Comparison of Supervised Learning Algorithms Used to Classify Human Activity Recognized by Smartphones

Alaa Alharthi Shreya Lohar

1 Introduction

The human activities record is collected from thirty volunteers while carrying a waist-mounted smartphone with embedded inertial sensors. Thirty volunteers within an age group of 19-48 years carried out this experiment. Each person performed six activities (Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing, and Laying) wearing a smartphone (Samsung Galaxy S II) on the waist. The experiment captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz using an embedded accelerometer and gyroscope. We used the supervised machine learning algorithms to classify those records into one of the corresponding target classes mentioned above. The obtained dataset from Kaggle[1] was randomly partitioned into two sets, where 70% of the volunteers' data was selected to generate the training data and 30% the test data.

Our project aims to compare some of the supervised learning algorithms for classifying human activities into one of the six activities. The linear classifiers applied are linear SVM and logistic regression. For non-linear classification, Multi-layer Perceptron (MLP) and Kernelized-SVM algorithms are used.

2 Methods/Case Study

We have used sklearn[2] package to implement all supervised learning algorithms used in this project. We have split the training dataset into training and validation sets for accurate examination of overfitting. To explore the effect of choosing different parameters values in each algorithm, we have applied the GridSearchCV suggested by sklearn for parameter tuning and cross validation. Briefly, the following paragraphs illustrate the algorithm and hyperparameters tuning.

Kernelized-SVM has been experimented using both Radial Basis Function (RBF) kernel, and polynomial Kernel. One hyperparameter we chose for tuning is regularization parameter C , based on sklearn , as C increases the strength of the regularization decreases. The range of values we chose for C are (1.0, 10, 100, 1000). Another hyperparameter is Kernel coefficient for RBF kernel and the poly kernel. According to sklearn[3], gamma measures how far the influence

[1] https://www.kaggle.com/uciml/human-activity-recognition-with-smartphones
[2] https://scikit-learn.org/stable/supervised_learning.html#supervised-learning
[3] https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html

of a single training point on the decision boundary. Therefore, when gamma has a high value the decision boundary will depend on the close points which will not prevent overfitting. Also, when gamma has low value the decision boundary will take the far points in consideration. Hence, the chosen values for gamma are (1e-3, 1e-4). After the refinement of the grid search ultimately in 5-fold cross-validation, we got the best parameters - the best kernel is RBF, best C is 100, and best gamma is 0.001.

For the linear-SVM, there was only one hyperparameter for tuning which is a regularization parameter C. The loss function applied on linear SVM is the hinge loss. The range values for C used when applying the 5-fold cross-validation are (1.0, 10 , 100). We got the ultimate value for C as 1.0. According to sklearn[4] linear SVM is less sensitive to large C, and it takes longer time to converge. Thus, it justifies that the best value is 1 among the value we chose.

In Multi layer Perceptron (MLP), the sklearn package has more parameters for tuning. We have used GridSearchCV with 5fold cross-validation for searching the best parameters. The optimizers tested are Limited-memory Broyden-Fletcher-Goldfarb-Shanno Algorithm (lbfgs) and adaptive moment estimation (adam). Tangent Hyperbolic (tahn) and Rectified Linear Unit (Relu) activation functions are chosen. The range of learning rate values are (0.0001, 0.005, 0.05). The hidden layers tested are ((50, 10), (15,), (9,)); the first choice has two hidden layer whereas the last two has only one hidden layers. The best solver is Adam with a learning rate of 0.05 having tahn as the best activation function. We got the first hidden layers with 50 units and the second layer with 10 units.

Logistic regression classifier in sklearn package has parameters such as C, max_iter, multi_class, penalty, and solver. We choose to find the best regularization C, and setting max_iter to be 10,000 as the linear model takes longer time than non-linear. As we used 'lbfgs' solver, the L2 (quadratic regularize) was used. The 'lbfgs' solver supports only L2 regularization[5], and does not support L1 (lasso regularize). We used parameter C for regularizing weight. Using Grid-SearchCV with 5 fold classification, the best C value among (0.01, 0.1, 1, 10, 100) was C =1. The value of C for both the linear models was found to be same.

