Melissa LaHoud

Grand Canyon University

DSC 540: Machine Learning

Dr. Aiman Darwiche

5/12/2021

Ensemble Methods

I used the protocol data containing 12 different activities with the following data collected: timestamp, activity ID, heart rate and IMU's for the hand, chest, and ankle. The IMU sensory collected the temperature, 3D-acceleration data, 3D gyroscope data, 3D-magnetometer data, and the orientation of each body part. There was quite a few missing values due to the fact that the wireless sensors were dropping data or there was a problem with the hardware setup causing connection loss.

I used this data to investigate whether the use of ensemble learning algorithms improve physical activity recognition accuracy compared to the single classifier algorithms, and to compare the classification accuracy achieved by conventional ensemble machine learning methods (bagging, boosting, random forest) and a custom ensemble model comprising four algorithms commonly used for activity recognition such as k nearest neighbor. The I compared the F1 scores of each algorithm. I used python to establish this process and came across some errors that I could not figure out after hours and hours of letting things run as the data set was just too large.

I made sure to clean up the data and I started by naming the columns. I then took out heartrate which seemed to have more NaN's then actual values that were helpful. I then removed all other NaN's to have less data rows and more helpful information. I then calculated the moving average of the accelerometer like they did in the article. The timestamp was in .01 sec increments, so I set n to 1000. I then standardize the moving average. I removed NaN's one more time to ensure all the data was useful to being my analysis. Y was the activity and X was all the standardized moving averages. I tried to pull a smaller sample in order to complete all of the ensembles they did and more.

I was able to achieve one of the ensembles learning algorithms F1 score called the random forest that produced a .99812 F score which is very high. I would expect then that the custom ensemble methods would improve classification performance because when each base classifier has good individual performance and also sufficient diversity (due to having different algorithms), fusion will significantly improve performance. This is where the kNN algorithm would come in play. My goal was to create the table similar to Table 1 below to show all of the F score to see the classification results and compare which ones did better or worse.

	Conventional Ensembles			Individual Classifiers				Custom Ensembles		
	Random Forest	Bagging Decision Tree	Boosted Decision Tree	BDT	kNN	SVM	ANN	WMV Fusion	NB Fusion	BKS Fusion
Lying	80.18	72.76	89.31	73.28	87.36	92.78	91.22	92.69	91.55	86.4
Sitting	76.92	74.34	79.57	70.94	78.97	85.71	82.39	85.5	85.8	79.4
Standing	87.65	82.08	81.6	76.02	84.91	86.04	85.16	88.89	88.7	87.65
Walking	84.5	86.55	88.96	70.07	76.99	87.45	84.34	87.4	84.38	82.02
Running	100	99.12	99.71	85.25	99.71	96.02	94.15	99.12	99.12	99.12
Cycling	95.68	92.93	93.67	92.89	95.62	95.96	86.4	96.61	96.8	96.12
Ascending Stairs	59.89	58.15	53.61	44.44	39.26	48.87	58.51	58.46	57.39	50.87
Descending Stairs	66.67	75.7	56.2	64.26	68.46	72.88	69.6	76.42	79.84	74.49
Average	**81.44**	80.2	80.33	72.14	78.91	**83.22**	81.47	**85.64**	85.45	82.01

Table 1

My python code will also give a goof understanding of the process that took place in order to achieve these results.

References

Chowdhury, K., & Tjondronegoro, D. (2017, September). Ensemble Methods for Classification of Physical Activities... : Medicine & Science in Sports & Exercise. Retrieved from https://journals.lww.com/acsm-msse/Fulltext/2017/09000/Ensemble_Methods_for_Classification_of_Physical.24.aspx