

Applied machine learning

mini project

December 9, 2016

**Abstract**:

We are a three-member team implementing 3 algorithms in Python using open source datasets from UCI & from Human Activity Recognition. We will evaluate our implementation of the algorithm and also compare those with the output of Weka as well. We draw conclusions from these comparison, provide pitfalls of our implementation, performance of the algorithm, and ways to improve our implementation and also the algorithm’s output.

# Group Members

Balaji Rajaram

Dipankar Biswas

Malabika Biswas

**Table of Contents**

Dataset 1

### **Dataset Name**: Human Activity Recognition

### **Dataset Description**:

Human Activity Recognition (HAR) - has emerged as a key research area in the last years and is gaining increasing attention by the pervasive computing research community, especially for the development of context-aware systems. There are many potential applications for HAR, like: elderly monitoring, life log systems for monitoring energy expenditure and for supporting weight-loss programs, and digital assistants for weight lifting exercises. This dataset has 5 classes (sitting-down, standing-up, standing, walking, and sitting) collected on 8 hours of activities of 4 healthy subjects [1].

### **Naïve Bayes Algorithm**:

In this section, we discuss the analysis performed on the dataset using Naïve Bayes algorithm, what are the pre-processing steps that we have carried out, analysis on the dataset, output of the algorithm from our python implementation, Weka output, comparison of both the results, learnings, pitfalls of our implementation and potential ways to improvise the performance.

#### Analysis & pre-processing:

This dataset has 159 columns including the target class. The first analysis we performed was to find out columns that can be dropped. We have identified that the below columns may not be helpful in the classification problem and hence we dropped these columns. Figure 1 shows the sample values for these columns.

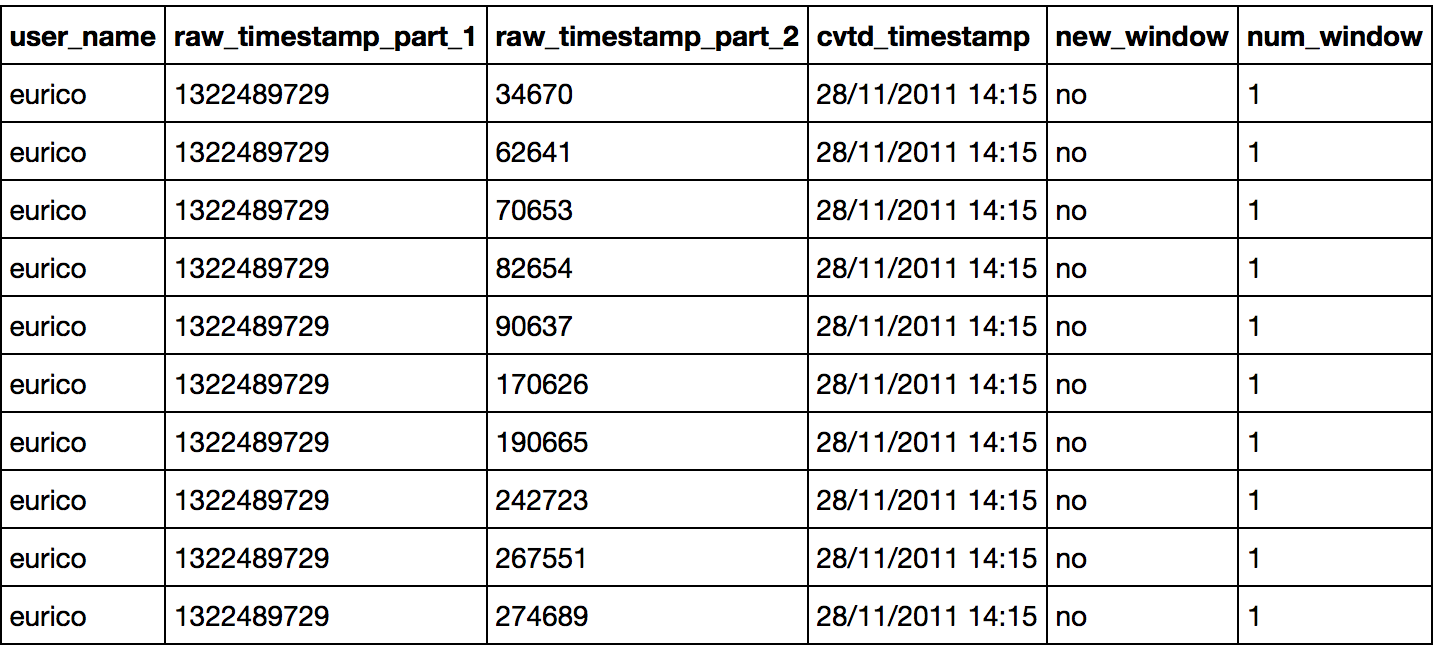


Figure 1

The next analysis we carried out to minimize the features was to find out the columns that has less than 5% value of the total record count in the dataset. Table-x shows the total record count and the threshold value to drop columns.

