N E W S
OF THE NATIONAL ACADEMY OF SCIENCES OF THE REPUBLIC OF KAZAKHSTAN
SERIES OF GEOLOGY AND TECHNICAL SCIENCES
ISSN 2224-5278
Volume 6, Number 426 (2017), 231 – 236

UDC 004.855 - 004.891.3

Sh. Shamiluulu[1], B. Y. Amirgaliyev[2], L. Cherikbayeva[3]

[1]Department of Computer Science, Suleyman Demirel University, Kaskelen, Kazakhstan,
[2]International University of Information Technologies, Almaty, Kazakhstan,
[3]Al-Farabi Kazakh National University, Almaty, Kazakhstan.
E-mail: shahriar.shamiluulu@sdu.edu.kz,amirgaliyev@gmail.com,lailash01@gmail.com

CRITICAL ANALYSIS OF SCIKIT-LEARN ML FRAMEWORK AND WEKA ML TOOLBOX OVER DIABETES PATIENTS MEDICAL DATA

Abstract. In this research study two ml tools tested i.e., scikit-learn and Waikato Environment for Knowledge Analysis, over three supervised machine learning algorithms. The comparative analysis has been performed on tree supervised machine learning algorithms i.e., decision trees, logistic regression and multi-layer perceptron neural network on medical data for patients with two types of diabetes disorders. Diabetes-Mellitus refers to the metabolic disorder that happens from malfunction in insulin secretion and action. It is characterized by hyperglycemia.The diagnosis of diabetes is very important now days using various types of techniques and thus selection of framework is important.The dataset has been obtained from UCI machine learning repository for Pima Indian Diabetes patients. During the research the comparative analysis studies have been performed which revealed that less complex algorithms can be used for disease diagnosis and possess better performance when properly configured.

Keywords: diabetes mellitus, supervised learning, performance analysis, clinical decision support systems.

Introduction. There are many machine learning frameworks and tools available for public use. This research study tries to thoroughly analysis two tools and reveal which all trade-offs.In this study, the comparative performance analysis of three supervised machine learning algorithms is studied i.e, decision trees, logistic regression and artificial neural network and advantages of using Waikato Environment for Knowledge Analysis (WEKA) machine learning toolbox [3] is shown in Figure 1. Using WEKA toolbox is relatively easy which has many build-in options as compare to scikit-learn framework where in order to use algorithms users need to write code as shown in Figure 2.

The comparative analysis is going to be performed on medical dataset. Currently the computer aided diagnosis plays an important role in the medical field. It has been shown that the benefits of introducing machine learning into medical analysis are to increase the diagnostic accuracy, to reduce costs and to reduce human resources. The algorithms are tested over the Pima Indian diabetes dataset. Pima Indian Diabetes database had been examined with several different machine learning methods in the past [5-9]. Diabetes-Mellitus refers to the metabolic disorder that happens from malfunction in insulin secretion and action. It is characterized by hyperglycemia. There are two types of diabetes disorder but generally the symptomatic and lab results are same. The diagnosis of diabetes is very important now days using various types of techniques.

