



SUPERVISED LEARNING
CLASSIFICATION PREDICTION FOR

PHYSICAL PERFORMANCE LEVEL

Dec 19, 2021
9:00 a.m.

Bushra AIOLit,Laila
Almjnuni,Shoug
alkhathran

Dr.Mejdal ALQahtani



AGENDA

- Introduction
- Methodology
- Data
- Model selection
- Conclusion Q & A

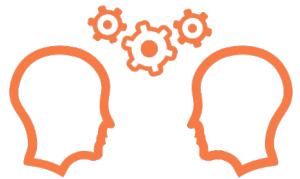
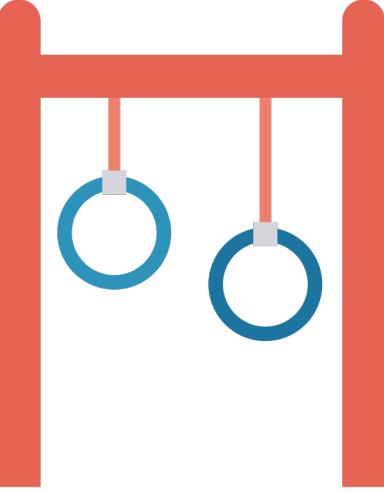


INTRODUCTION

Physical performance of our body is the most important thing to improve and give it a time to grow based on moves and jumping and to help our coaches find best class for any client we creates & improve these models to be an Easy App



Data Science Methodology



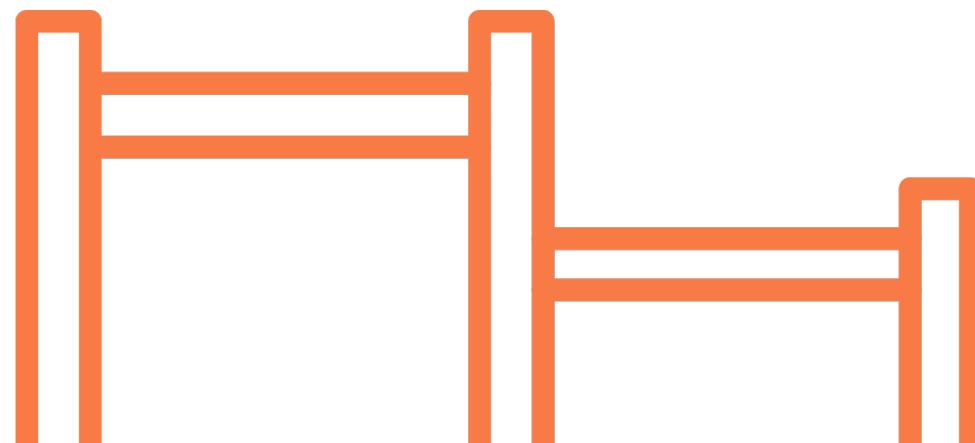
**Identifying the problem and the approach
to fix the problem**



