READ ME

Bayesian Logistic Regression

There are 2 main files:

1. **JAGS\_final\_5features**: includes the model with 5 features
2. **JAGS\_final\_4features**: includes the model without area\_mean

Each file contains the following programs which are required to run the model:

|  |  |
| --- | --- |
| **Name** | **Description** |
| * JAGS\_main\_program.R * Jags\_Ydich\_weightOnly-SummaryInfo.csv * DBDA2E-utilities.R | Three R programs which are required for implementing the Bayesian logistic model. To run the complete model, it is only required to run **JAGS\_main\_program.R**. The other two files contain the functions needed to perform the MCMC algorithm. |
| * prepare\_train\_test\_for\_JAGS.py | This python program takes as input the data (format: pickle), encodes the target “diagnosis” from M-F to 1-0, and splits the data into train (70%) and test (30%). As an output, the program creates 2 csv files: **df\_train.csv** and **df\_test.csv**. |
| * df\_train.csv * df\_test.csv | Training and test data. The training data is directly used by JAGS to fit the model. The test data is used, afterwards, to test the performance of the model. |

Once the model is implemented, the following output is obtained:

|  |  |
| --- | --- |
| **Name** | **Description** |
| * TEMPmodel.txt | This file contains a description of the Bayesian logistic model. It is created and used by JAGS to perform the MCMC algorithm. |
| * weightOnly-SummaryInfo.csv | This file is generated by JAGS. It contains information about the mean, model, and posterior distribution of each parameter (beta). |
| * visualizations | File containing all visualizations created by the software. |
| * test\_performance.ipynb | This Jupyter Notebook file takes as input the df\_test.csv file and measures accuracy, identifies false negatives and false positives, and creates a confusion matrix. The model used the mean value of the parameters (which can be found in the file **weightOnly-SummaryInfo.csv**). |