Human Activity Recognition Using Smartphones Data

Jens Kobler

February 10, 2025

Contents

1 Main Objective of Analysis

The main objective of this analysis is to correctly classify test data from the human activity recognition dataset. This version of the report aims at delivering baseline results. Future version could include improvement recommendations stated in the last chapter.

This project was inspired by the "IBM Machine Learning Professional Certificate".

2 Desription of Data

Description from Website (Link):

The Human Activity Recognition Trondheim (HARTH) dataset is a professionally-annotated dataset containing 22 subjects wearing two 3-axial accelerometers for around 2 hours in a free-living setting. The sensors were attached to the right thigh and lower back. The professional recordings and annotations provide a promising benchmark dataset for researchers to develop innovative machine learning approaches for precise HAR in free living.The provided sampling rate is 50Hz. Video recordings of a chest-mounted camera were used to annotate the performed activities frame-by-frame.

Columns:

- timestamp: date and time of recorded sample
- back_x: acceleration of back sensor in x-direction (down) in the unit g
- back_y: acceleration of back sensor in y-direction (left) in the unit g
- back_z: acceleration of back sensor in z-direction (forward) in the unit g
- thigh_x: acceleration of thigh sensor in x-direction (down) in the unit g
- thigh_y: acceleration of thigh sensor in y-direction (right) in the unit g
- thigh_z: acceleration of thigh sensor in z-direction (backward) in the unit g
- label: annotated activity code

Activity code from original dataset: (1) walking, (2) running, (3) shuffling, (4) stairs (ascending), (5) stairs (descending), (6) standing, (7) sitting, (8) lying, (13) cycling (sit), (14) cycling (stand), (130) cycling (sit, inactive), (140) cycling (stand, inactive).

The file with the filename *S006.csv* is selected from the 22 different files. It is used to create baseline results. In Figure 1 the class distribution of the dataset is displayed. As visible the dataset is heavily unbalanced. Class 5 represents sitting (class labels are encoded (again); see next chapter).

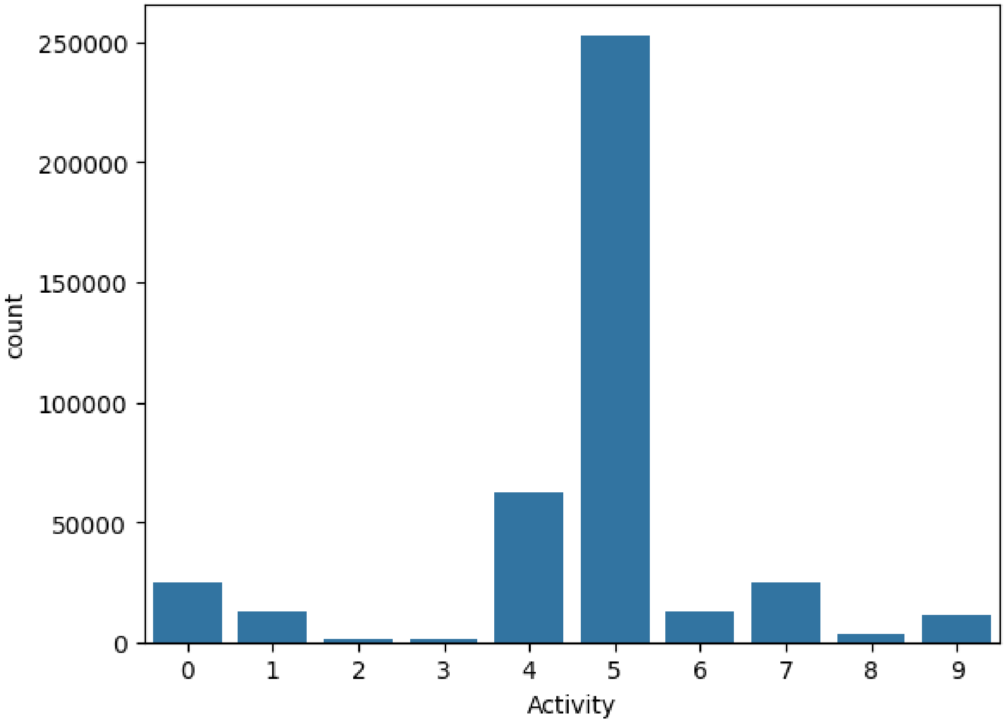


Figure 1: Class distribution.

