Daily Tracker

Smartphone Computing Term Project (Autumn, 2017)

Group

Abhisek Chowdhury (16CS71P07)
Aditi Chandra (16CS71P02)

Mentor

Snigdha Das

1. Problem Statement

Monitoring daily
activity routine
to know

HOW WAS YOUR DAY ?

2. Motivation

Analysis of human daily activities is important for **Daily Routine Regularity Checking**

3. Road Map

1. Data Acquisition

- Gathering of labelled Accelerometer sensor data to generate ground-truth
- Get current day's unlabelled Accelerometer sensor data

2. Data Pre-processing

- Filtration of the dataset
- Smoothening
- Displacement, velocity Measurement

3. Detection of Daily Activity

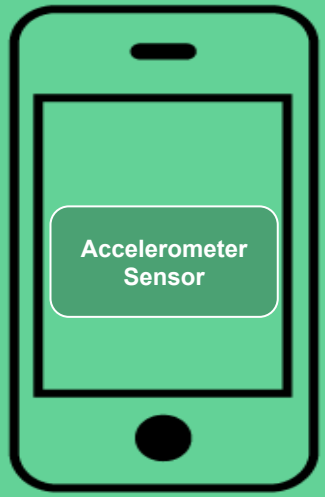
- Feature Extraction
- Classification using Decision Tree
- Activity Detection

4. Daily Tracking

How was your day?

Today was a regular day.
Or
Today was an irregular day !

4. Framework



Data Acquisition

Data Filtration

Smoothening

Displacement

Velocity

Feature Extraction

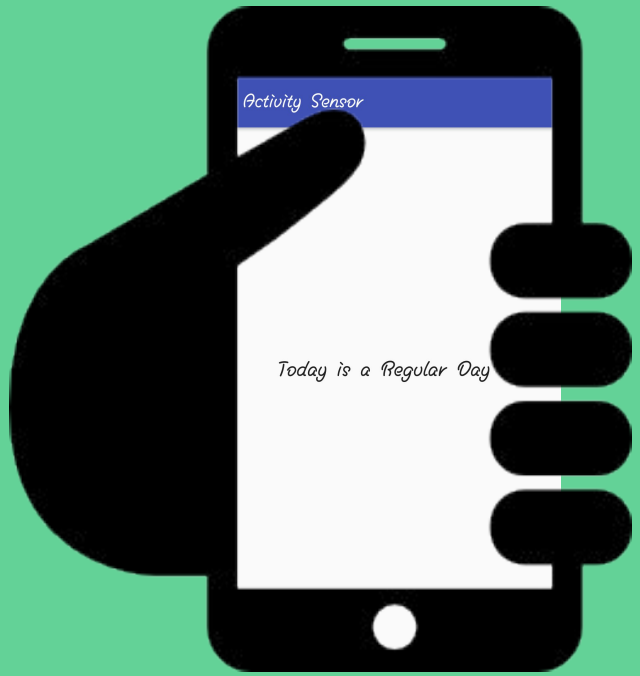
Predictive Model

Detection

How was your day !

Regular / Irregular ?

