CS3244 Project

40

Adithya Narayan Rangarajan Sreenivasan A0206420X
Samuel Suther David A0182488N
Soh Yi Zhi Zeke A0190203W
Umar Bin Moiz A0180913H
Yap Dian Hao A0184679H

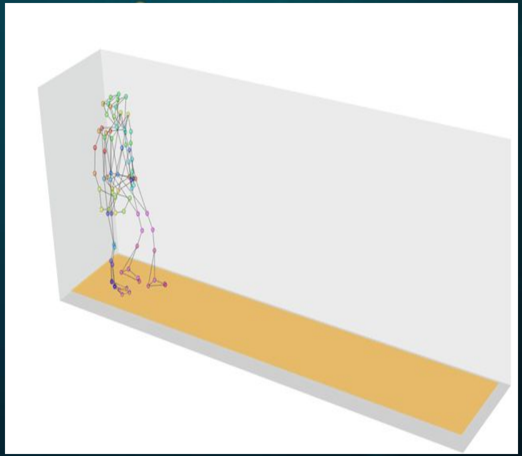
Smartphone-based Human Activity Recognition Dataset with Postural Transitions (SBHAR) Dataset [1]

Remote monitoring of physiotherapy patients using HAR analysis

Dataset Breakdown

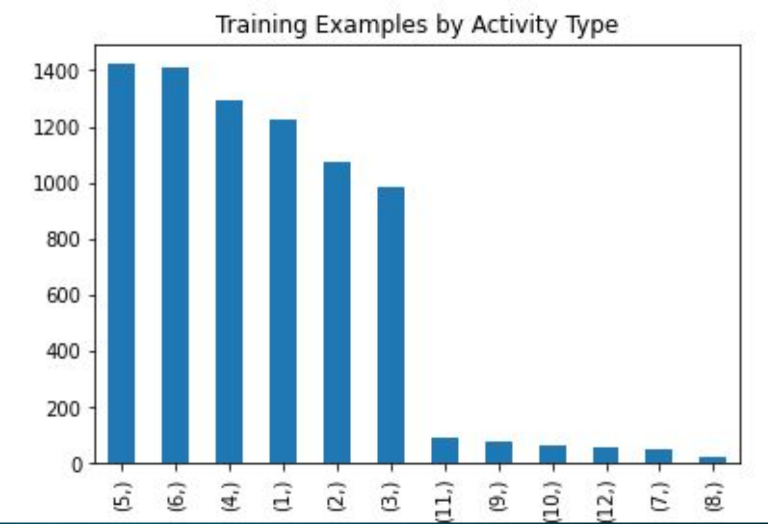
Basic Activities
- Walking
- Walking upstairs
- Walking downstairs
- Sitting
- Standing
- Laying

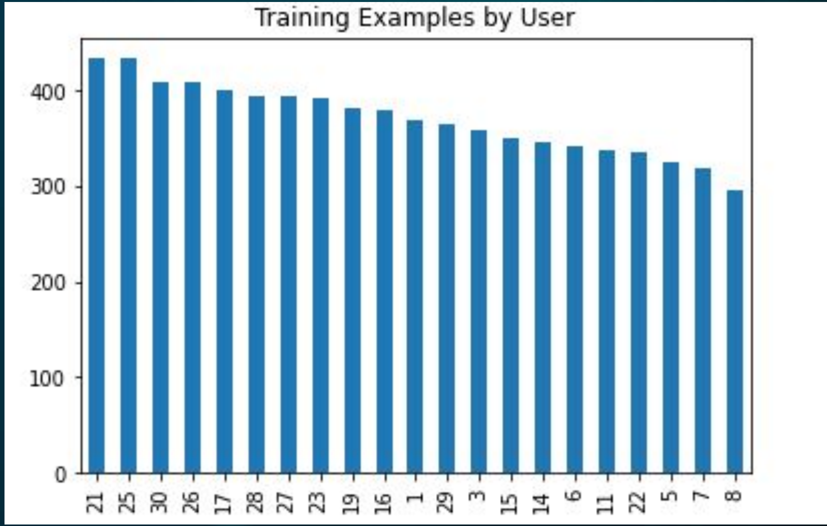
Postural Transitions
- Stand-to-sit
- Sit-to-stand
- Sit-to-lie
- Lie-to-sit
- Stand-to-lie
- Lie-to-stand



Data source:
- 30 volunteers aged 19-48
- Done with Samsung Galaxy S embedded with accelerometer and gyroscope

Dataset Breakdown





Number of samples: 10929

Related Works

Ubiquity of smartphone over the years

- 6.37 billion smartphone users globally[2]
- Easier to collect Data and perform analysis
- Since its first work in the late 90s by Foerter and Smeja[3], field of HAR analysis has become more popular with a focus on wearable devices to classify human activity [4]

Applications of HAR:

- surveillance, healthcare and human-computer interactions.

For this dataset in particular, classification of activity being completed and identification of which user is completing the action[5] have been popular areas of interests.

Related Works

Classification

- Feature engineering
    - Handpicked features [7]
    - Mapping with different window
- Capstone project by Xiaoshan Sun [6]
    - Utilised PCA for dimensionality reduction with 103 features
    - Models explored: SVC (Support vector classifier) and CNN
- Multilayer Architectures [9][10][11][12][13]

Paper	Method	Accuracy
[40]	Hidden Markov Models	83.51
[41]	Dynamic Time Warping	89.00
[42]	Handcrafted Features + SVM	89.00
[38]	Convolutional Neural Network	90.89
[13]	Hidden Markov Models	91.76
[43]	PCA + SVM	91.82
[43]	Stacked Autoencoders + SVM	92.16
[8]	Hierarchical Continuous HMM	93.18
[9]	Convolutional Neural Network	94.79
[10]	Convolutional Neural Network	95.18
[9]	FFT + CNN Features	95.75
	MANN	**96.24**
[11]	Handcrafted Features + SVM	96.37
[39]	Convolutional Neural Networks	97.63

Credits: Acampora G, Minopoli G, Musella F, Staffa M. Classification of Transition Human Activities in IoT Environments via Memory-Based Neural Networks. Electronics. 2020; 9(3):409. https://doi.org/10.3390/electronics9030409

01 Problem statement and motivation

02 Implementation

03 Evaluation

04 Results interpretation

Motivation

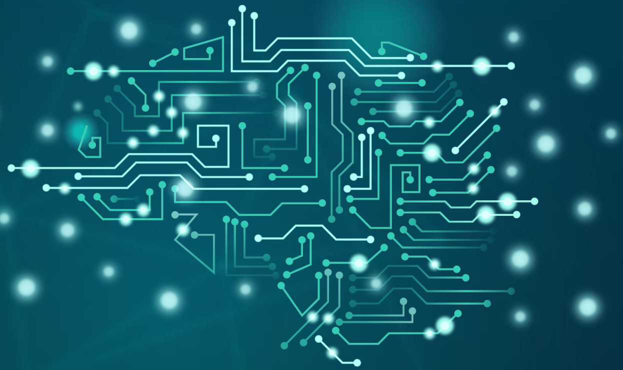
Current pandemic

- Discouraged from going outside unnecessarily.
- Hospital resources are valuable and need to be reserved for those in more urgent need.