3 Data Cleaning and Feature Engineering

1. delete timestamp column
2. scale features with minmax scaler
3. encode target labels

4 Variations of Supervised Learning Models

The dataset is split into train and test set with test size 0.2 in a stratified way.

4.1 Logistic Regression (LR)

The standard logistic regression model of the sklearn library is used with solver liblinear.

4.2 K Nearest Neighbor (KNN)

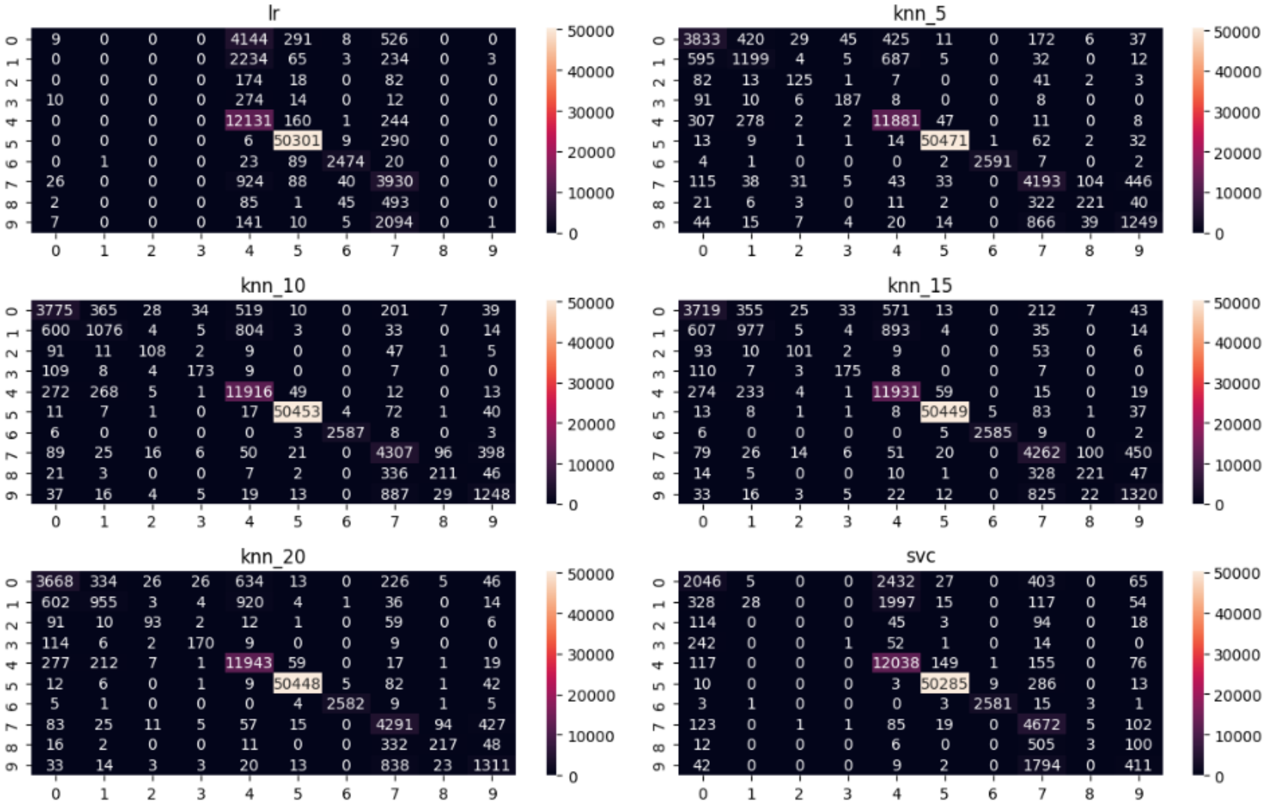
The standard K Nearest Neighbor model of the sklearn librar is used with different k $\in [5, 10, 15, 20]$.