Client - Server Model



Notification

History Data ,
Today's Data

Track the day

Personalized Model

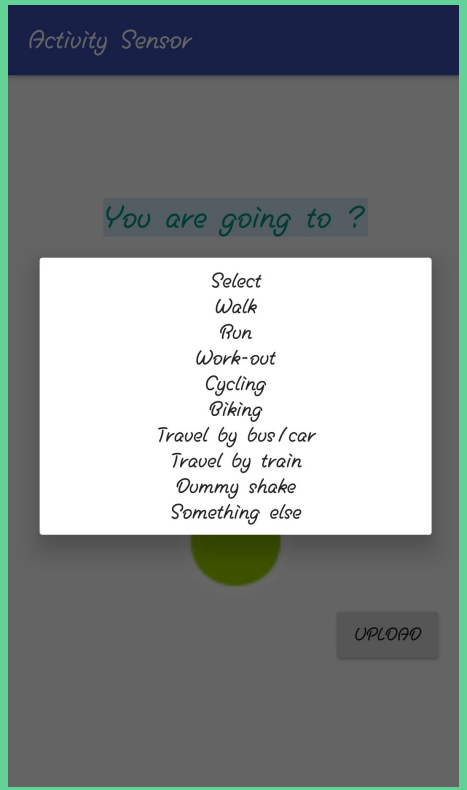
Last n days log

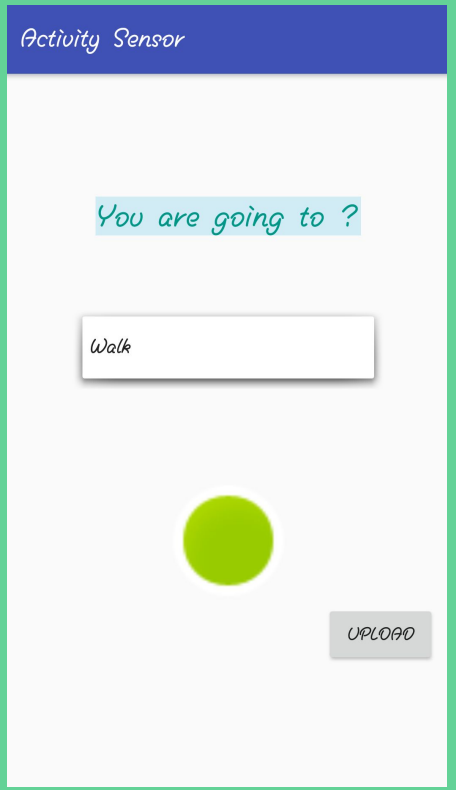
CLIENT

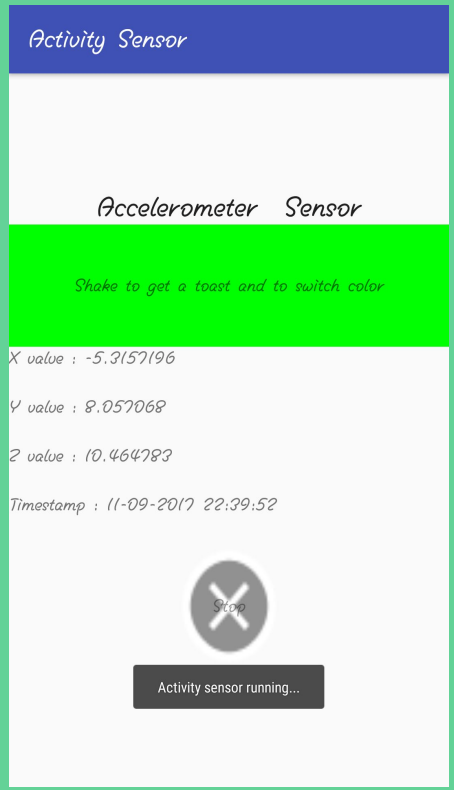
SERVER

5. Data Preprocessing

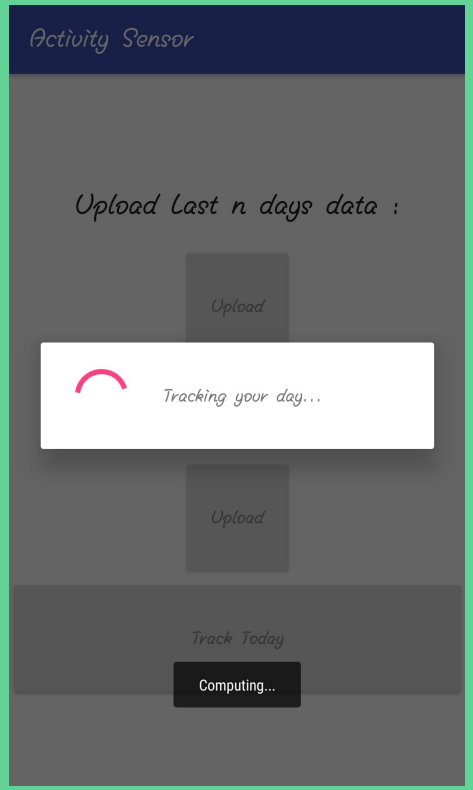
Data Acquisition







Track The Day !