HAR analysis can help Physiotherapist monitor movements made that the patients are performing exercise regimes correctly to prevent potential injuries.

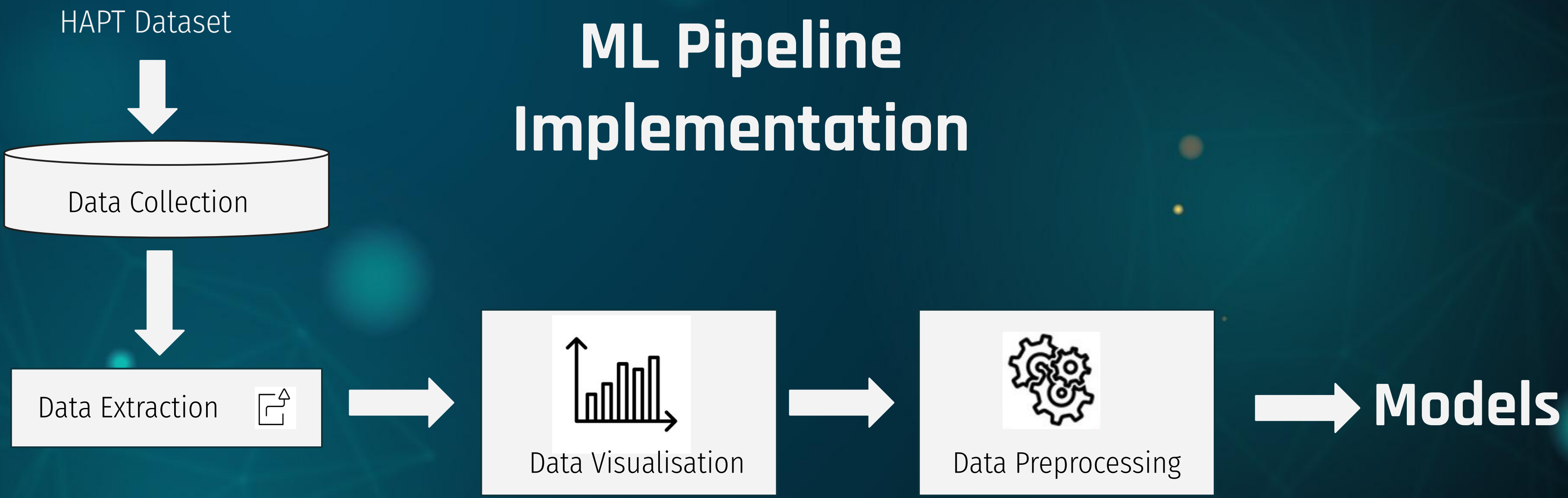
Problem statement

As a physiotherapist, with the current pandemic, I want to be able to provide remote therapy for my patients. This will be helpful for patients who are close to recovery and require less maintenance. These patients are able to follow and complete the recommended exercises on their own correctly so that they can recover faster and I can provide attention to newer patients and patients that require more dedicated care. Furthermore, I can reduce the movements within hospitals.



