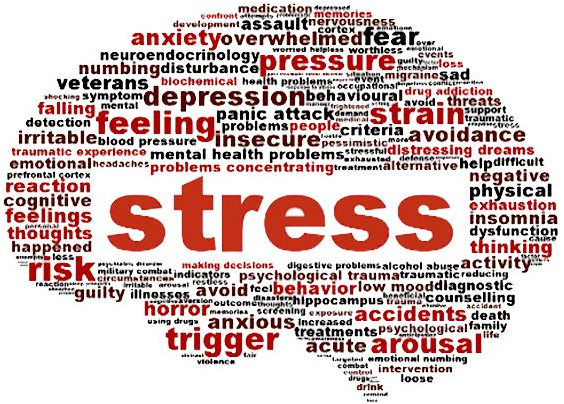
Detecting Stress with Wearable Sensor Data and Machine Learning

by Deep Patel

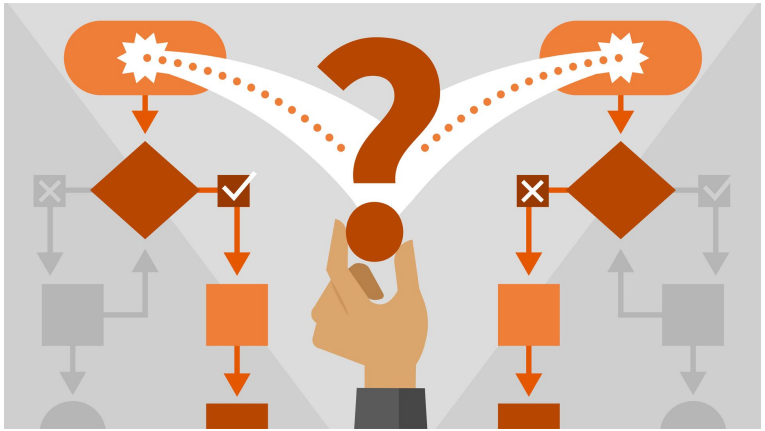
Motivation

- Stress is known to have **detrimental health effects** and mental health is a big concern nowadays
- **Detection** can lead to **Mitigation**
- Also useful to detect stress in automobile driving applications
- With the rise of **wearables**, ML models to detect stress can be very **useful**
- Most **current algorithms** use techniques that are **not suitable** for **real-time** applications



Goal

Create a **fast** and **low-computation cost** machine learning model that can **classify** the user's binary **stress state** in **real-time** with **high accuracy** using common **physiological signal data** obtained from modern **wearables**



Pipeline

Data Preprocessing & Analysis

Feature Extraction

Build Model with Machine Learning

The Dataset: WESAD (Wearable Stress & Affect Detection)

15 subjects (graduate students) attached with sensors during 5 labelled individual sessions