Data Filtration and Smoothening

ALGORITHM FILTRATION

for each tuple $tp(x_i, y_i, z_i, t_i, l_i)$

threshold$_{acc} \leftarrow 2$

acceleration $\leftarrow \sqrt{ (x_i^2 + y_i^2 + z_i^2)}$

if (acceleration > threshold$_{acc}$)

$filter\_tp(x_i, y_i, z_i, t_i, l_i) \leftarrow tp(x_i, y_i, z_i, t_i, l_i)$

ALGORITHM SMOOTHENING

for every $min(t_i)$ in each tuple $filter\_tp(x_i, y_i, z_i, t_i, l_i)$

calculate mean(x), mean(y), mean(z)

for every min in t_i

$smooth\_tp(x_i, y_i, z_i, min(t_i), l_i) \leftarrow filter\_tp(mean(x), mean(y), mean(z), t_i, l_i)$

	A	B	C	D	E
1	x-value	y-value	z-value	timestamp	travel_mc
2	5.568634	29.42409	-0.97112	7/9/2017 10:01	Biking
3	-1.76418	13.06555	4.343353	7/9/2017 10:01	Biking
4	1.101227	20.59097	-0.4758	7/9/2017 10:01	Biking
5	-0.70773	14.25478	3.088318	7/9/2017 10:02	Biking
	0.465942	15.17841	0.872559	7/9/2017 10:02	Biking
	0.465942	15.17841	-0.04628	7/9/2017 10:02	Biking
	1.383591	18.30225	0.081726	7/9/2017 10:02	Biking
	0.6801	13.8468	0.958694	7/9/2017 10:02	Biking
10	-0.59767	15.0827	3.545334	7/9/2017 10:02	Biking
11	1.004318	13.90782	0.123611	7/9/2017 10:02	Biking
12	0.053177	16.12238	0.896484	7/9/2017 10:02	Biking
13	-0.89796	13.82646	1.439652	7/9/2017 10:02	Biking
14	1.632446	16.78639	-0.54997	7/9/2017 10:02	Biking
15	1.766434	14.19377	-0.07858	7/9/2017 10:02	Biking
16	-3.90910	-15.8455	-0.89334	7/9/2017 10:03	Biking
17	-2.41142	-15.6304	-2.96793	7/9/2017 10:03	Biking
18	-1.30116	-13.6934	-4.13802	7/9/2017 10:03	Biking
19	1.393158	-15.9414	-2.99304	7/9/2017 10:03	Biking

Data Acquisition

	A	B	C	D	E
1	x-value	y-value	z-value	timestamp	travel_mode
2	7.090394	-5.9873	13.54592	7/9/2017 9:57	Biking
3	10.4464	8.637634	13.58801	7/9/2017 9:58	Biking
4	3.419159	10.77203	9.686996	7/9/2017 9:59	Biking
	0.875107	14.69507	0.36528	7/9/2017 10:02	Biking
8	0.951088	16.02367	1.382828	7/9/2017 10:04	Biking
9	-0.41791	15.21311	2.11832	7/9/2017 10:05	Biking
10	1.869092	21.68641	1.739478	7/9/2017 10:06	Biking
11	-0.33147	16.82826	1.933777	7/9/2017 10:07	Biking
12	0.384927	15.10731	1.231995	7/9/2017 10:08	Biking
13	0.053177	16.12238	0.896484	7/9/2017 10:09	Biking
14	0.833639	14.93554	0.270365	7/9/2017 10:10	Biking
15	-0.92378	-15.0781	-0.88617	7/9/2017 10:11	Walk
16	-3.93853	-13.759	-0.01303	7/9/2017 10:12	Walk
17	-2.69458	-13.9913	0.371262	7/9/2017 10:13	Walk
18	-4.55779	-14.5875	0.776047	7/9/2017 10:14	Walk
19	-2.45689	-20.7076	0.930671	7/9/2017 10:15	Walk