Implementation of ML Pipeline

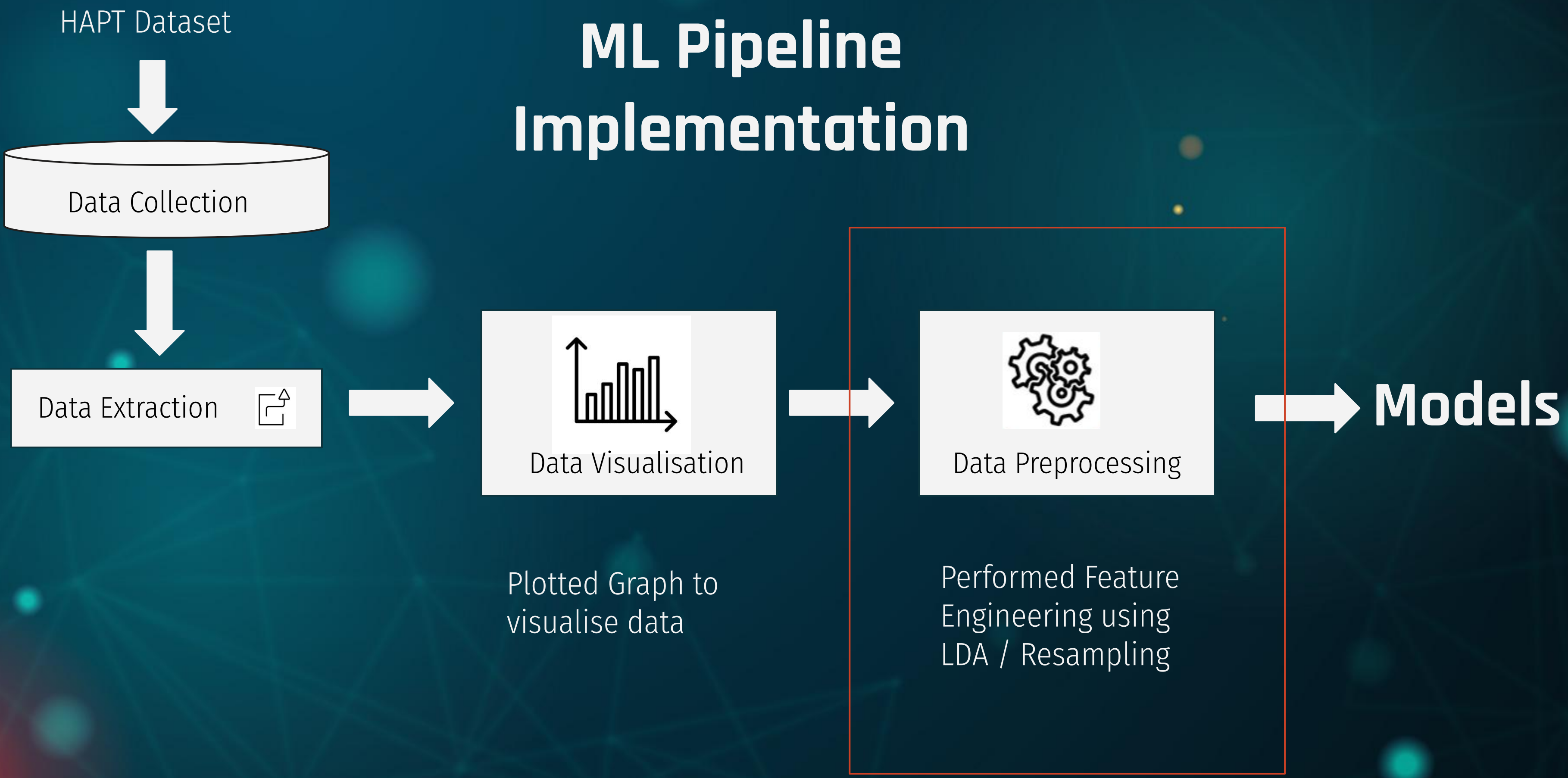
ML Pipeline Implementation



Plotted Graph to visualise data

Performed Feature Engineering using LDA / Resampling

ML Pipeline Implementation



Curse of Dimensionality

561 features obtained by calculating variables from time and frequency domain.

Applied LDA

- More interested in separability of classes
- 11 features obtained

ML Pipeline Implementation

HAPT Dataset

Data Collection

Data Extraction

Data Visualisation

Plotted Graph to visualise data

Data Preprocessing

Performed Feature Engineering using LDA / Resampling

Models

Models

- SVM
- MLP
- KNN
- K - Means Clustering
- Ensemble

Support Vector Machine

Kernels Provided by Scikit

- Linear Function (Selected)
- Polynomial Function
- Radial Basis Function (Selected)
- Sigmoid Function

Polynomial/Sigmoid function
- Computationally expensive
- Suited for NLP problems

SVM using Linear Kernel

- Knowledge of linearly separable data

Initial accuracy = 94.5%
Accuracy after LDA = 97.2%

SVM using RBF

- No prior knowledge of dataset
- More computationally expensive

Initial accuracy = 94.8%
Accuracy after LDA = 97.2%

Hyperparameters

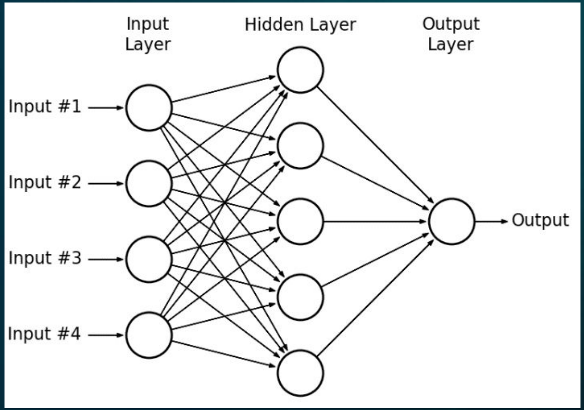
- C - error margin
- Gamma - how much curvature we want in our decision boundary

Multi Layer Perceptron (MLP)

Activation Function: Relu
Optimizer: Adam's optimizer
Loss function: Softmax

Hyperparameters tuned:

- Dropout
- Learning rate
- Epochs
- Batch size



Src: https://www.researchgate.net/figure/A-hypothetical-example-of-Multilayer-Perceptron-Network_fig4_303875065

K-Nearest Neighbours

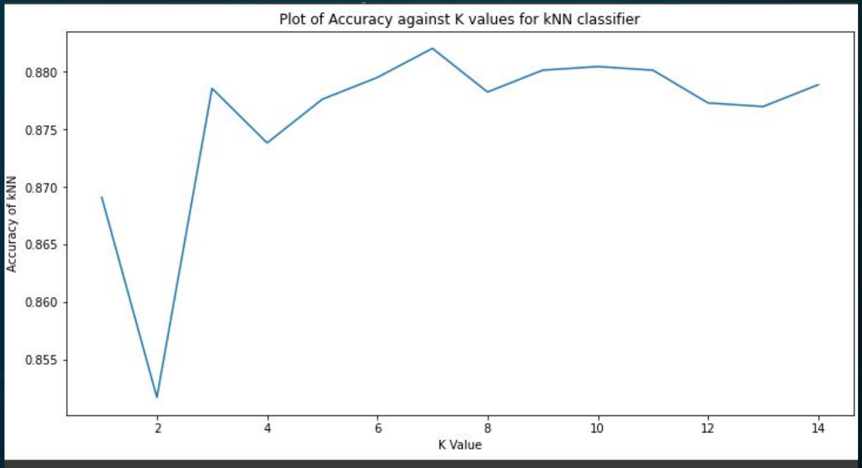
if the sensor readings of each action are sufficiently different, knn will perform well

$$\frac{\text{number of instances}}{\text{number of features}} > 5 \text{ then knn can be used}$$