Table 1

|  |  |
| --- | --- |
| Total Record Count | 39242 |
| Threshold Record Count | 1962.1 |

We identified 100 columns that were below the threshold and we dropped those columns. With this initial pre-processing, we came up with 53 columns including the target class. We then did feature correlation analysis to identify how the features are correlated. Figure 2 shows the correlation analysis chart.

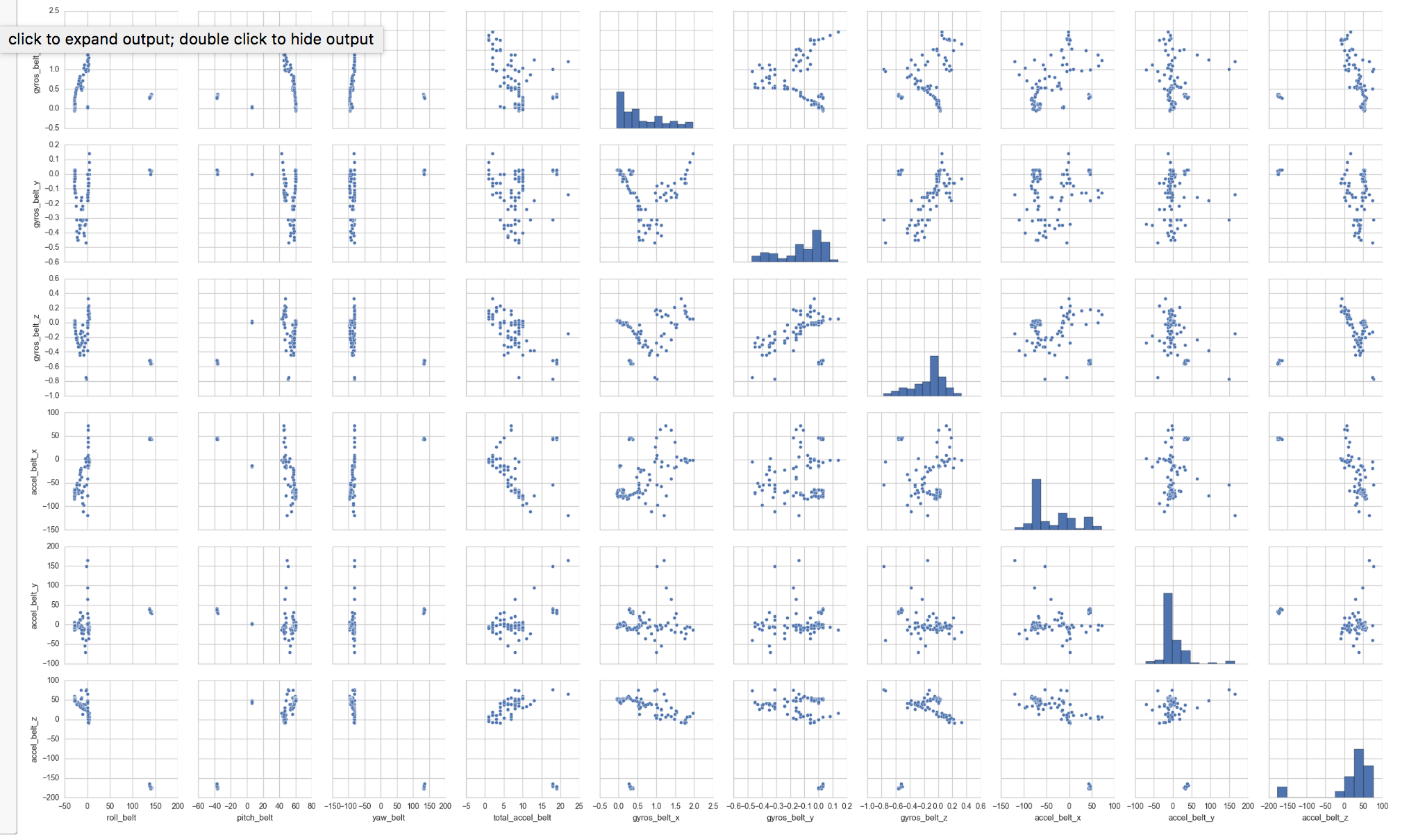


Figure 2

We could see that there is correlation between variables, which is not a good sign to perform Naïve Bayes as the algorithm works based on naive assumptions that all the variables are uncorrelated to each other, which is not true in this case.

#### Python Implementation Results:

We have implemented the Naïve Bayes algorithm in Python and we used the dataset with 53 columns. Figure 3 shows the output of our python implementation and Figure 4 shows the confusion matrix. We divided the dataset into 70-30 ratio for training data and test data respectively.