Data requirements and collection methods



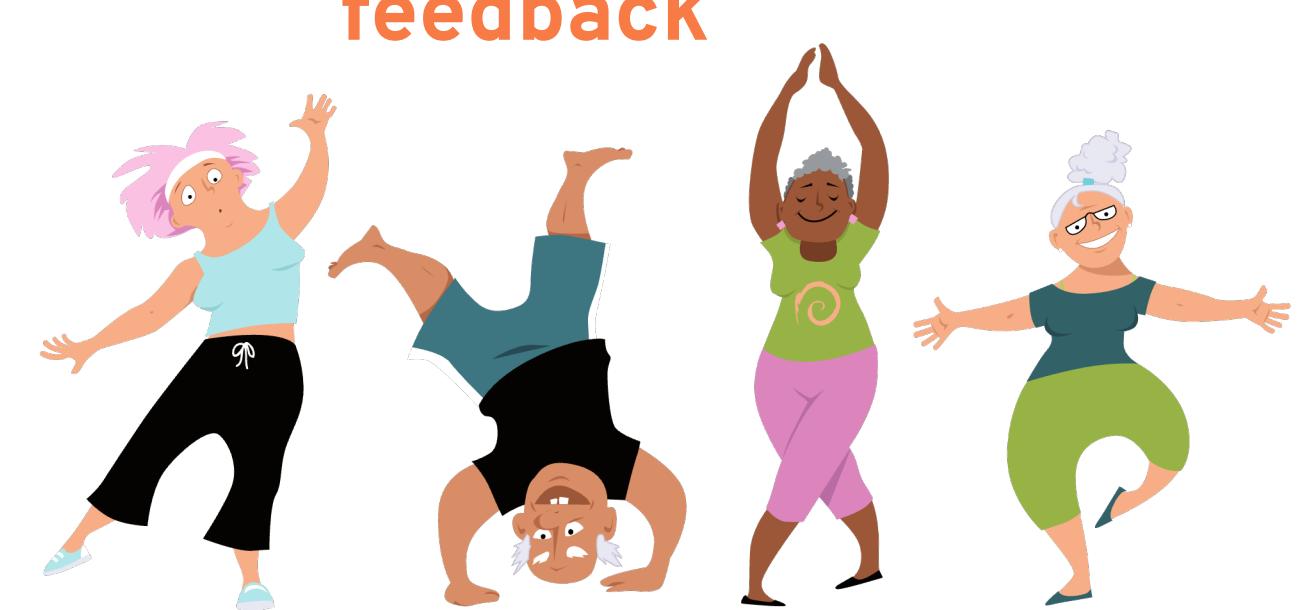
Understand the data



**Generate models and
evaluate them**



**Deploy the model and get
feedback**



DATA



Understanding and Preparing the Data

is this data going to answer problem that I am having?