7767 / 561 = 13.8 > 5

Elbow method gives k = 7

Distance metric: Euclidean

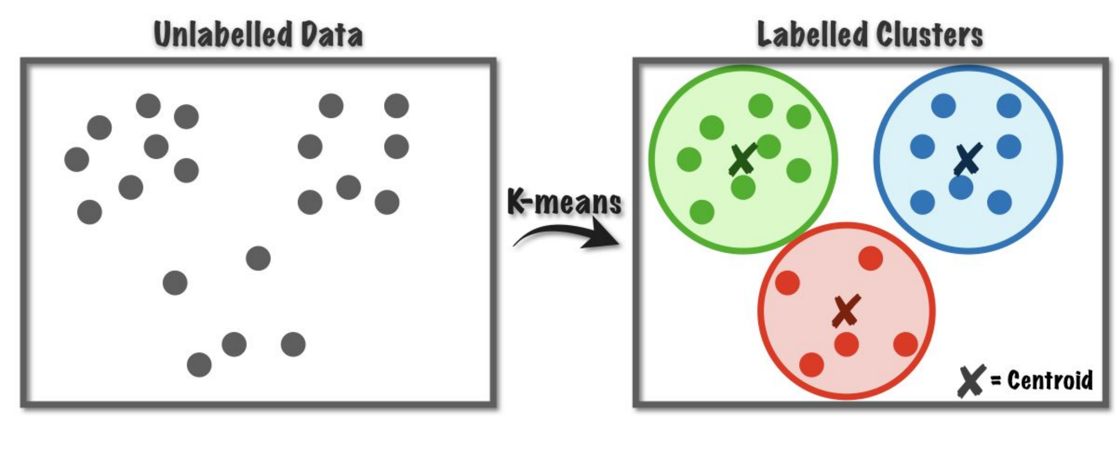


Initial accuracy = 88.2%
Accuracy after LDA = 97.3%

K-Means Clustering

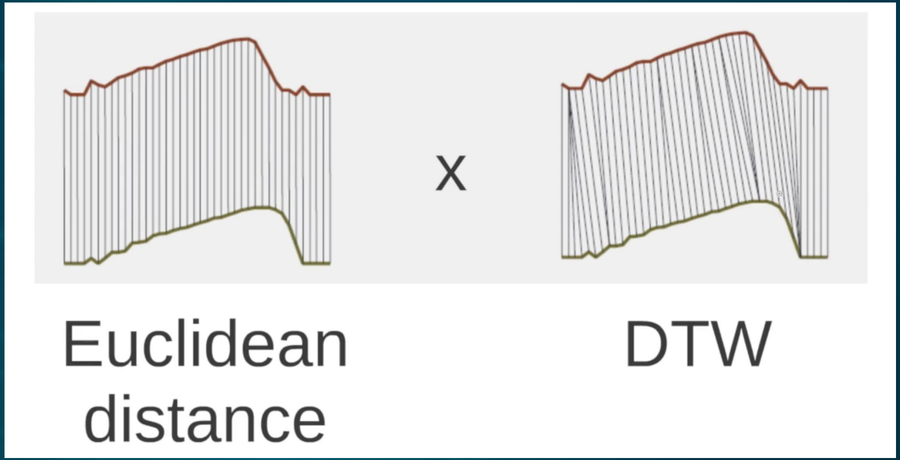
- Unsupervised learning algorithm for classification?
- Added as a part of Feature engineering and then given to a Supervised Classification Algorithm
- Used Euclidean distance and Logistic Regression as classifier

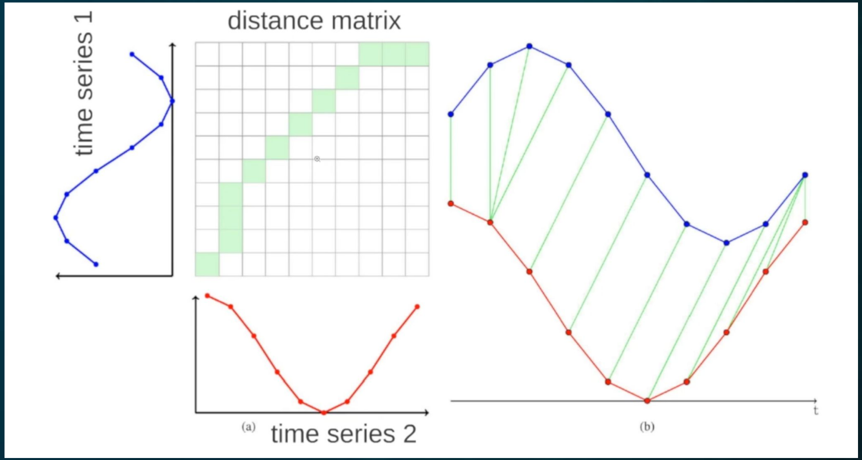
Can we do better than Euclidean distance?

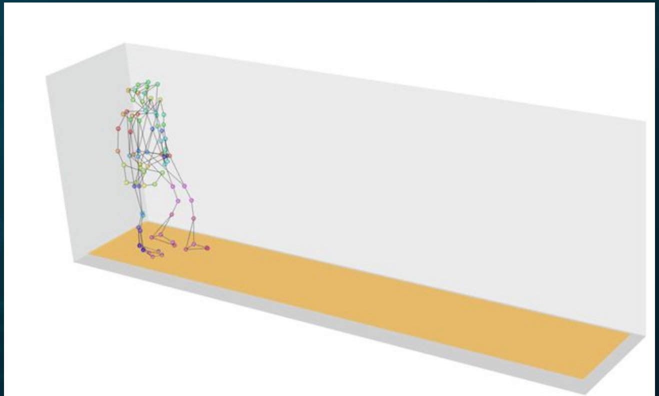


Initial accuracy = 93.3%
Accuracy after DTW = 95.7%

Dynamic Time Warping (Distance Metric)

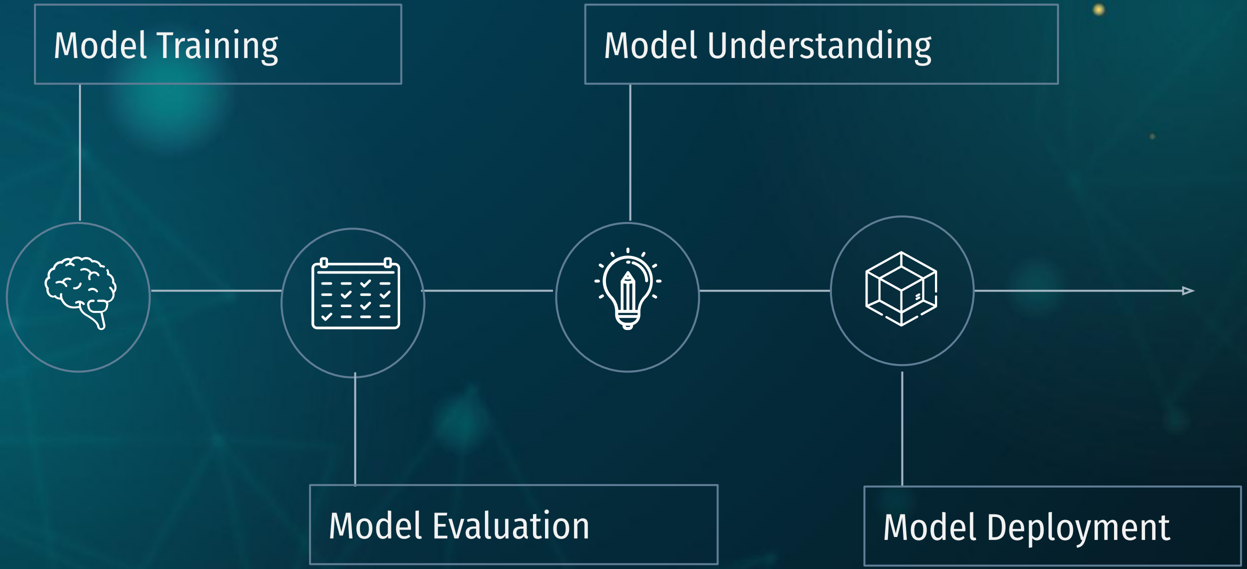






Evaluation (Dian Hao)