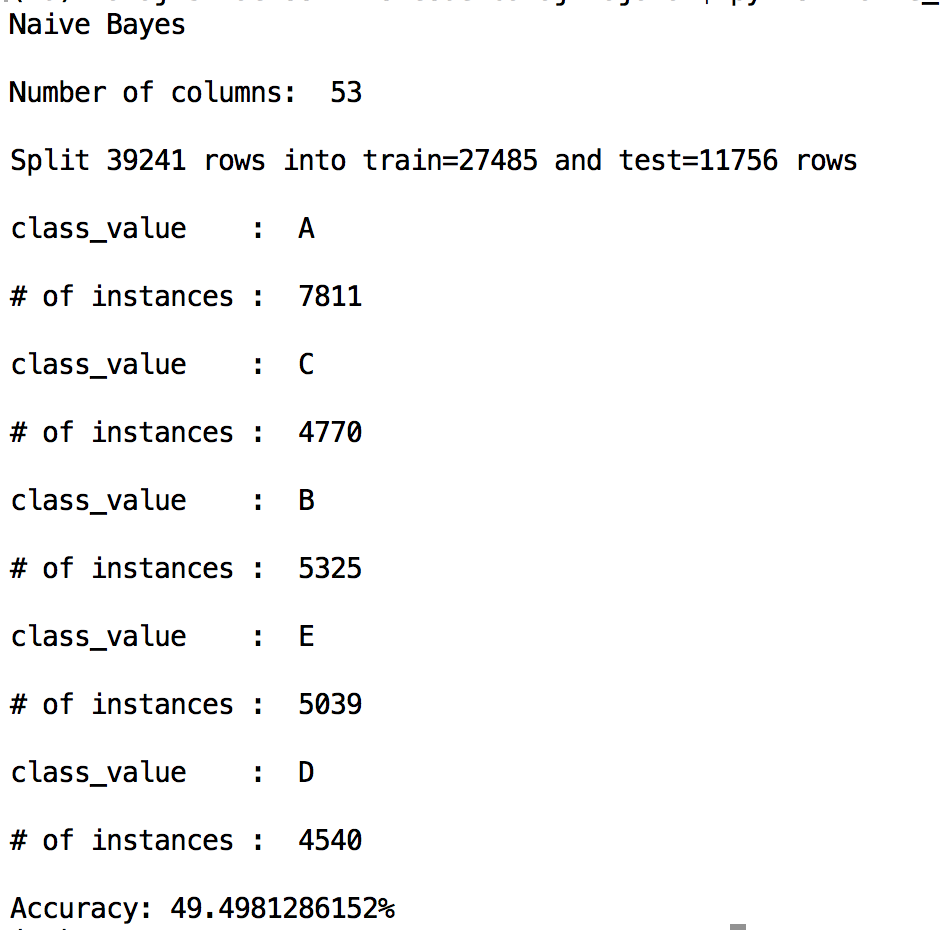


Figure 3

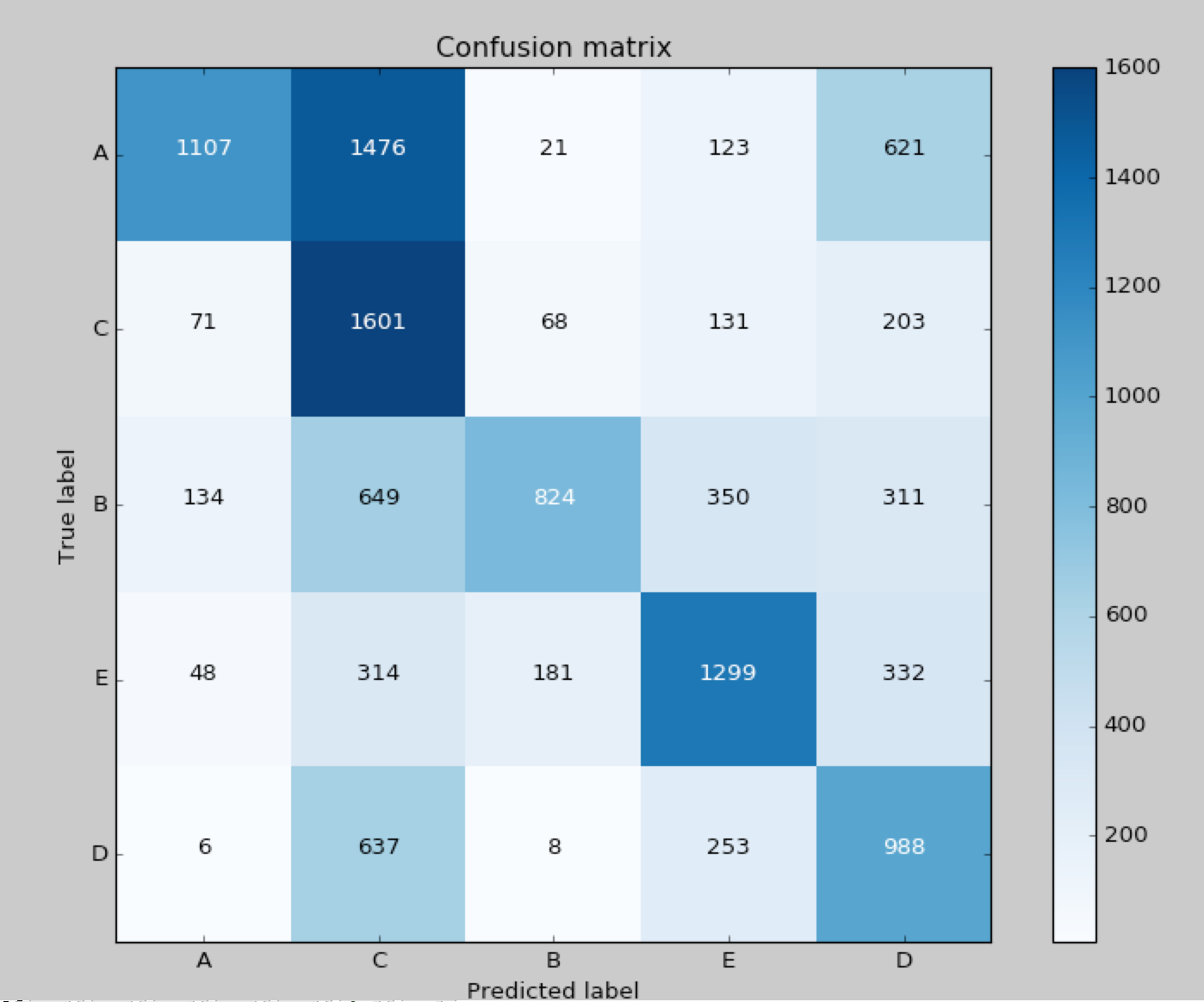


Figure 4

#### Weka Result:

#### Learning:

We were able to correlate the theoretical understanding of the algorithm with the practical implementation especially with the assumption of Naïve Bayes that the features are not correlated to each other.

#### Pitfalls & ways to improvise the classification:

It was clear that the correlation between features made the algorithm to perform poor. We need to focus more on the feature engineering part to identify the variables that have more correlation and measures to remove the correlation by retaining the features that has no correlation and by creating new variables so that we can remove the correlation between variables. Feature engineering plays a vital role in improving the algorithm’s performance.

Dataset 2

**Dataset Name**: Student Alcohol Consumption

**Dataset Description**: The dataset provides the result of correlation between alcohol usage and the social, gender and study time attributes for each student. This is a multivariate classification dataset [2]. The dataset has 2 target classes: one is workday alcohol consumption and the other is for weekend alcohol consumption.

**Naïve Bayes Algorithm**:

In this section, we discuss the analysis performed on the dataset using Naïve Bayes algorithm, what are the pre-processing steps that we have carried out, analysis on the dataset, output of the algorithm from our python implementation, Weka output, comparison of both the results, learnings, pitfalls of our implementation and potential ways to improvise the performance.