4.3 Support Vector Machine (SVC)

The standard support vector machine of the sklearn library is used.

5 Final Result and reasoning

In Figure 2 the heatmaps for the different models are displayed. The logistic regression model is highly biased to class 4, 5 and 7. The LR model is not able to deal with the unbalanced dataset.



lr

	0	1	2	3	4	5	6	7	8	9
0	9	0	0	0	4144	291	8	526	0	0
1	0	0	0	0	2234	65	3	234	0	3
2	0	0	0	0	174	18	0	82	0	0
3	10	0	0	0	274	14	0	12	0	0
4	0	0	0	0	12131	160	1	244	0	0
5	0	0	0	0	6	50301	9	290	0	0
6	0	1	0	0	23	89	2474	20	0	0
7	26	0	0	0	924	88	40	3930	0	0
8	2	0	0	0	85	1	45	493	0	0
9	7	0	0	0	141	10	5	2094	0	1

knn_5

	0	1	2	3	4	5	6	7	8	9
0	3833	420	29	45	425	11	0	172	6	37
1	595	1199	4	5	687	5	0	32	0	12
2	82	13	125	1	7	0	0	41	2	3
3	91	10	6	187	8	0	0	8	0	0
4	307	278	2	2	11881	47	0	11	0	8
5	13	9	1	1	14	50471	1	62	2	32
6	4	1	0	0	0	2	2591	7	0	2
7	115	38	31	5	43	33	0	4193	104	446
8	21	6	3	0	11	2	0	322	221	40
9	44	15	7	4	20	14	0	866	39	1249

knn_10

	0	1	2	3	4	5	6	7	8	9
0	3775	365	28	34	519	10	0	201	7	39
1	600	1076	4	5	804	3	0	33	0	14
2	91	11	108	2	9	0	0	47	1	5
3	109	8	4	173	9	0	0	7	0	0
4	272	268	5	1	11916	49	0	12	0	13
5	11	7	1	0	17	50453	4	72	1	40
6	6	0	0	0	0	3	2587	8	0	3
7	89	25	16	6	50	21	0	4307	96	398
8	21	3	0	0	7	2	0	336	211	46
9	37	16	4	5	19	13	0	887	29	1248

knn_15

	0	1	2	3	4	5	6	7	8	9
0	3719	355	25	33	571	13	0	212	7	43
1	607	977	5	4	893	4	0	35	0	14
2	93	10	101	2	9	0	0	53	0	6
3	110	7	3	175	8	0	0	7	0	0
4	274	233	4	1	11931	59	0	15	0	19
5	13	8	1	1	8	50449	5	83	1	37
6	6	0	0	0	0	5	2585	9	0	2
7	79	26	14	6	51	20	0	4262	100	450
8	14	5	0	0	10	1	0	328	221	47
9	33	16	3	5	22	12	0	825	22	1320

knn_20

	0	1	2	3	4	5	6	7	8	9
0	3668	334	26	26	634	13	0	226	5	46
1	602	955	3	4	920	4	1	36	0	14
2	91	10	93	2	12	1	0	59	0	6
3	114	6	2	170	9	0	0	9	0	0
4	277	212	7	1	11943	59	0	17	1	19
5	12	6	0	1	9	50448	5	82	1	42
6	5	1	0	0	0	4	2582	9	1	5
7	83	25	11	5	57	15	0	4291	94	427
8	16	2	0	0	11	0	0	332	217	48
9	33	14	3	3	20	13	0	838	23	1311