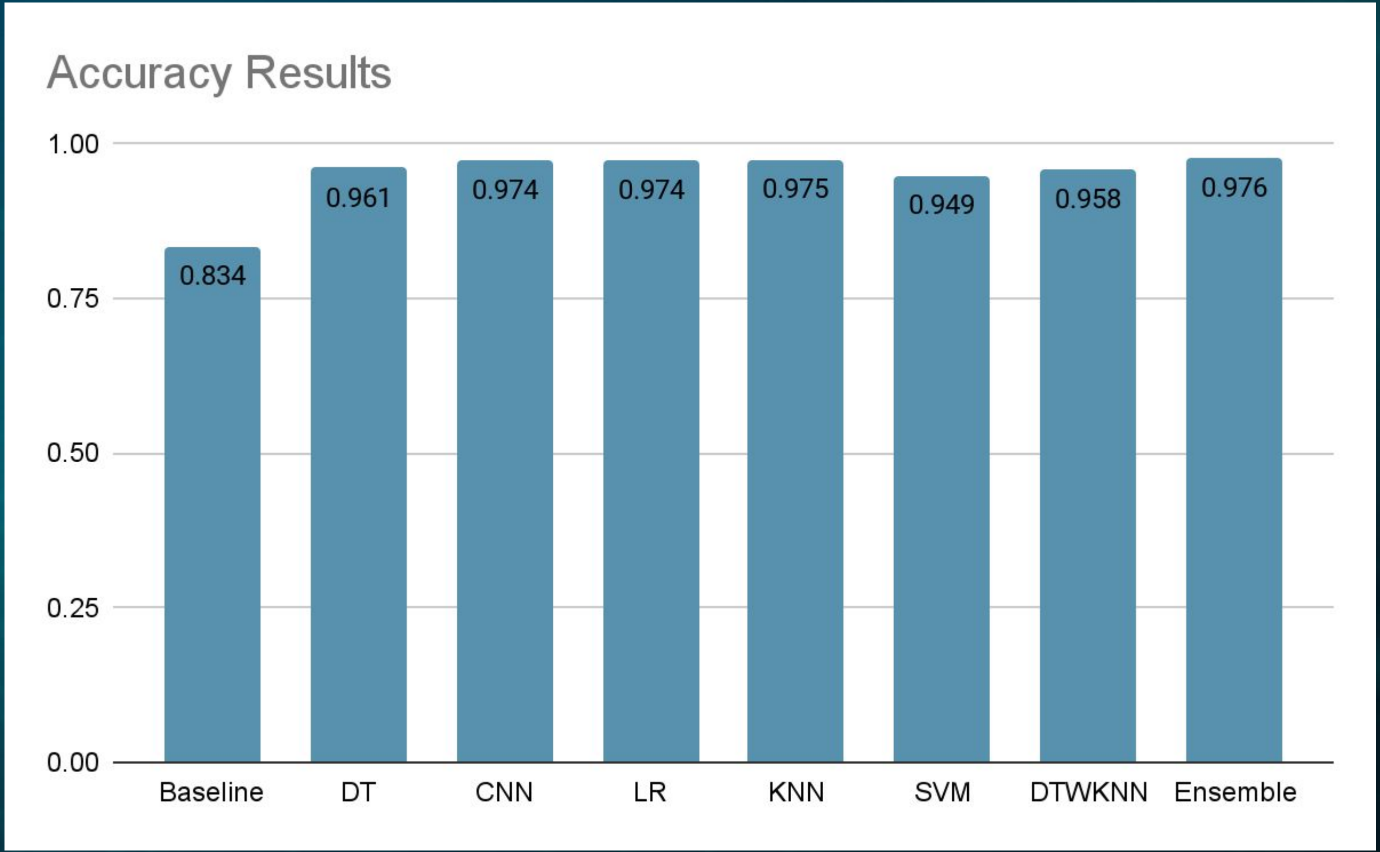
ML Pipeline Implementation



Model Evaluation Metrics

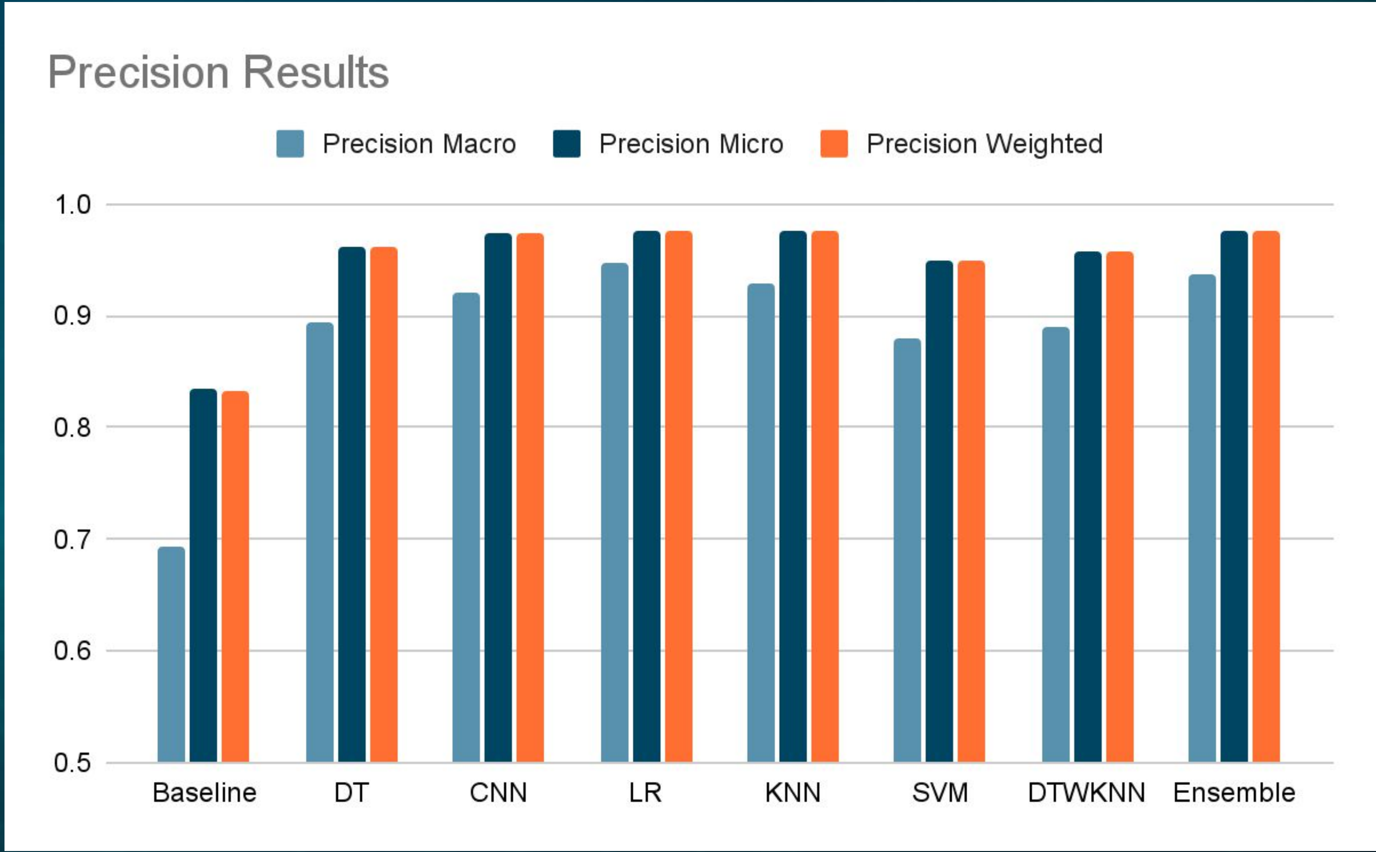
- Accuracy
- F1 macro
- F1 micro
- F1 weighted
- Precision macro
- Precision micro
- Precision weighted
- Recall macro
- Recall micro
- Recall weighted