Table 1: The accuracy comparison among the four classifiers

	Training	Validation	Testing
Linear SVM	99%	98%	95%
Kernelized-SVM	100%	99%	96%
Logistic Regression	99.67%	98.6%	95.8%
Multi Layer Perceptron (MLP)	99%	98%	95.2%

[4]`https://scikit-learn.org/stable/modules/svm.html`

[5]`https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html`

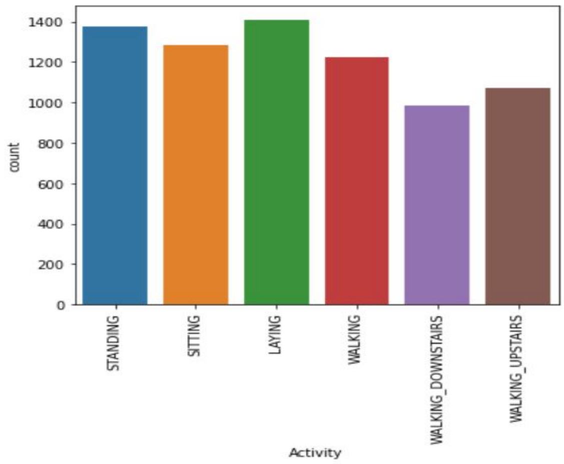

Figure 1: The six classes distribution

3 Results and Discussion

We conducted some experiments on two Linear Supervised learning algorithm- Linear SVM and Logistic Regression and two supervised non linear algorithms - Kernelized-SVM, and Multi Layer Perceptron (MLP) . The hyperparameters for all the four methods are selected using a grid search through a 5-fold cross validation. The dataset used in this experiment has been randomly splitted. The training set size is 5881 data points, the validation set size is 1471 date points, and 2947 data points for the testing set. Figure 1. shows the distribution of the six classes samples of the training dataset. The samples' percentage of Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing, and Laying are 18.22%, 16.66%, 19.05%, 16.83%, 14.25%, and 15.98% respectively. Therefore each class has approximately 17% samples out of the total samples.

Table 1. shows the accuracy for kernel-SVM, Linear-SVM, MLP, and Logistic Regression on training, validation and testing sets. To observe the overfitting, we compared the validation accuracy with the training accuracy in all the four methods. We found that for each supervised classifier, the accuracy of the validation and the training relatively converges, so the difference is roughly 1%. We believe that using a cross validation assisted us in preventing the occurrence of overfitting.

The classification results on testing dataset using the multi class linear SVM is 95% whereas the kernelized-SVM accuracy is 96%. Both of SVM methods performance is good, they differ in number of the maximum iteration to converge. Linear SVM takes roughly 100 thousand iteration while the kernelized SVM takes 10 thousand. The possible justification for this observation is that kernelized SVM create a mapping implicitly to the higher dimensional space, and the RBF kernel maps to infinite dimensional space, so it easily finds the correct classification compared to the linear SVM [2]. Logistic Regression and MLP's accuracy are 95.8%, and 95.2% respectively.

Table 2. compares the precision, recall, and f1-scores values of kernelized SVM to MLP. Also, Table 3. shows the comparison of precision, recall, and f1-scores values between Linear-SVM

and Logistic Regression. The numbers of the first column from 0 to 5 represent the six classes for our problem. Additionally, to observe the errors in terms of false-positive and false-negative, we used the confusion matrix shown in Figure 2. We see a strong response in the diagonal of each of these matrices, which confirms the excellent accuracy that the four classifiers achieved.