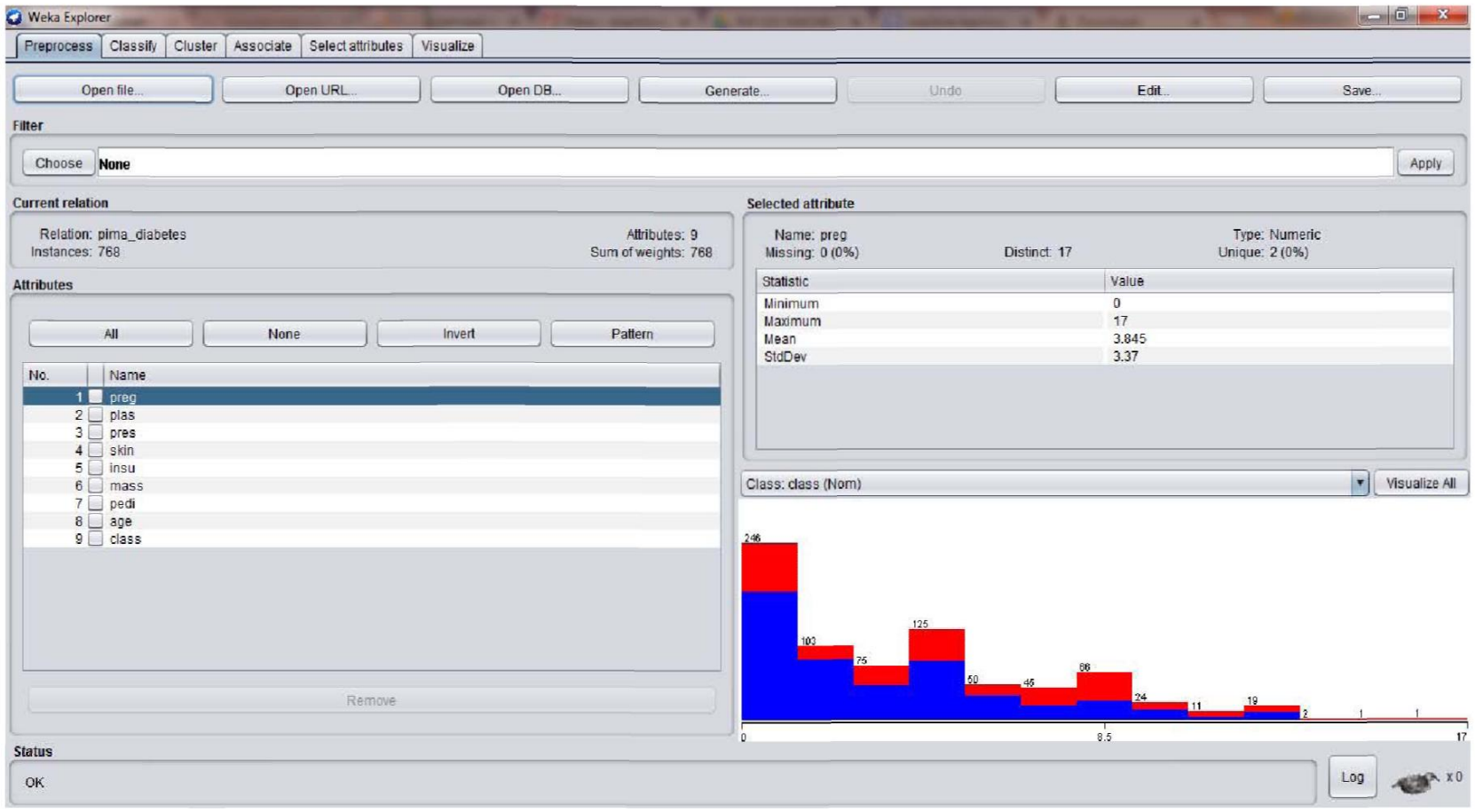

Figure 1 – Dataset visualization of WEKA toolbox

```
#Read pima dataset file and extract instances
fp = open('pima_indians_diabetes.txt', 'r')
cnt=1

data=[]
target=[]
for line in fp.xreadlines():
    instance=split(line,',')
    dp = []
    for idx in range(0,8,1):
        dp.append(float(instance[idx]))
    data.append(dp)

    target.append(float(instance[8]))
    #print cnt,'.',line
    cnt+=1

option=raw_input('Do you want to normalize-n or standardize-s data or none ? : ')
if(option=='n'):
    data_norm=preprocessing.normalize(data)
elif(option=='s'):
    data_norm=preprocessing.scale(data)
else:
    data_norm=data

fig = plt.figure()
ax = fig.add_subplot(111)
type1=ax.scatter([d[5] for d in data_norm], [d[7] for d in data_norm], color='red')

X_train, X_test, Y_train, Y_test = train_test_split(data_norm, target, test_size=0.3, random_state=0)
dt=DecisionTreeClassifier()
dt.fit(X_train,Y_train)

predictedClassDT = dt.predict(X_test)
print 'DT Accuracy  :',metrics.accuracy_score(expectedClass,predictedClassDT), '--- MSE:',metrics.mean_squared_err
```

Figure 2 – Initialization and simulation of algorithms code on scikit-learn framework