Accuracy

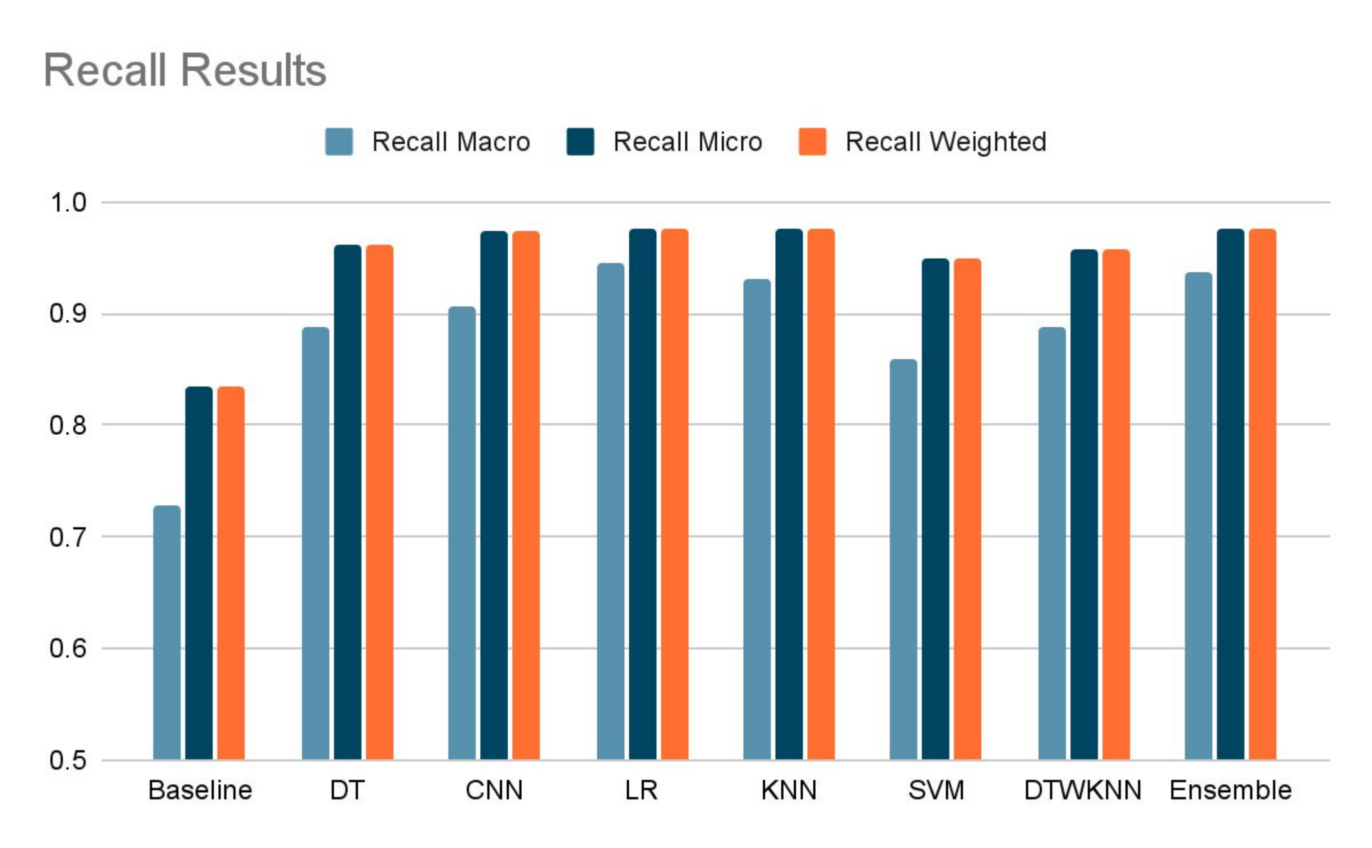


Accuracy Results
1.00
0.75
0.50
0.25
0.00
0.834
0.961
0.974
0.974
0.975
0.949
0.958
0.976
Baseline
DT
CNN
LR
KNN
SVM
DTWKNN
Ensemble