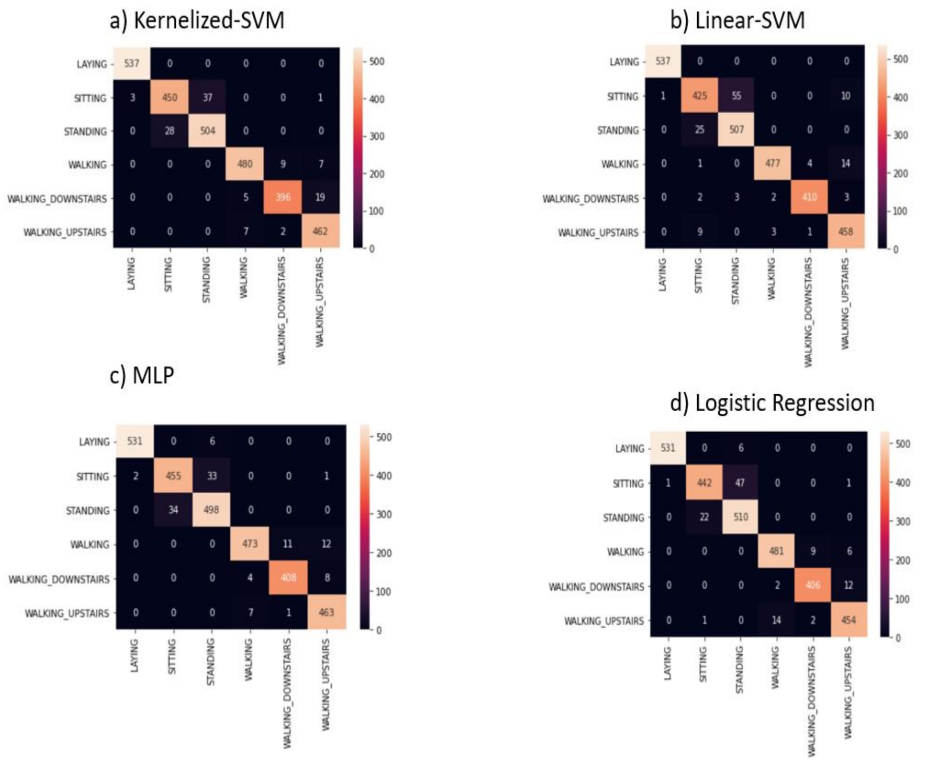

Figure 2: Confusion matrix for Kernelized-SVM, Linear-SVM, MLP, and Logistic Regression

Table 2: The accuracy of testing data using precision, recall, and f1-score. The first three column shows the accuracy of Kernel-SVM and the last three shows the accuracy of MLP

	Precision	Recall	f1-score	Precision	Recall	f1-score	Samples
0	0.99	1.00	1.00	1.00	0.97	0.98	537
1	0.94	0.92	0.93	0.96	0.89	0.92	491
2	0.93	0.95	0.94	0.89	0.96	0.92	532
3	0.98	0.97	0.97	0.99	0.95	0.97	496
4	0.97	0.94	0.96	0.95	0.96	0.95	420
5	0.94	0.98	0.96	0.95	0.99	0.97	471
macro average	0.96	0.96	0.96	0.95	0.95	0.95	2947
weighted average	0.96	0.96	0.96	0.95	0.95	0.95	2947

Table 3: The accuracy of testing data using precision, recall, and f1-score. The first three column shows the accuracy of Linear-SVM and the last three shows the accuracy of Logistic Regression.

	Precision	Recall	f1-score	Precision	Recall	f1-score	Samples
0	1.00	1.00	1.00	1.00	0.99	0.99	537
1	0.92	0.87	0.89	0.95	0.90	0.92	491
2	0.90	0.95	0.92	0.91	0.96	0.93	532
3	0.99	0.96	0.98	0.97	0.97	0.97	496
4	0.99	0.98	0.98	0.97	0.97	0.97	420
5	0.94	0.97	0.95	0.96	0.96	0.96	471
macro average	0.96	0.95	0.96	0.96	0.96	0.96	2947
weighted average	0.96	0.95	0.95	0.96	0.96	0.96	2947

4 Conclusion

To conclude, we found that the state of art highest accuracy on Human Activity Recognition dataset is 96% using SVM and Gaussian kernel [1]. Comparing it with our experiments, we have achieved 96% using kernelized SVM, and 95% using MLP. For linear models, Linear Regression gives the accuracy of 95.82% and linear SVM gives the accuracy around 95% . The results for linear models of Linear SVM and Logistic Regression have similar accuracy. Also, we observed that all the supervised algorithms used in our experiment have shown comparable performance. Although, kernel SVM has the highest accuracy among the four approaches, we conclude that if the supervised algorithms having similar data and set up with the best hyperparameters using a cross validation, they will perform well with similar results. By using sklearn libraries, GridSearchCV function, we got the best hyperparameters for our models. We have learned to apply different machine learning approaches and exploiting the available tools, and packages to solve a real problem.

References

[1] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. *A Public Domain Dataset for Human Activity Recognition Using Smartphones*. 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013. Bruges, Belgium 24-26 April 2013

[2] G. Daqi and Z. Tao. *Support vector machine classifiers using RBF kernels with clustering-based centers and widths*. 2007 International Joint Conference on Neural Networks, 2007, pp. 2971-2976.