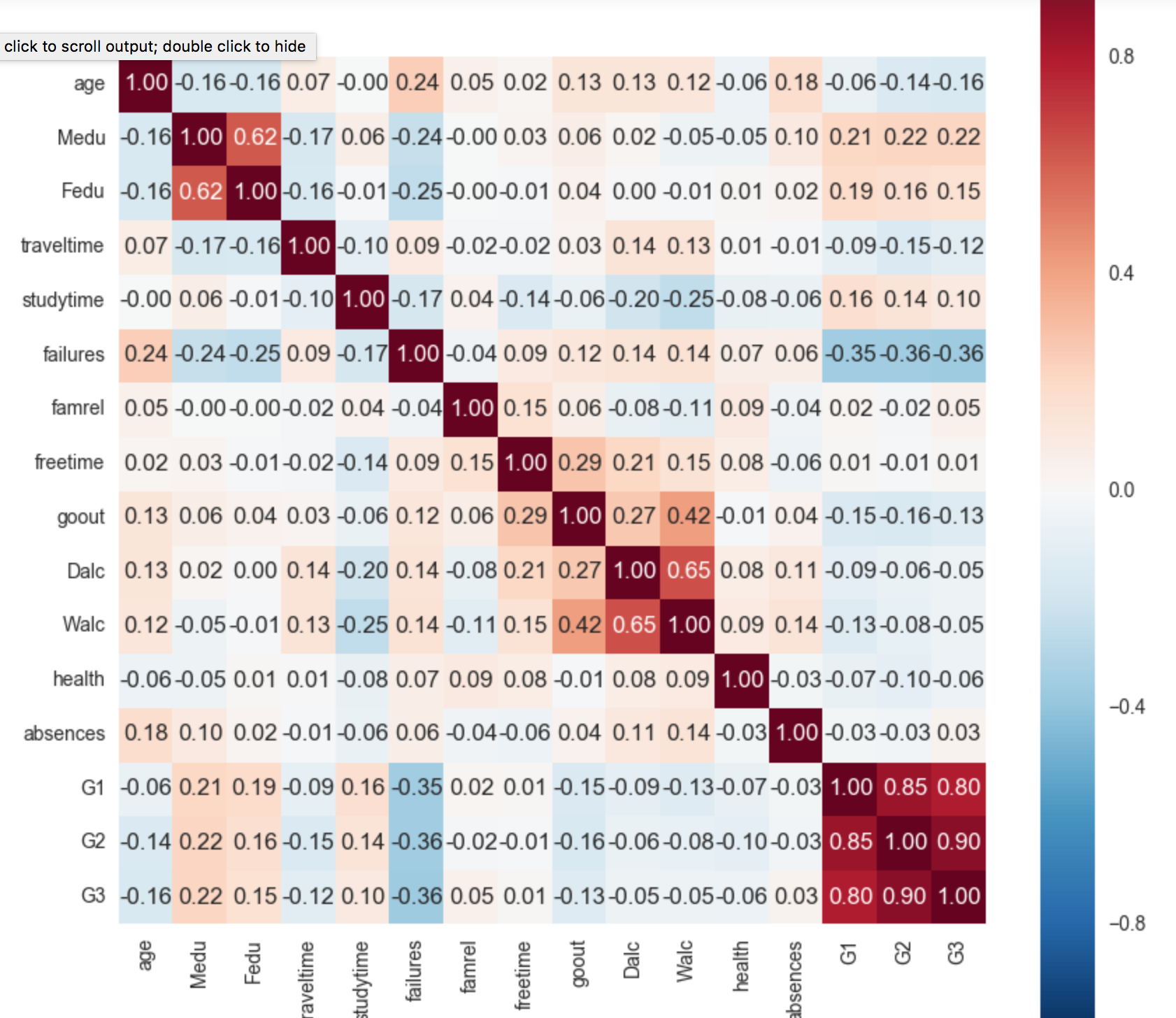
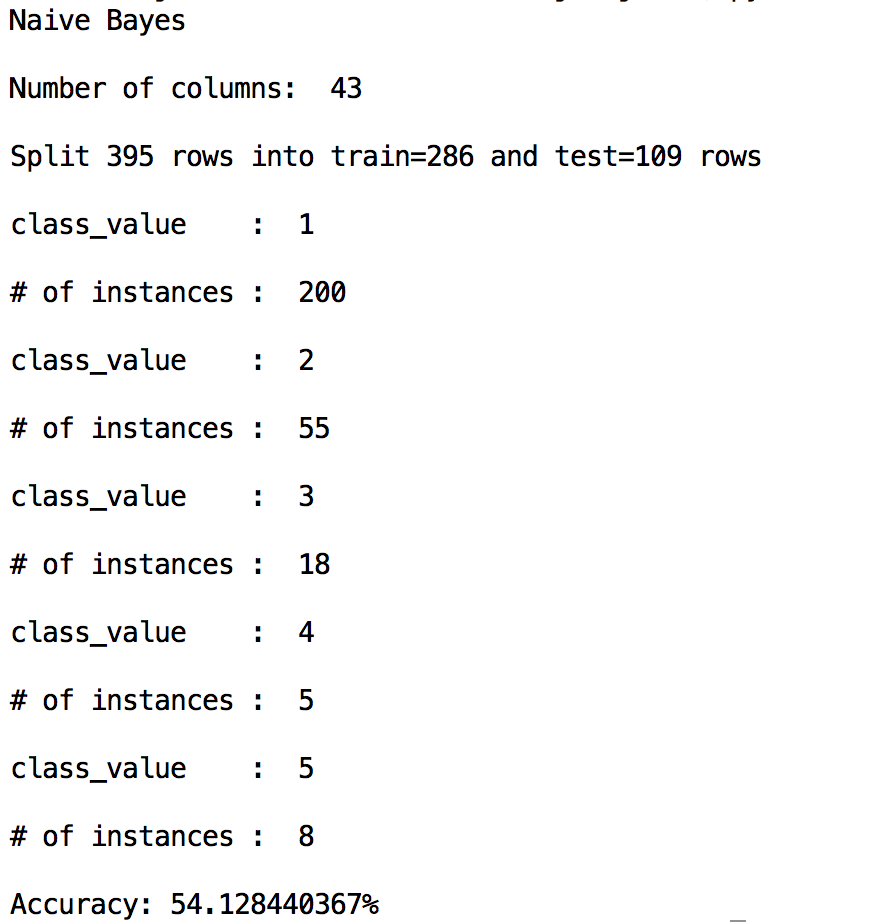
