM algorithm provided results closer to the True Priors than Confusion Matrix except in the case True Priors = 10%, 20%, 30%

Robustness Evaluation: Experimental Design

The experiment was replied 10 times to ensure the reliability



Ringnorm Dataset

- 7400 records,
- 20 numerical features,
- 2 classes

Training step

Training set

- 4 independent training set sizes: (500, 500), (250, 250), (100,100), (50, 50)

Test set

- 4 independent test set sizes: (200, 800), (100, 400), (40, 160), (20, 80)

Training model

- Using Multilayer perceptron
- Configuration: one hidden layer containing 10 hidden units

Training model

- Using Bayesian Classifier

Estimation step

Estimation prior probability

- by EM method
- by Confusion Matrix method

Adjust posteriori prob.

- Adjust posteriori prob. by est. priori prob. From EM, Confusion Matrix and True Priors method

Output

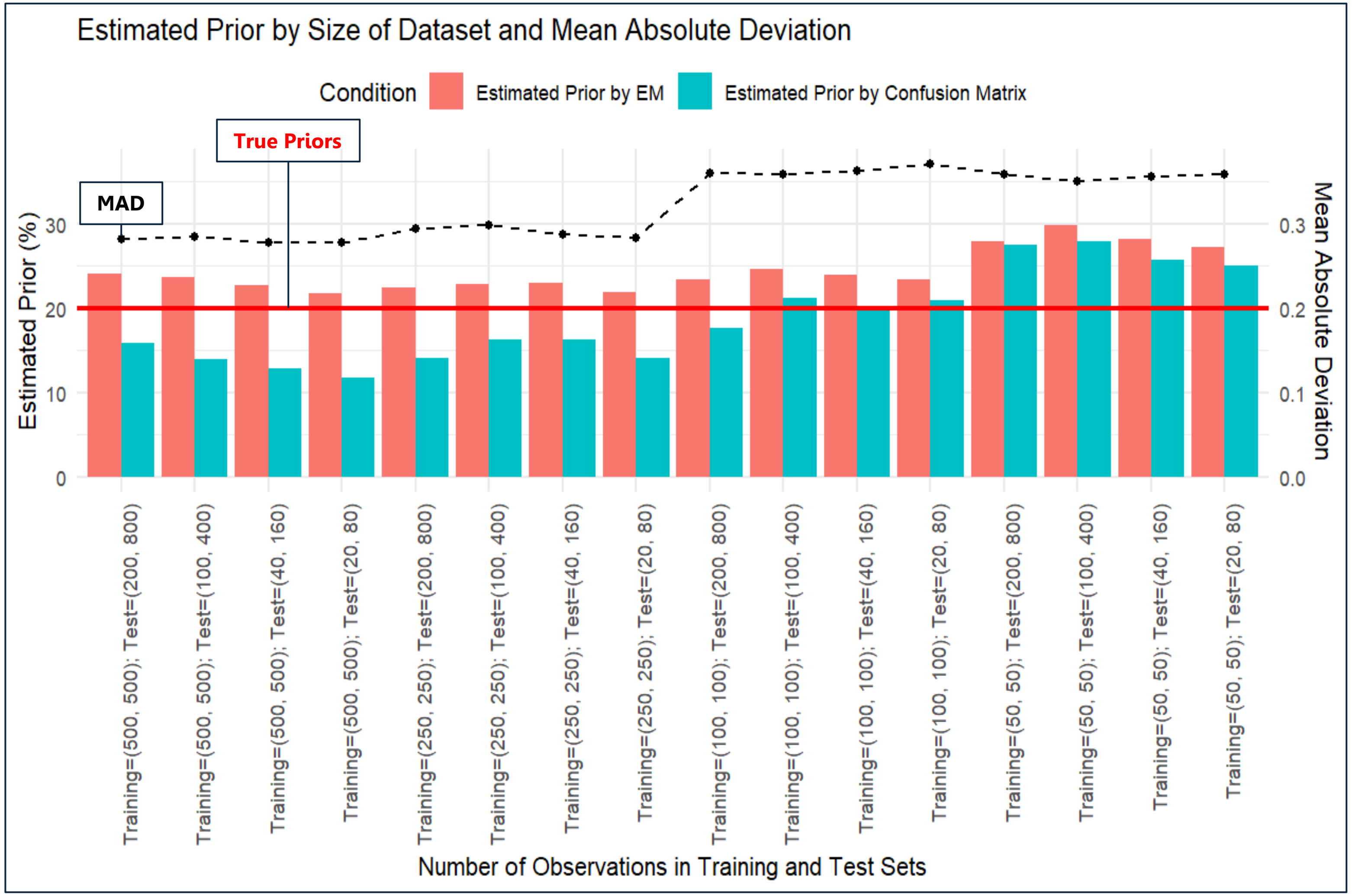
Classification rate

MAD Computation

- Using $MAD = \frac{1}{N}\sum_{k=1}^{N}|b(x_k) - g(x_k)|$
- $b(x_k)$: provided by Bayesian classifier
- $g(x_k)$: provided by MLP classifier

Robustness Evaluation: Estimated Priors

Average Results for Estimation of the Priors



Comments

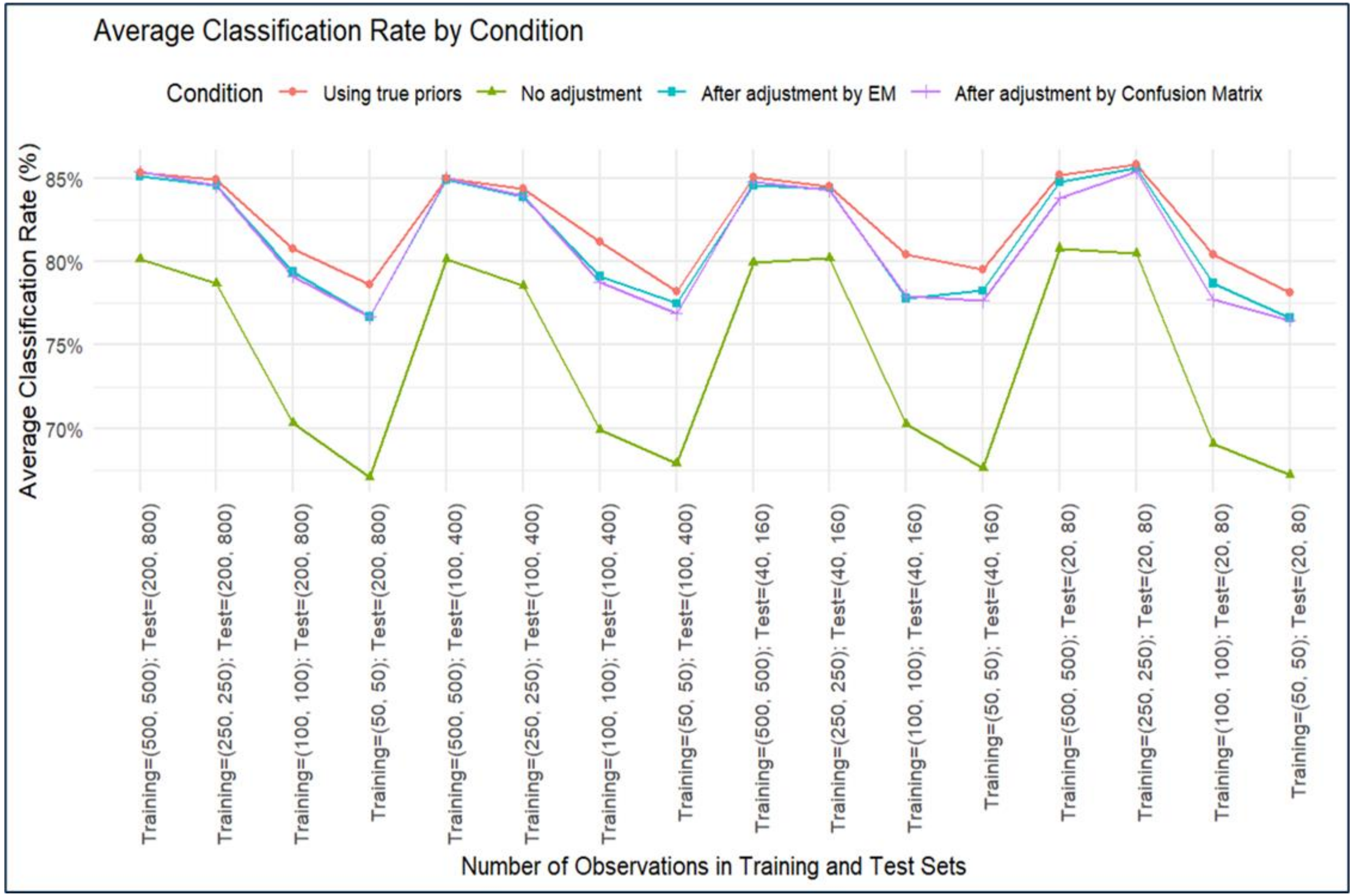
Key Assessments:

1. At MAD, decreasing the training set size resulted in an increasing of MAD
2. In large training set size (500, 250), MAD were low. In contrast, small training set size (100, 50), MAD were higher than
3. At the large training set (500, 250), the estimated prior by EM gave closer results than the Confusion Matrix. However, in small training set (100, 50), the Confusion Matrix gave closer than EM
4. In both methods, with the same training set, the estimated priors showed a slight down trend as the test set size decreased
5. At the smallest training set (50), the estimated priors in both methods were higher than others.

Conclusion: decreasing size of test set seems to have few effects on the results

Robustness Evaluation: classification rate before and after adjustment

Classification Rates obtained on the different Training and Test Sets



Comments

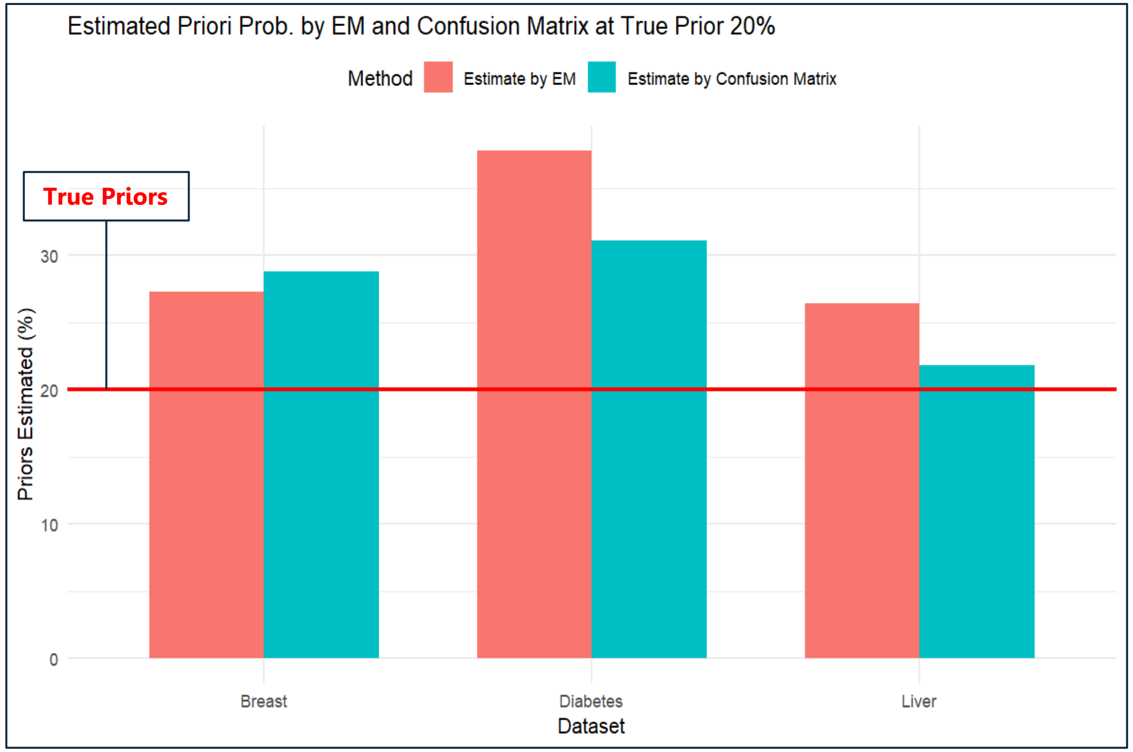
Key Assessments:

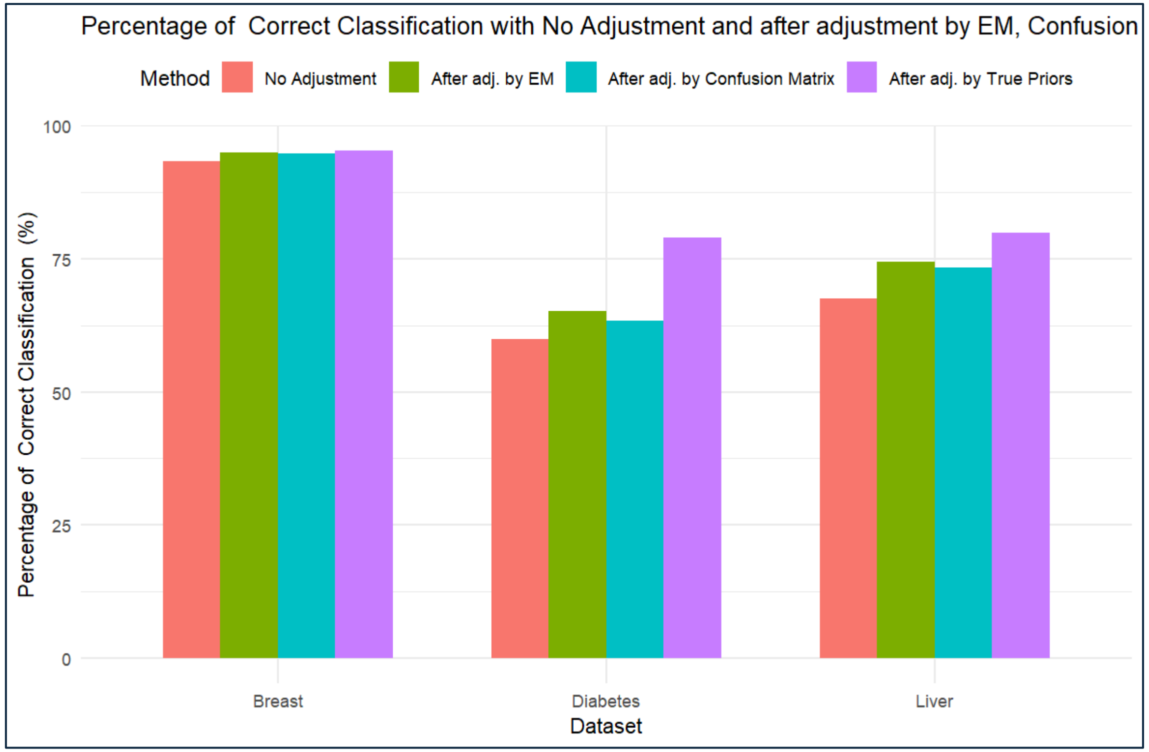
1. There are the degradation in the classifier performances due to the decrease in the size of the training sets
2. The classification rates after adjustment in overall were always higher than no adjustment
3. The classification rates obtained after the adjustments by Confusion Matrix method were very close to those obtained with the EM method
4. The EM method always provided better results

Conclusion: Adjust the outputs of the classifier, the classification rate will be increased significantly

Tests on Real Data: a Priori Estimation And Output Readjustment Method

Classification Results on Three Real Data Sets





Key assessment

- The Confusion Matrix prior estimates were better than the EM (except for the Breast dataset)
- At Classification rates, the rate after adjustment by the EM were always higher than the Confusion Matrix

Conclusion: Adjust the classifier outputs based on the new prior probabilities improved classification rates and accuracy, approaching the results obtained when using True Priors for output adjustment

Conclusion

Findings

1 Priori Estimation

- The EM method is able to provide good estimation of the new a priori probabilities

2 Classification rate

- The classifier with adjusted output always give better results than the original one
- The classification performances after adjustment by EM give results close to the results obtained by using the true priors
- The adjustments made by EM and Confusion Matrix method give same effect on the accuracy improvement

3 Robustness evaluation

- Decreasing size of test set seems to have few effects on the results of estimated priori
- The estimates from EM method gives more robust than Confusion Matrix

Limits

1 Training model

- The experimental test applied on the simple MLP model. It causes the outputs may not be optimized

2 Dataset

- In the real dataset, we only apply in Medical problem. We could expand more to geographical, image processing problems

3 Output

- The output of group differ slightly from the numerical results in the paper. But in general, we highlighted key finding that the authors studied

Thanks for your listening