Sessions

Baseline
Given neutral magazines to read for 20 min

Amusement
Showed funny video clips for 7 min

Meditation 1
Performed controlled breathing meditation exercises

Stress
Given realistic public speaking & mental arithmetic tasks

Meditation 2
Performed controlled breathing meditation exercises

Sensors

electrical activity - **ECG**

moisture/sweat - **EDA**

muscle activity - **EMA**

Temperature

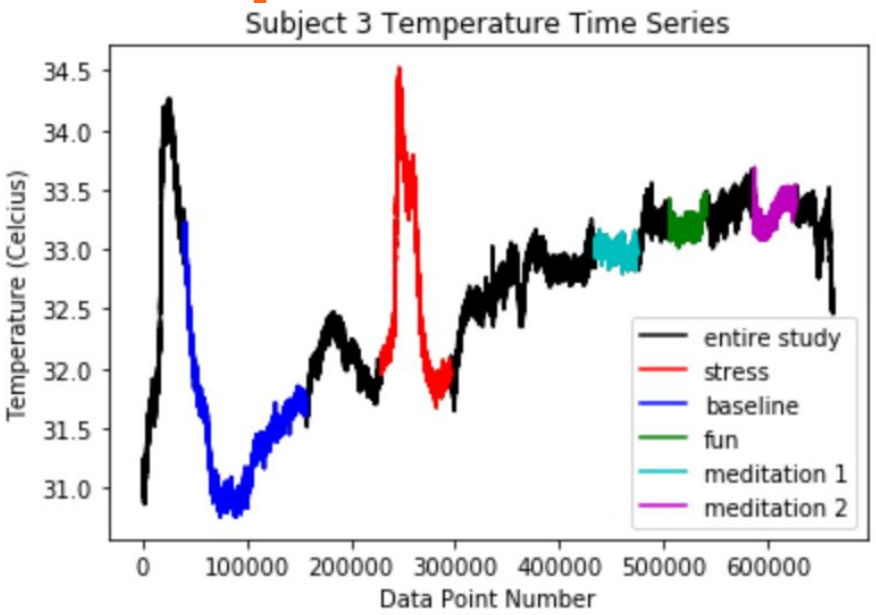
Respiration

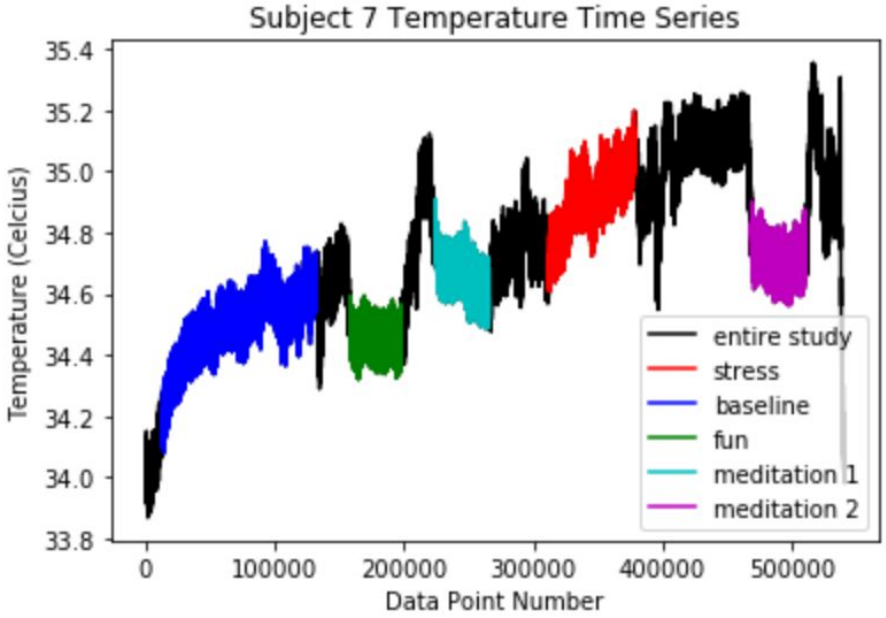
XYZ Acceleration

Preprocessing & Segmentation

1. **Parsing** csv files & loading into **DataFrames** in Spark
2. **Reducing size** of data from 700 Hz to 100 Hz
3. Converting **raw signals** to sensor **SI units**
4. **Labelling** data using time stamps and frequency to decide which **discrete intervals** correspond to which **sessions**