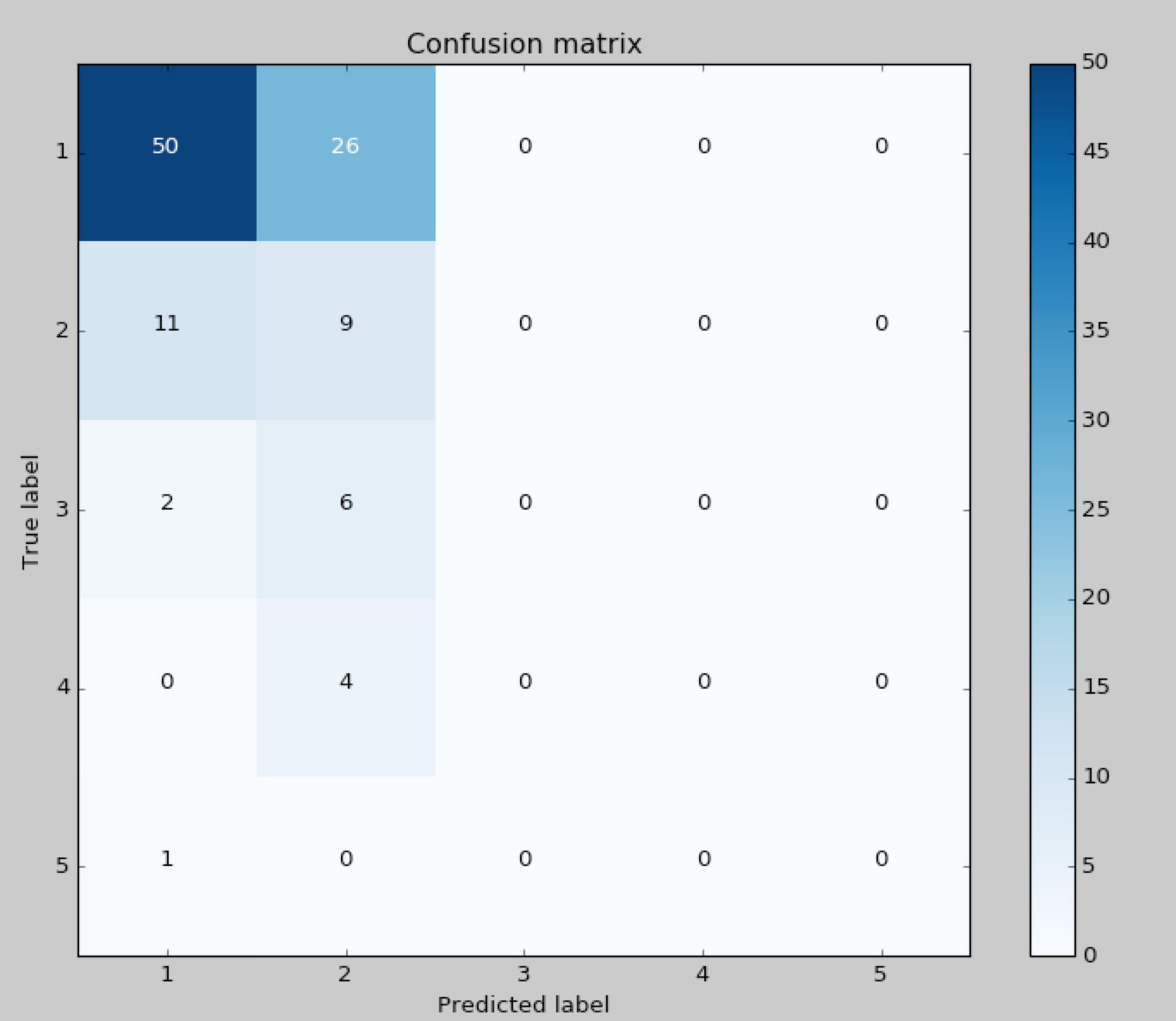