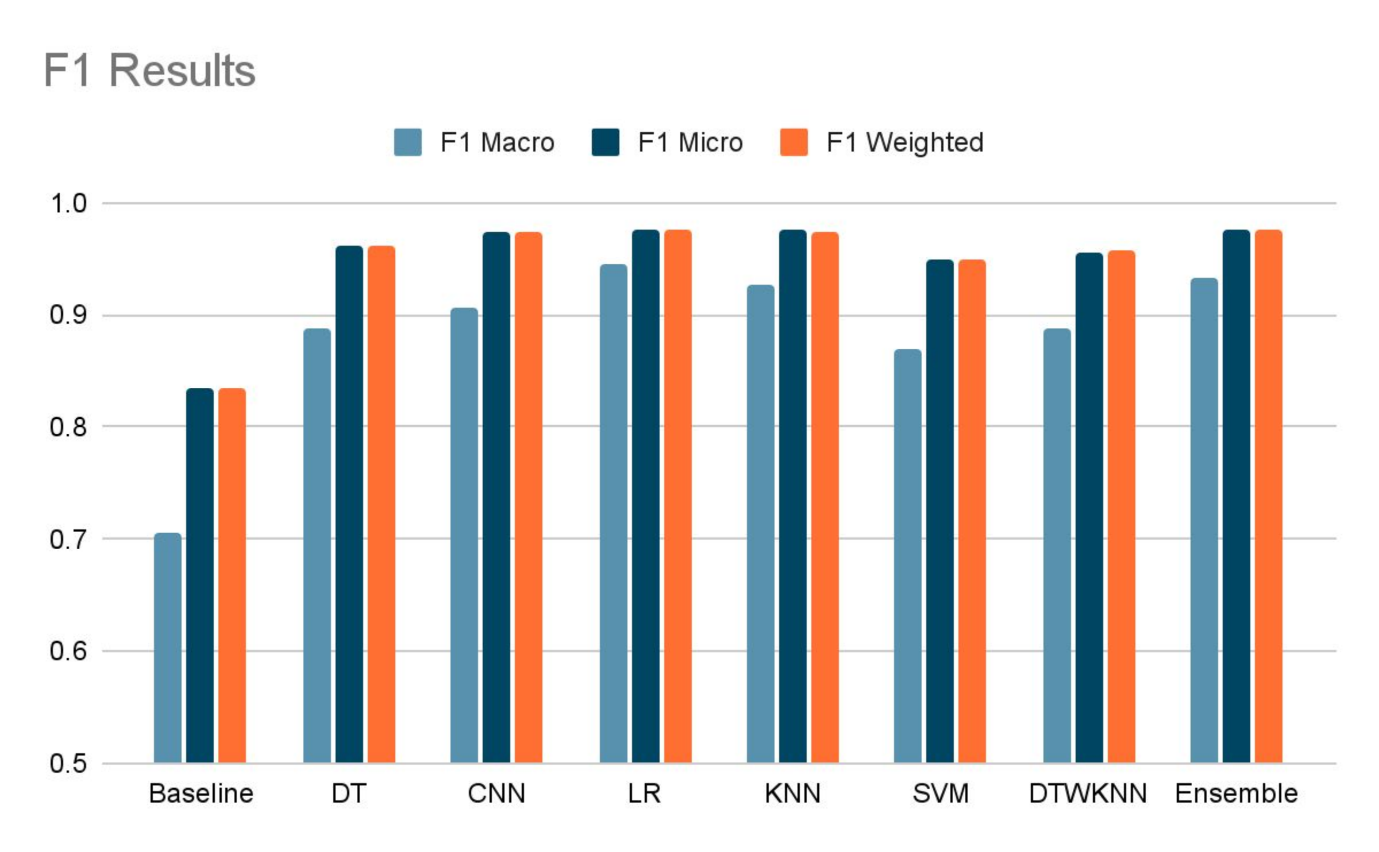
Precision


Precision Results
Precision Macro
Precision Micro
Precision Weighted
1.0
0.9
0.8
0.7
0.6
0.5
Baseline
DT
CNN
LR
KNN
SVM
DTWKNN
Ensemble

Recall


Recall Results
Recall Macro
Recall Micro
Recall Weighted
1.0
0.9
0.8
0.7
0.6
0.5
Baseline
DT
CNN
LR
KNN
SVM
DTWKNN
Ensemble

F1 Score


F1 Results
F1 Macro
F1 Micro
F1 Weighted
1.0
0.9
0.8
0.7
0.6
0.5
Baseline
DT
CNN
LR
KNN
SVM
DTWKNN
Ensemble

Evaluation: Observations

- A baseline model before feature extraction and hyperparameter tuning is suboptimal.
- Metrics with macro-average tend to have lower scores than micro-average counterparts.
- Several models have the same accuracy, F1 micro, precision micro, and recall micro.
- Ensemble model has the best performance out of all models.
- All the models showed good performance and improvements after hyperparameter tuning.

General Analysis

- Macro average does not perform as well as micro average due to imbalanced dataset.
- LDA maps the dataset to be more linearly separable, causing identical accuracies in different models.
- Further analysis from different models' classification reports shows that the class "SIT_TO_LIE", "LIE_TO_STAND" is the most challenging in predicting.

Results interpretation

LDA works well on our dataset

Maximizing distance
Between groups

LDA

Maximizing distance
Between groups

561 to 11 features

kNN

Accuracy:
0.975

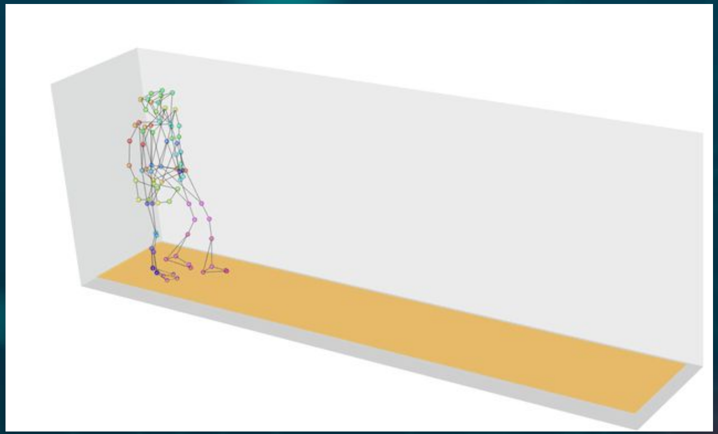
Dynamic Time Warping doesn’t work well with LDA