Filtration and Smoothening

Displacement Measurement

$$\text{displacement} = \iint_0^T \text{acceleration}$$

ALGO DISPLACEMENT

for each tuple smooth_tp(x_i, y_i, z_i, min(t_i), l_i)

init t ← 60

$\text{acceleration}_i \leftarrow \sqrt{(x_i^2 + y_i^2 + z_i^2)}$

$\text{displacement}_i \leftarrow \iint_0^t \text{acceleration}_i$

	A	B	C	D	E	F
1	x-value	y-value	z-value	timestamp	travel_m	disp(m)
2	7.090394	-5.9873	13.54592	7/9/2017 9:57	Biking	280.21
3	10.4464	8.637634	13.58801	7/9/2017 9:58	Biking	291.5965
4	3.419159	10.77203	9.686996	7/9/2017 9:59	Biking	310.4425
5	0.613098	15.05638	-0.32445	7/9/2017 10:01	Biking	322.5362
6	0.875107	14.69507	0.36528	7/9/2017 10:02	Biking	313.3628
7	0.856689	17.07473	1.52843	7/9/2017 10:03	Biking	355.5822
8	0.951088	16.02367	1.382828	7/9/2017 10:04	Biking	341.0557
9	-0.41791	15.21311	2.11832	7/9/2017 10:05	Biking	352.6828
10	1.869092	21.68641	1.739478	7/9/2017 10:06	Biking	332.9181
11	-0.33147	16.82826	1.933777	7/9/2017 10:07	Biking	327.4711
12	0.384927	15.10731	1.231995	7/9/2017 10:08	Biking	319.5812
13	0.053177	16.12238	0.896484	7/9/2017 10:09	Biking	288.0737
14	0.833639	14.93554	0.270365	7/9/2017 10:10	Biking	210.4806
15	-0.92378	-15.0781	-0.88617	7/9/2017 10:11	Walk	30.56619
16	-3.93853	-13.759	-0.01303	7/9/2017 10:12	Walk	31.15578
17	-2.69458	-13.9913	0.371262	7/9/2017 10:13	Walk	27.12662
18	-4.55779	-14.5875	0.776047	7/9/2017 10:14	Walk	25.65132
19	-2.45689	-20.7076	0.930671	7/9/2017 10:15	Walk	29.4368