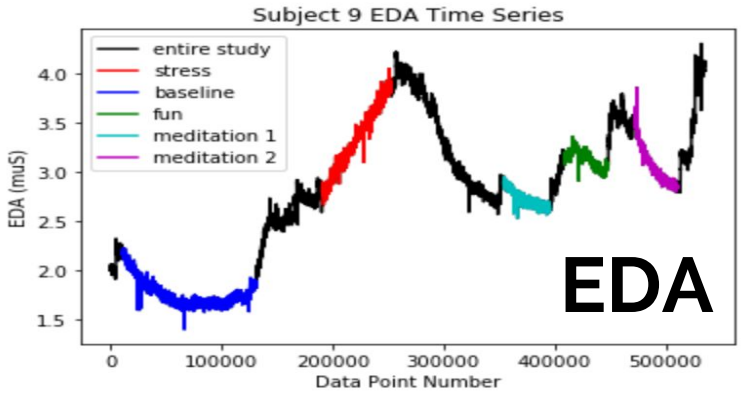
Analyzing the Raw Data - Temperature

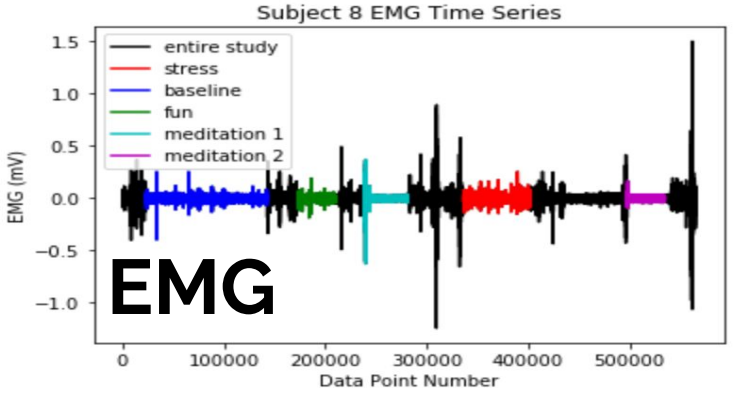


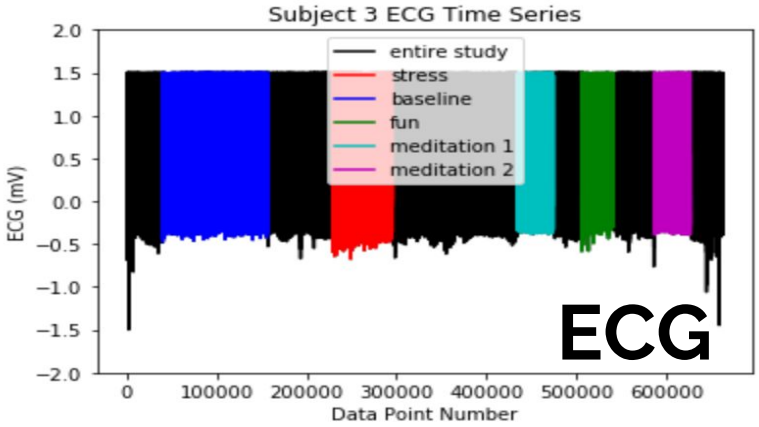


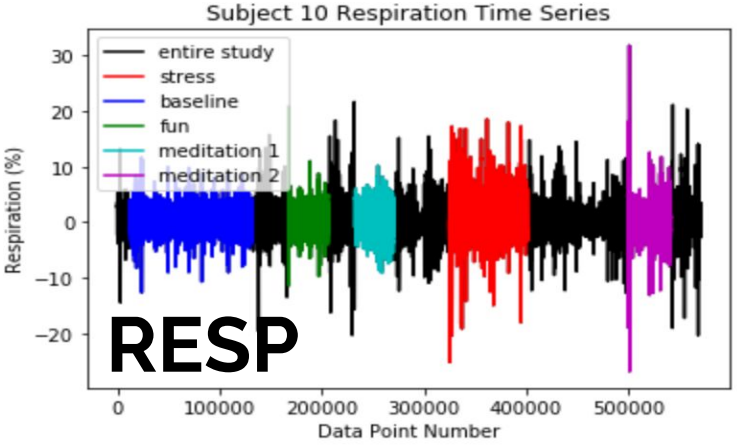
Stress in red

Analyzing the Raw Data

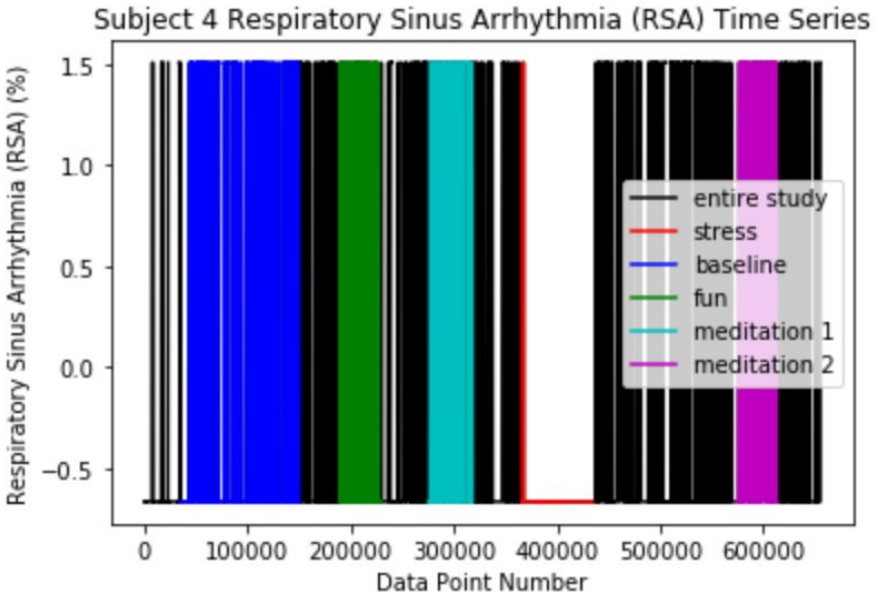


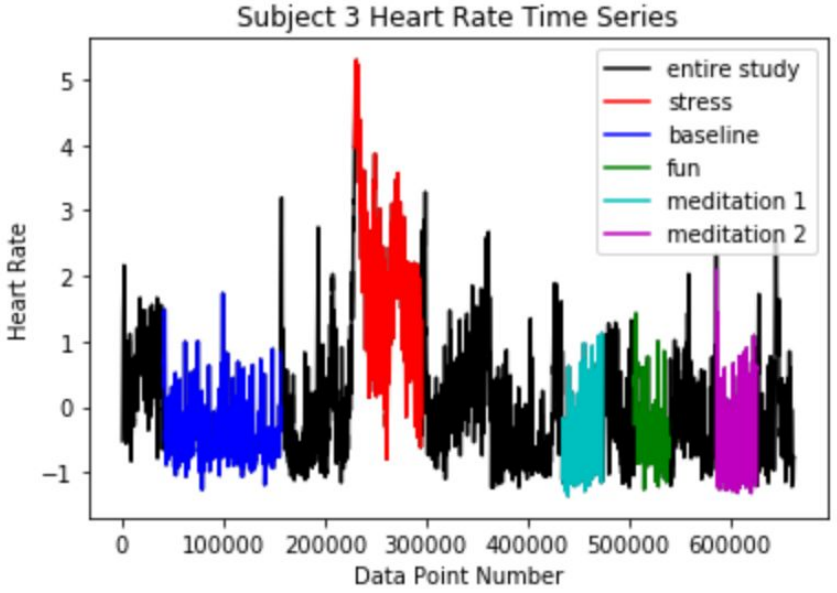






Extracting Features





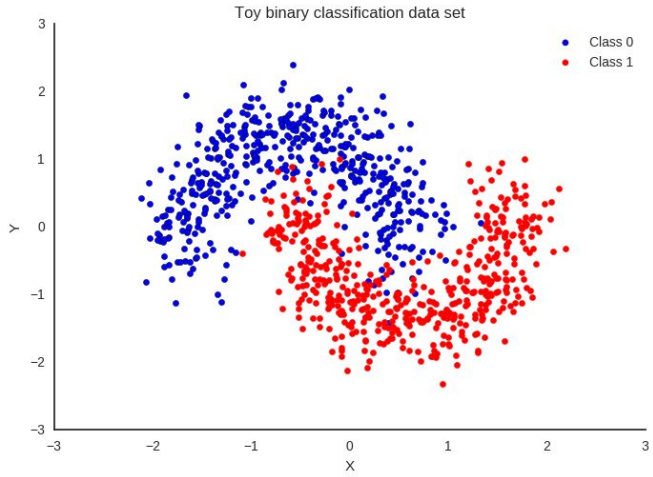
How to evaluate stress?

- A **sliding window** of **60 seconds** is used for the sensors with **50% overlap**
- Essentially a **stress prediction** is made every **30 seconds**