LDA reduces from
561 to 11 features

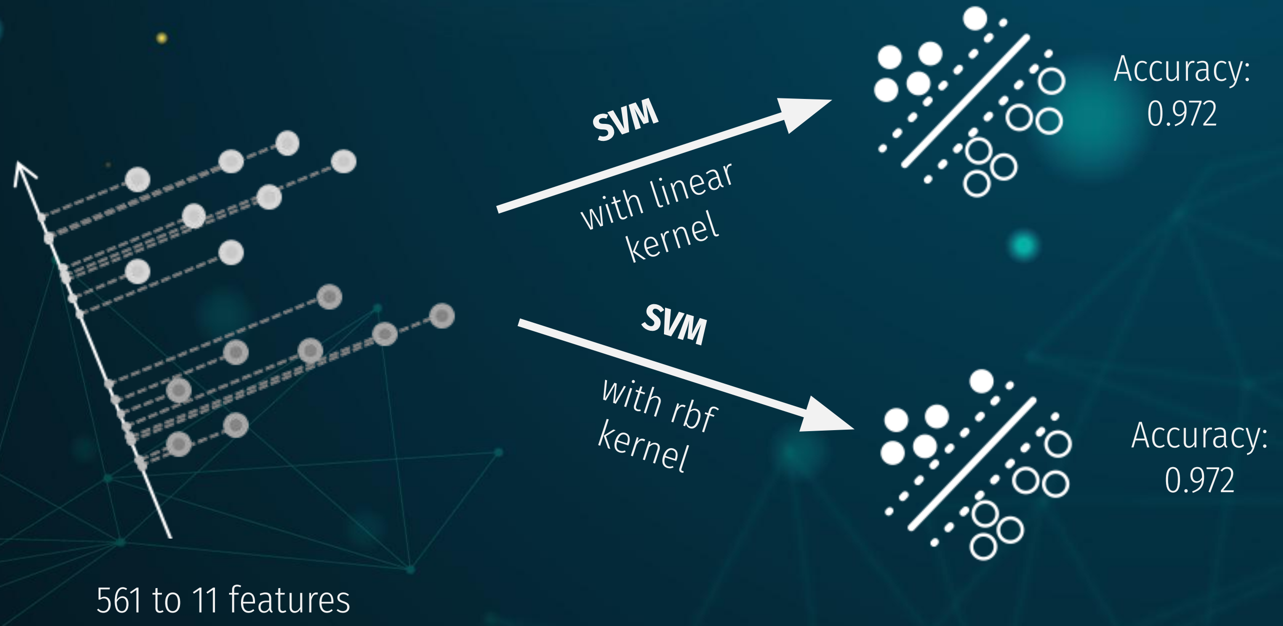
kNN

With DTW

Accuracy:
0.958



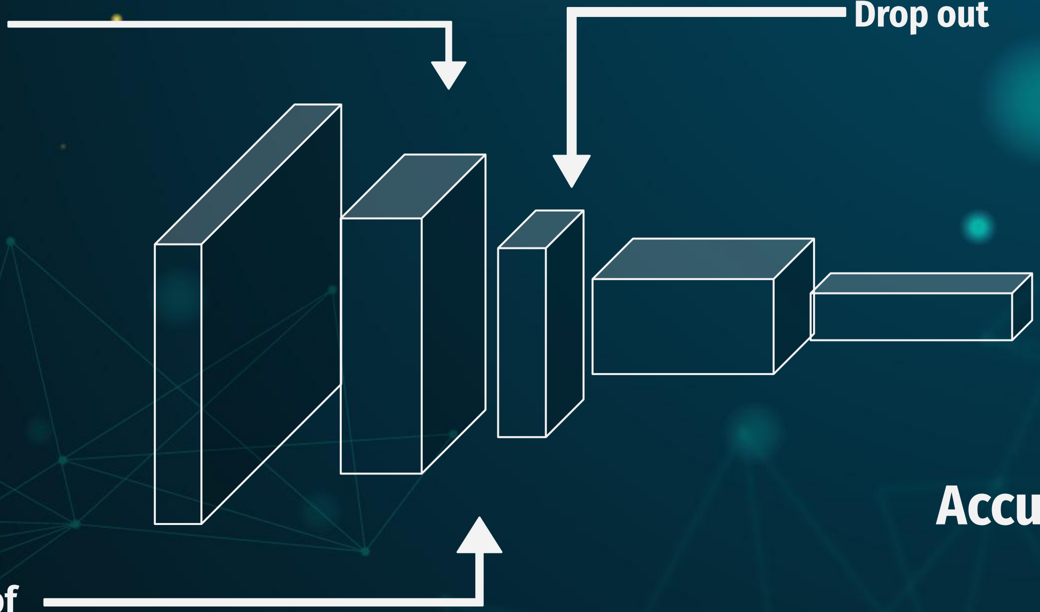
LDA vastly improved linear separability



CNN shows generalization works better

Small batches

Drop out



Low number of epochs

Accuracy: 0.974

L1 reduces overfitting for Logistic Regression

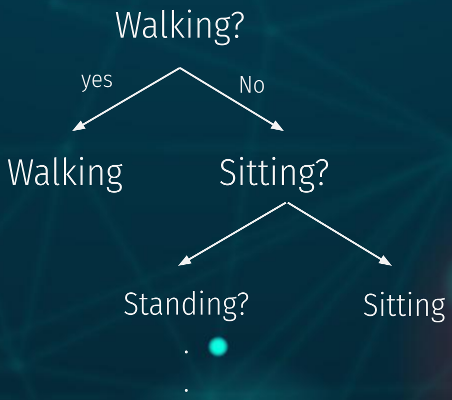
L1 distance metric

- L2 loss penalizes outliers heavily
- L1 loss is better as it pulls weights to 0 to reduce overfitting

Accuracy: 0.974

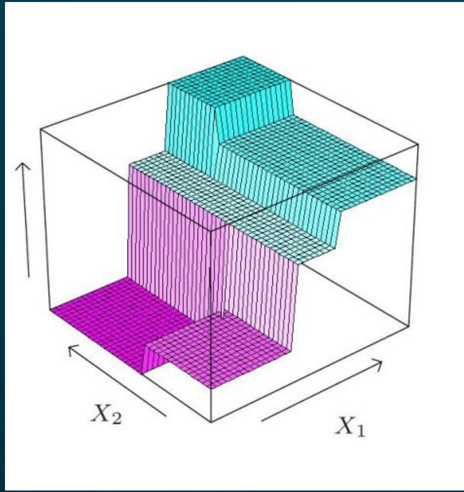
Future: Multilayer Logistic Regression

- Logistic regression is for binary classification
- Multilayer logistic regression might work better



LDA produced good decision boundaries

Decision Tree
Accuracy: 96.1%



Good axis
perpendicular
decision boundaries

Ensemble of models with Majority Voting

Logistic Regression

kNN

SVM

Random Forest

Ensemble
Accuracy: 97.6%

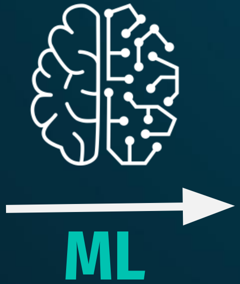
Some issues

- Ensemble accuracy is **not much** better than composite models
- Most models are predicting the **same** instances wrongly
- Shows **high correlation** among predictions
- LR and kNN have same accuracies which imply that they could possibly contribute to **double voting**
- Future improvements: correlation could be studied further and removed

Never stop improving

- Explore ways to speed up Dynamic Time Warping with kNN
- Explore multilayer logistic regression approach
- Explore other models such as LSTM which is good for time series data
- Improve ensemble model by including other models and removing correlations
- SMOTE/oversampling could be explored to handle imbalance dataset
- Additional dataset/data collection could be done to improve and validate our model

Remote Physiotherapy may be possible



THANKS!