Displacement

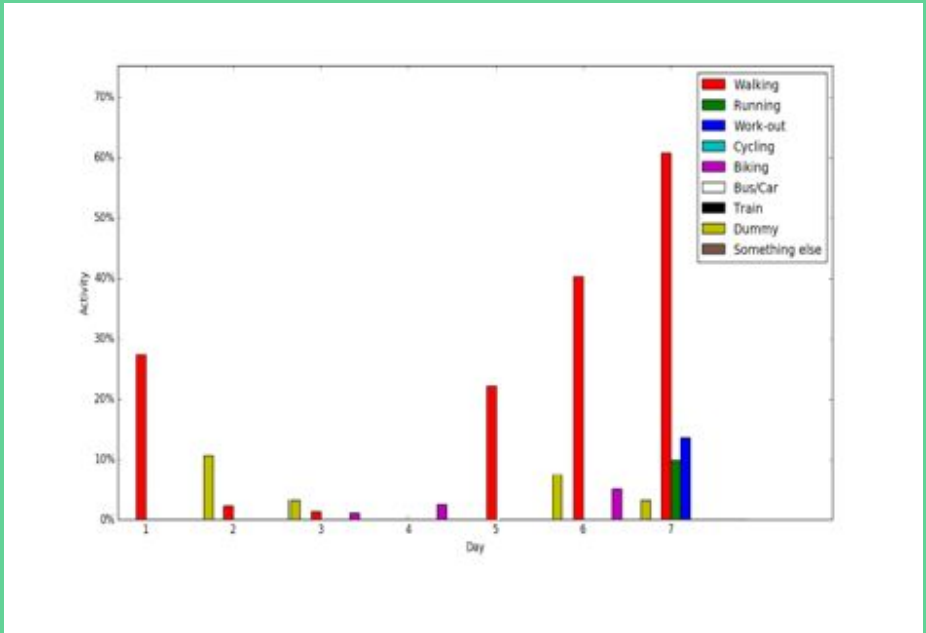
6. Feature Extraction

Feature Extraction

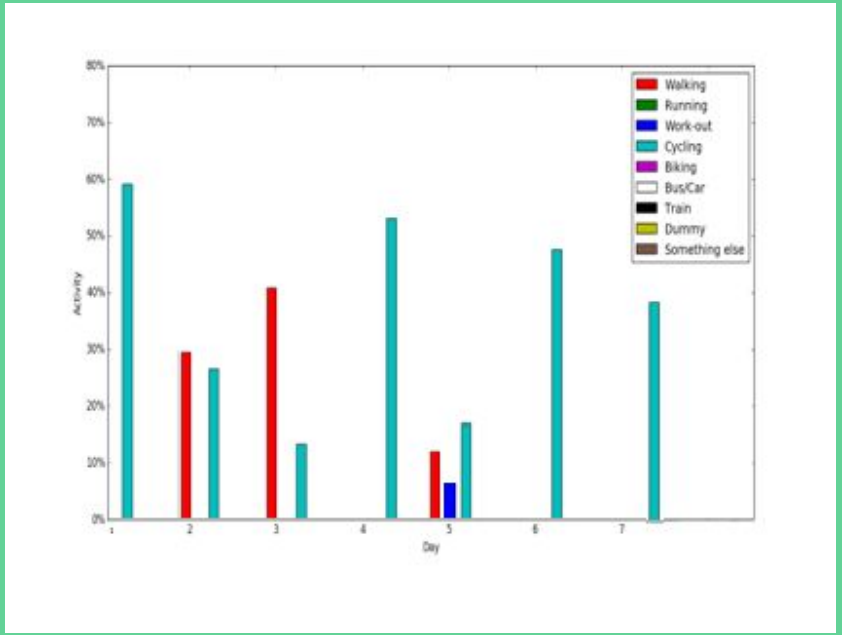
1. Displacement
2. Velocity
3. Acceleration
4. Time of the day (Early Morning, Morning, Day, Afternoon, Night, Midnight)
5. Day of the week (Sunday = 1st ... Saturday = 7th)