+10,000 rows

13 Columns\Features

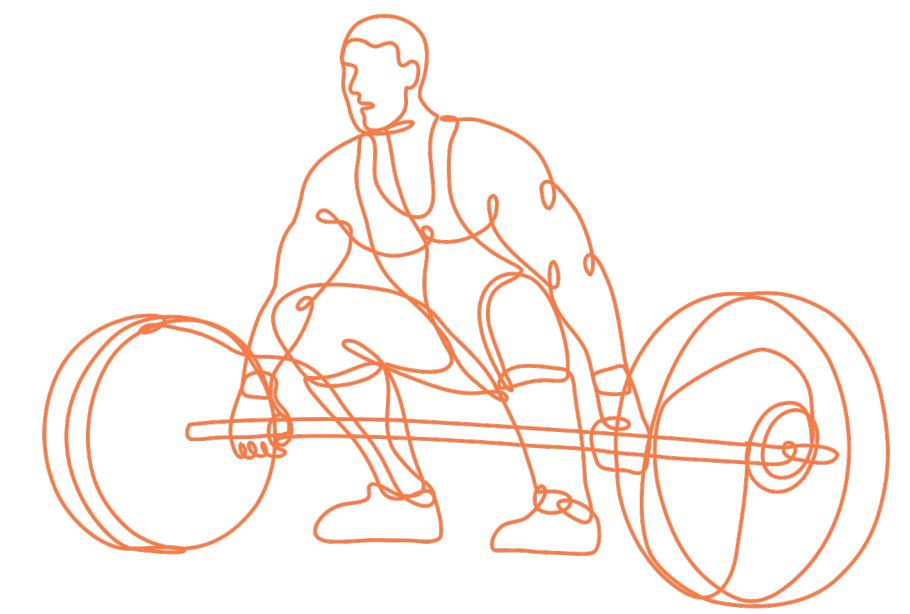
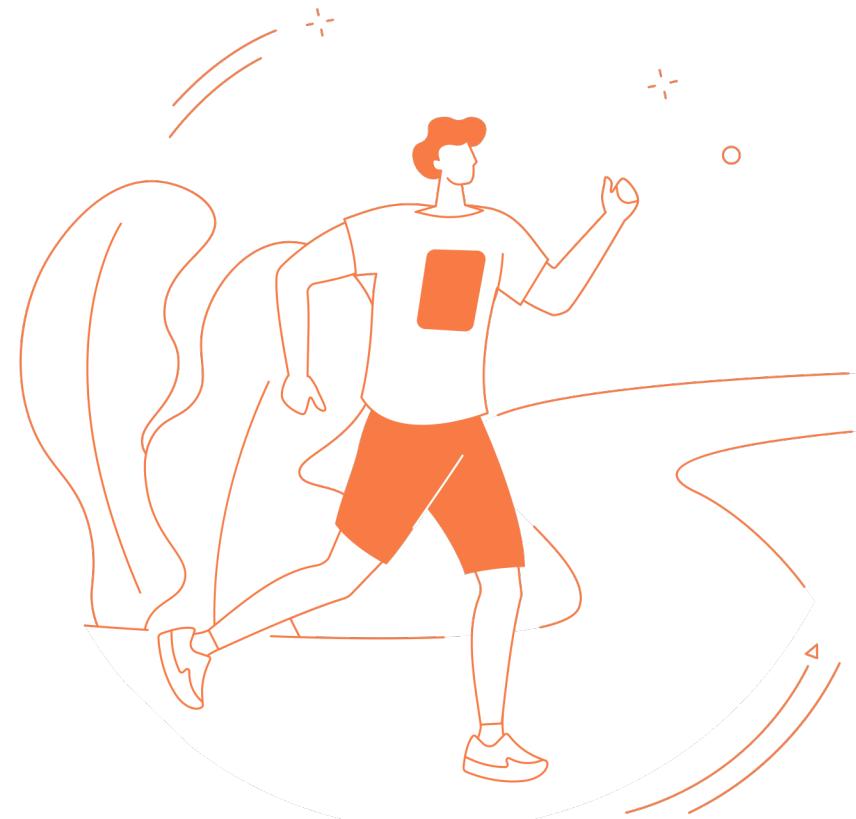
-adding BMI

BMI give

weight_kg/height_cm

-Other 12 features

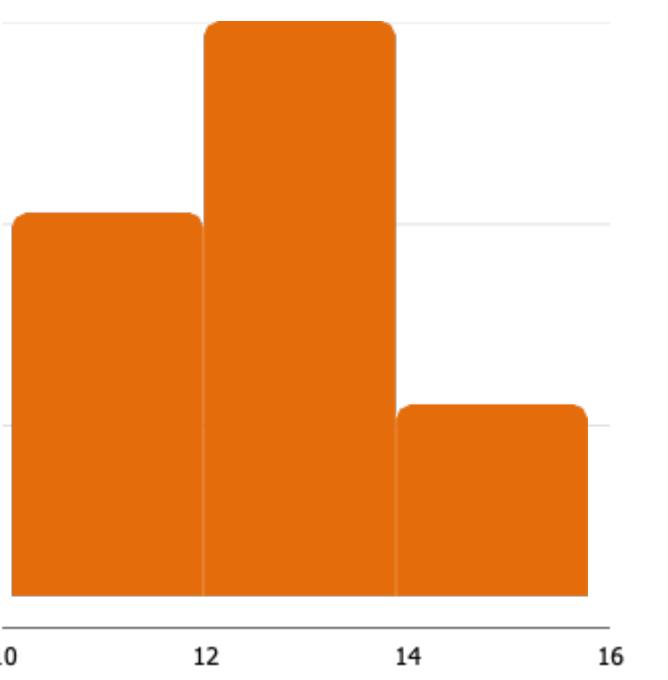
Target is called [Class]