svc

	0	1	2	3	4	5	6	7	8	9
0	2046	5	0	0	2432	27	0	403	0	65
1	328	28	0	0	1997	15	0	117	0	54
2	114	0	0	0	45	3	0	94	0	18
3	242	0	0	1	52	1	0	14	0	0
4	117	0	0	0	12038	149	1	155	0	76
5	10	0	0	0	3	50285	9	286	0	13
6	3	1	0	0	0	3	2581	15	3	1
7	123	0	1	1	85	19	0	4672	5	102
8	12	0	0	0	6	0	0	505	3	100
9	42	0	0	0	9	2	0	1794	0	411

Figure 2: Heatmaps.

In Figure 3 the results for different metrics and models is illustrated. The K-Nearest-Neighbor algorithm achieves the highest results. The chocie between the different k is not having a big difference. Pick KNN algorithm for classification.

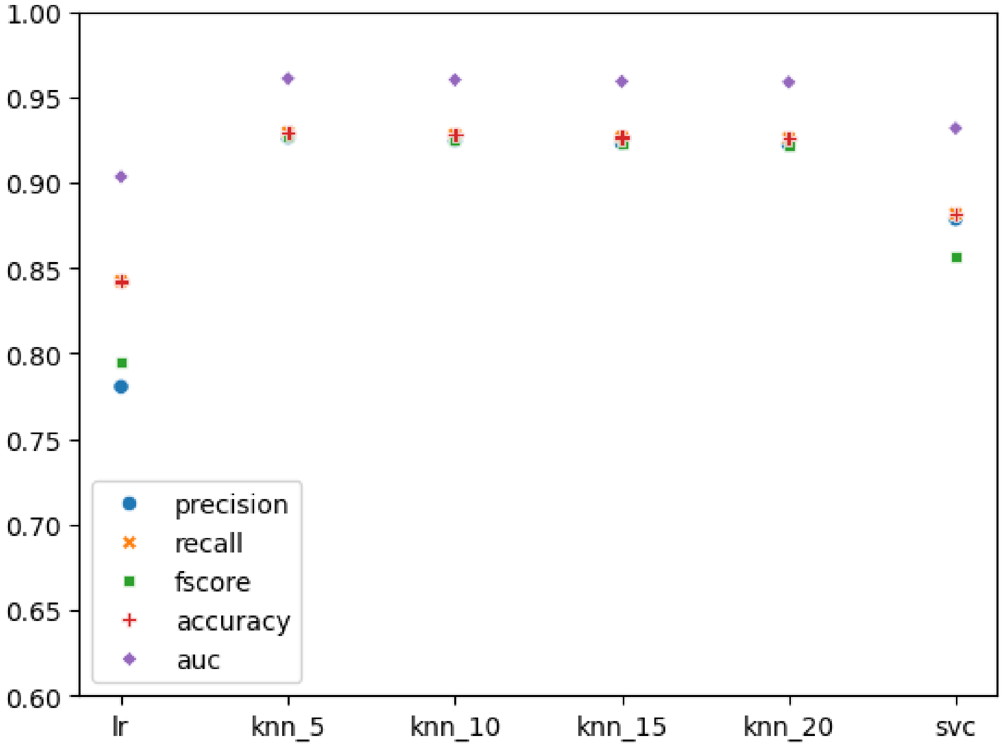


Figure 3: Results.

6 Flaws in model and Revisiting Analysis

- Only one out of 22 tables were used. Concatenate all tables from all participants.
- One could balance the classes. For example one can take the minimum of all classes as a number of samples.
- One could try different unsupervised (clustering) methods for example KMeans.
- One could train a deep neural network.
- Try different hyperparameters.
- One could try Principal Component Analysis (PCA), to reduce the number of features. However, all features seem necessary considering the physics.
- One could visualize the features space and the labeled samples with TSNE or PCA...