7. Data Analysis

Activities



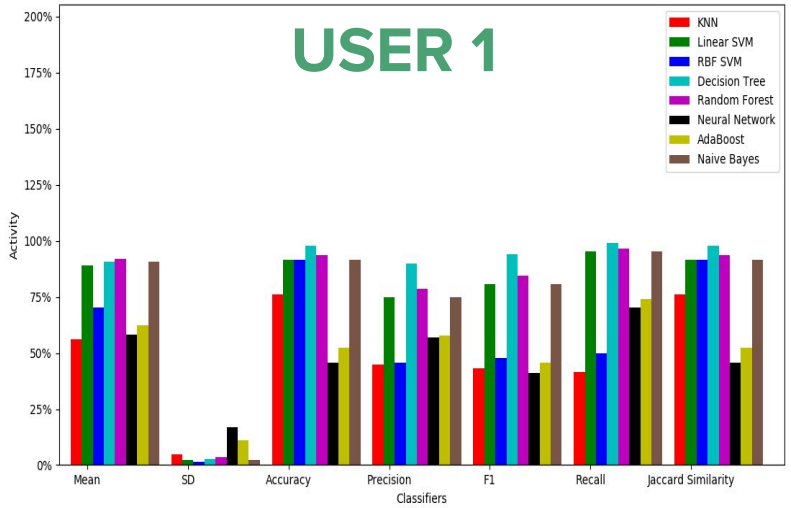
USER 1

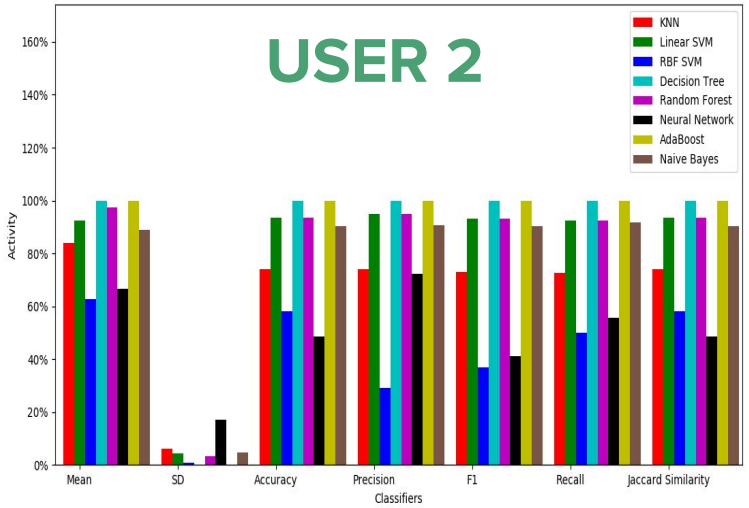


USER 2

7. Model Building

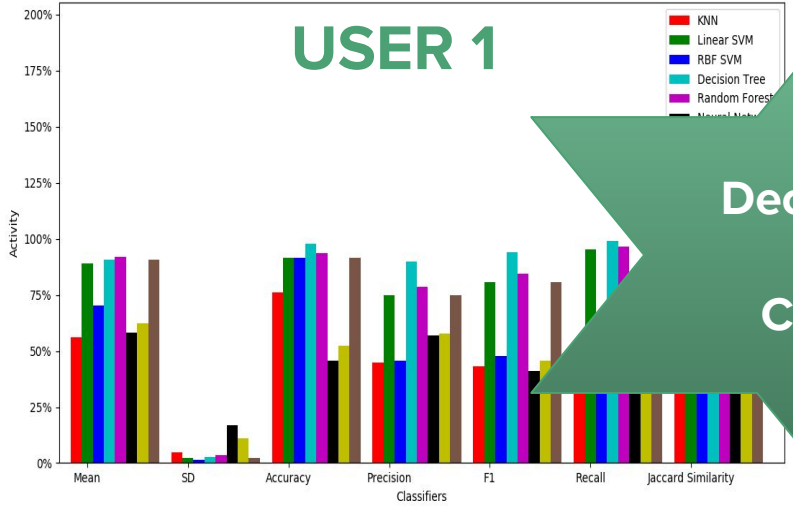
Classifications & Accuracy



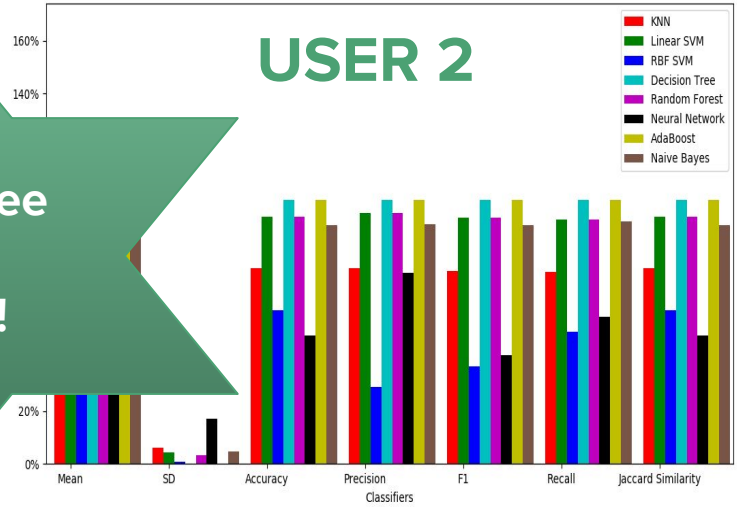


Train Data : 75 % , Test Data : 25%

Classifications & Accuracy



Decision Tree
Is
Choosed !



8. Methodology

Algorithm

ALGO DAILY_TRACKER (model , Reg[n] , dayT)

featuresT ← extract_features (dayT)

classify_activities (model , featuresT)

threshold ← min (cross_correlation (Reg[n]))

c ← min (cross_correlation (Reg[n] , dayT))

if (c > threshold)

    Alert (“Regular Day”)

    Reg[n] ← Reg[n] U dayT

otherwise

    Alert (“ Irregular Day”)