- **Features**: Temperature slope, RSA, Frequency of Heartbeats, RMS of HRV, Mean & STD of HRV, etc

Modeling the data

Classification algorithms used:

- **K-Nearest Neighbors**
- **Adaboost** (Adaptive Boosting)



How well does it do?

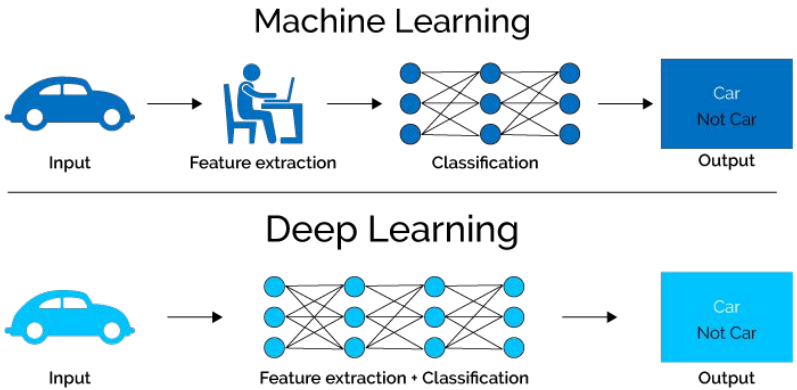
- **Leave-One-Subject-Out** (LOSO) Cross Validation Procedure
- **Train** on n-1 subjects data and **test** on the one left out.
- Do this for all n configurations & take average

Results

	F1-score	Accuracy	Prediction Time
KNN	69.19 ± 1.03	76.28 ± 0.83	**1.22** sec
AdaBoost	**85.53** ± 0.52	**88.79** ± 0.47	1.57 sec

Future Goals

vs.



questions?