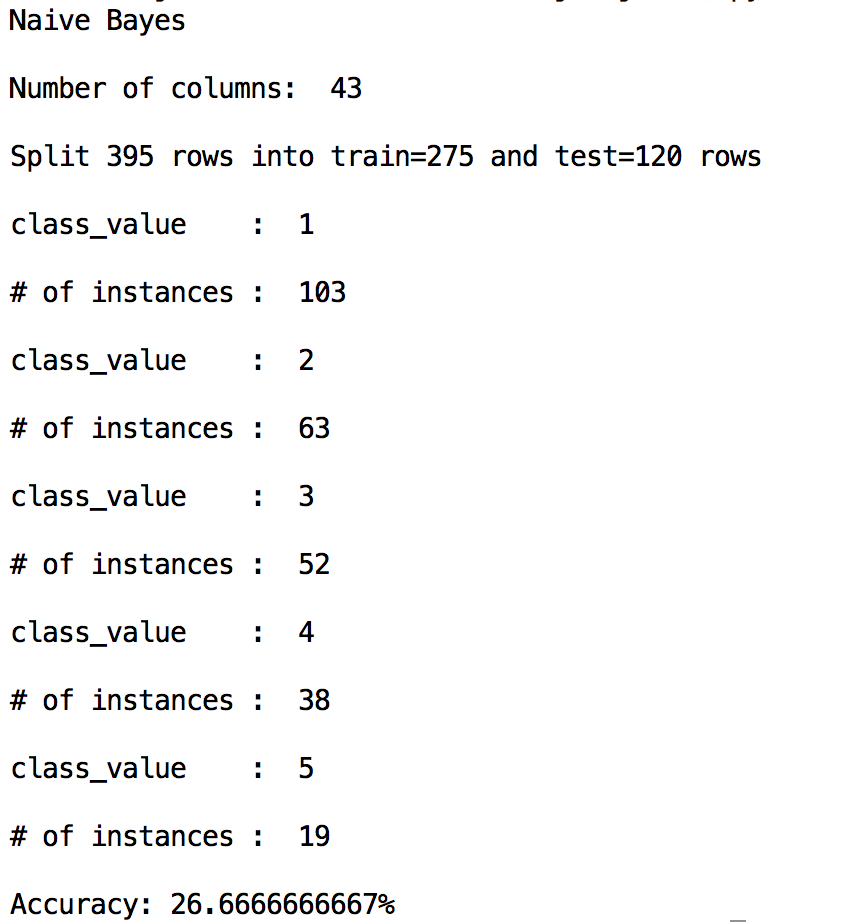
Figure 5

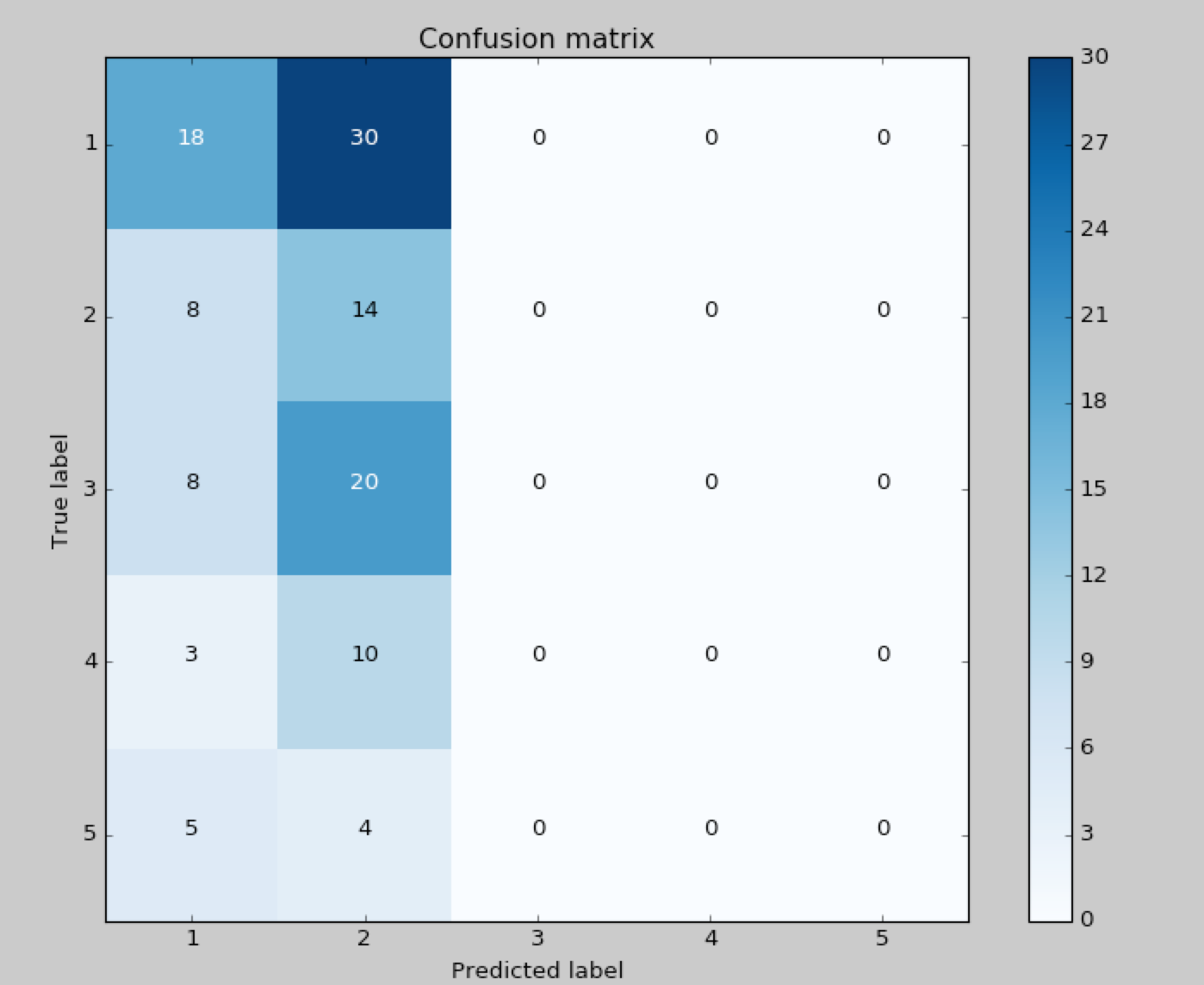
#### Work Day Alcohol Consumption Results:



****

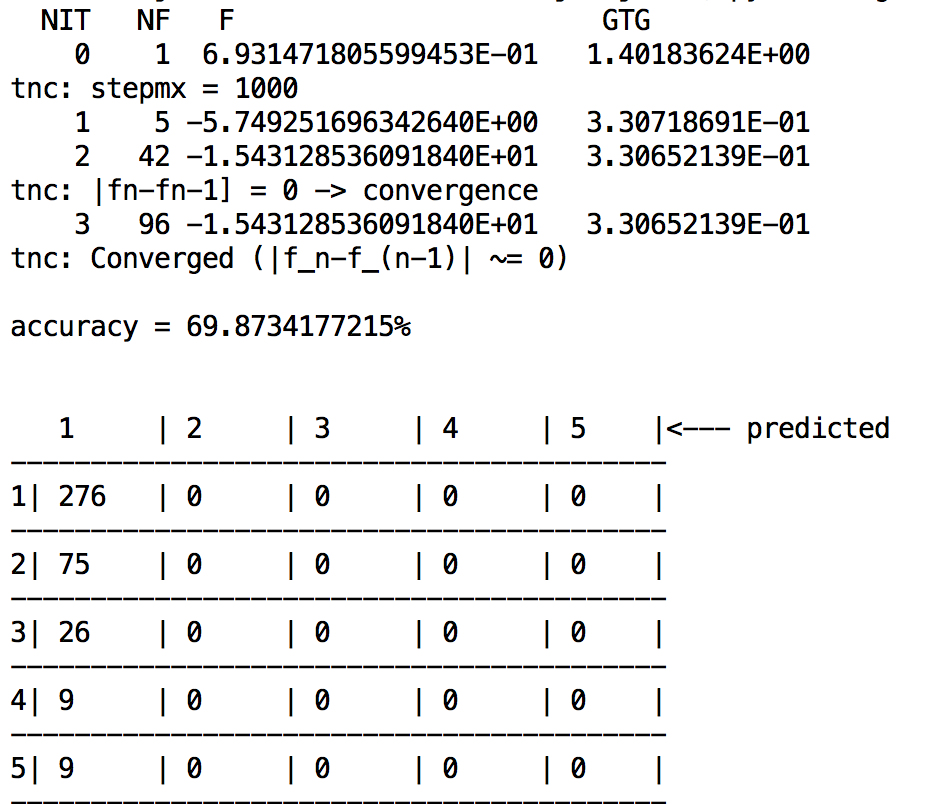
#### Weekend Alcohol Consumption Results:



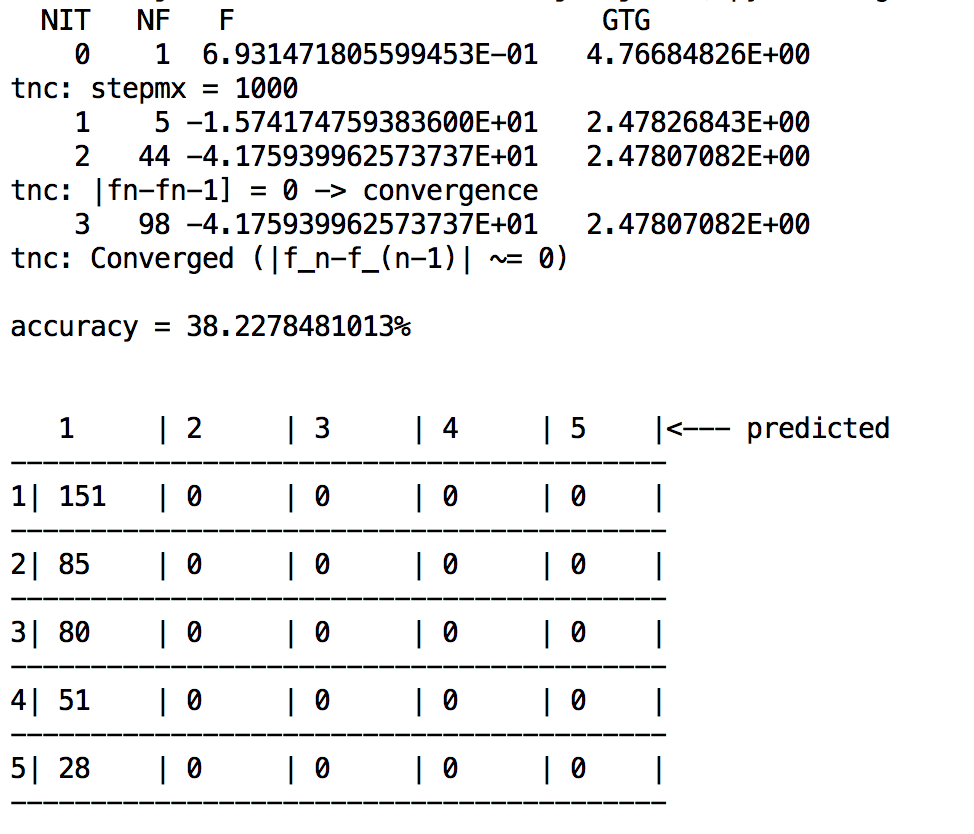


#### **Logistic Regression**:

#### Work Day Alcohol Consumption Results:



#### Weekend Alcohol Consumption Results:



References:

1. Ugulino, W.; Cardador, D.; Vega, K.; Velloso, E.; Milidiu, R.; Fuks, H. Wearable Computing: Accelerometers' Data Classification of Body Postures and Movements. Proceedings of 21st Brazilian Symposium on Artificial Intelligence. Advances in Artificial Intelligence - SBIA 2012. In: Lecture Notes in Computer Science. , pp. 52-61. Curitiba, PR: Springer Berlin / Heidelberg, 2012. ISBN 978-3-642-34458-9. DOI: 10.1007/978-3-642-34459-6\_6.
2. P. Cortez and A. Silva. Using Data Mining to Predict Secondary School Student Performance. In A. Brito and J. Teixeira Eds., Proceedings of 5th FUture BUsiness TEChnology Conference (FUBUTEC 2008) pp. 5-12, Porto, Portugal, April, 2008, EUROSIS, ISBN 978-9077381-39-7.