    Ignore dayT

INPUT

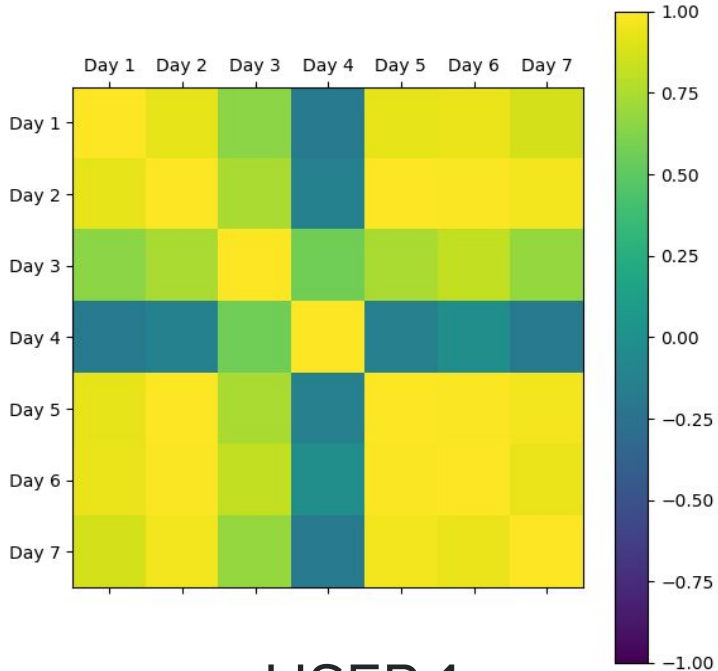
model : The personalised model for each user

Reg[n] : Last N days data

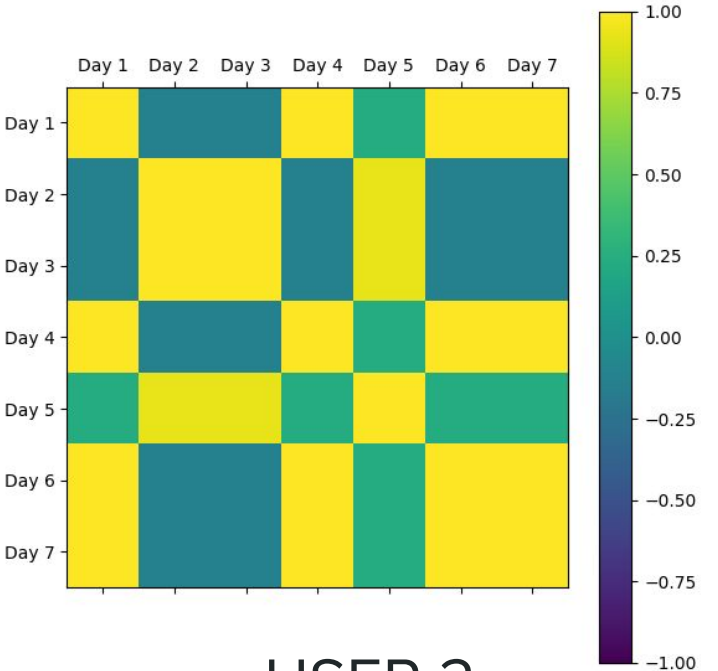
dayT : The current / target day

9. Results

Cross - Correlation for Last n Days



USER 1

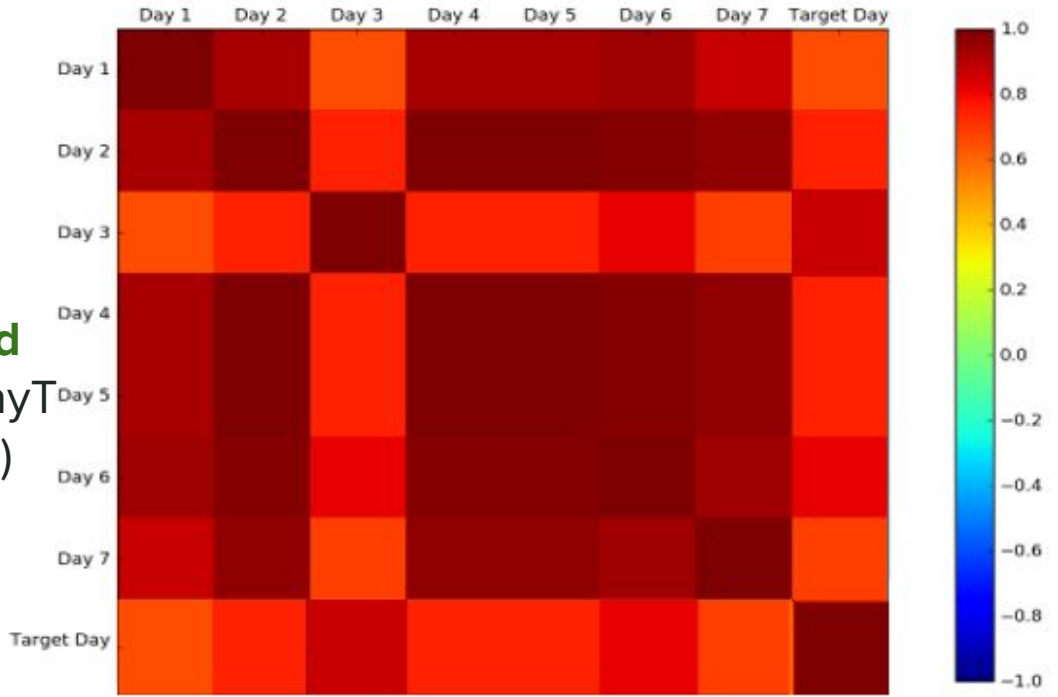


USER 2

Daily Tracker detects as Regular Day

threshold = 0.440
c = 0.463

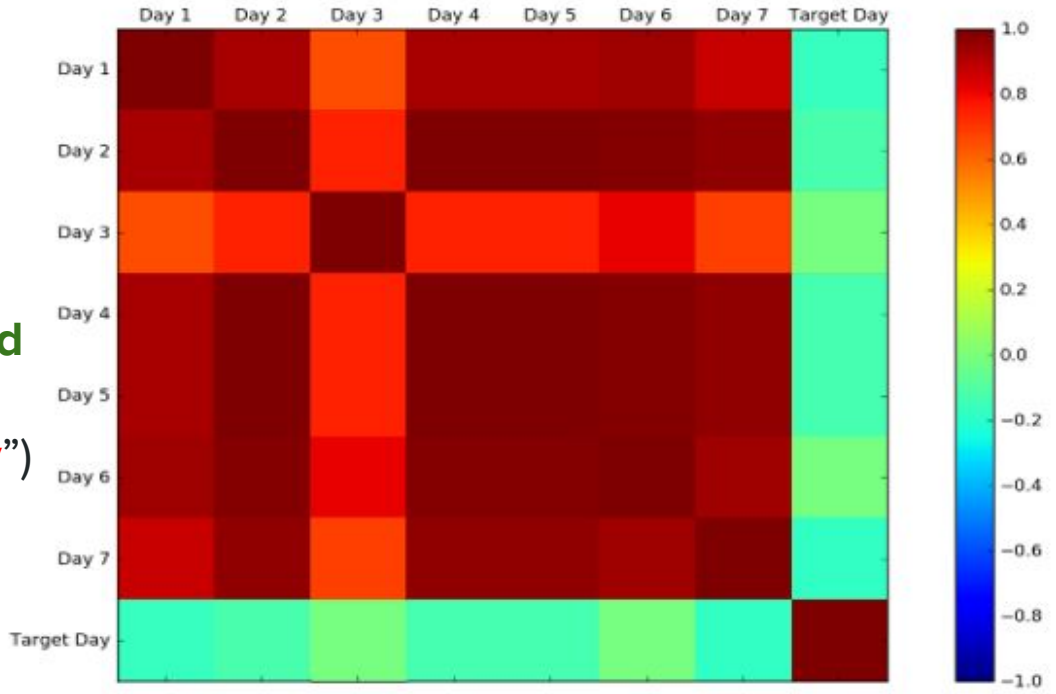
Hence, **c > threshold**
Reg[n] = Reg[n] U dayT
alert (“**Regular Day**”)



Daily Tracker detects as Irregular Day

threshold = 0.440
c = - 0.172

Hence, **c < threshold**
ignore dayT
alert (“**Irregular Day**”)

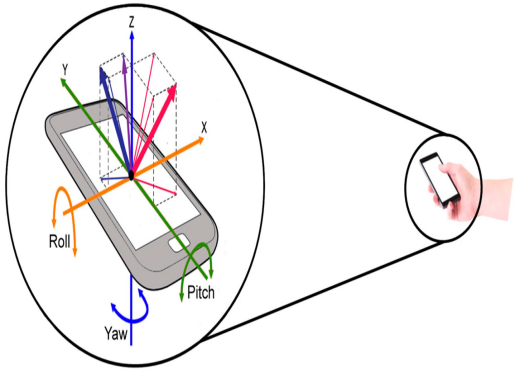


DAILY TRACKER

A system to track daily activity routine

to know

HOW WAS YOUR DAY ?



Thank You !