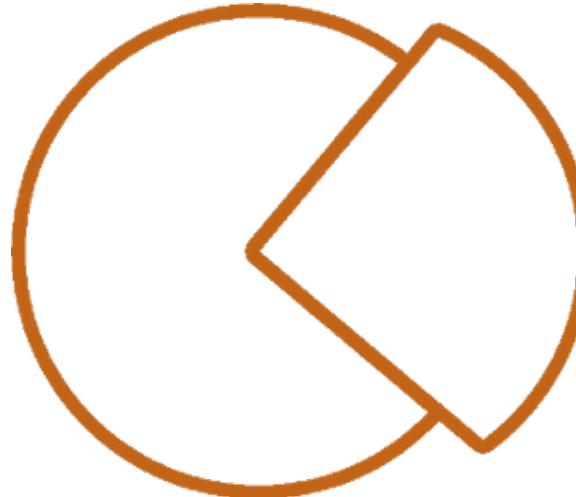
Tools



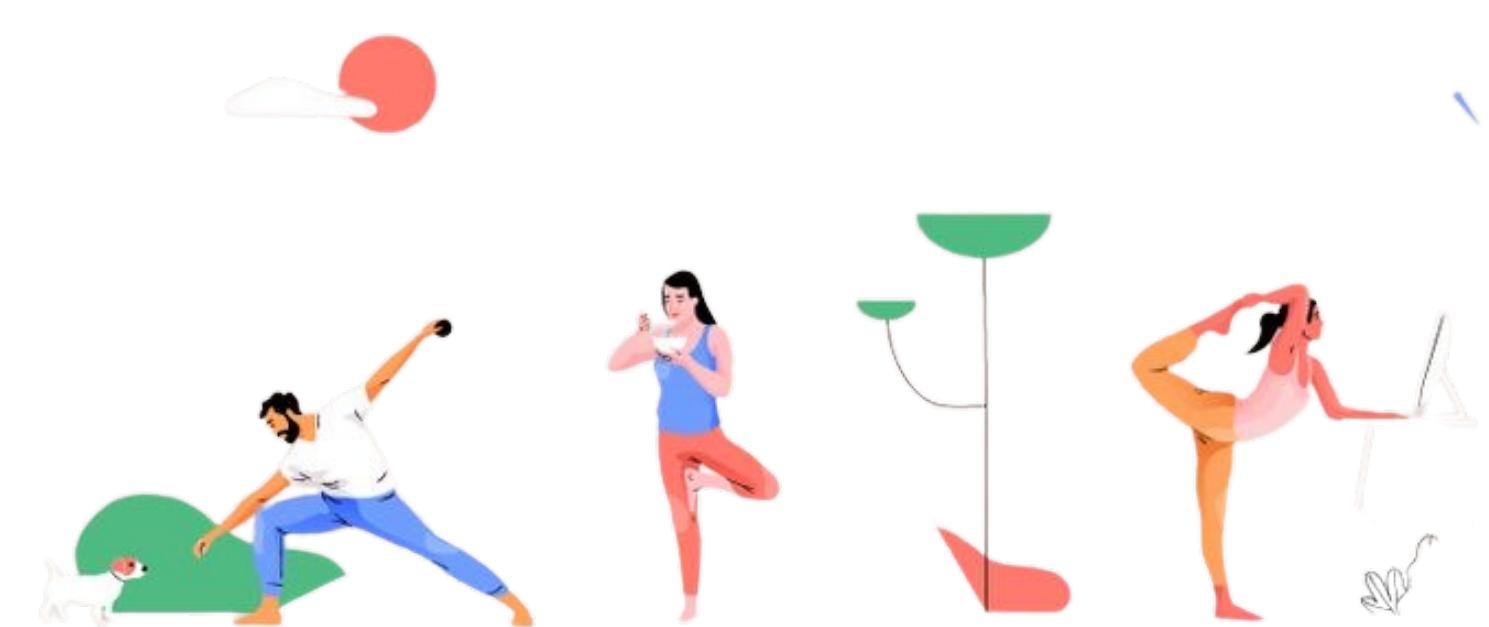
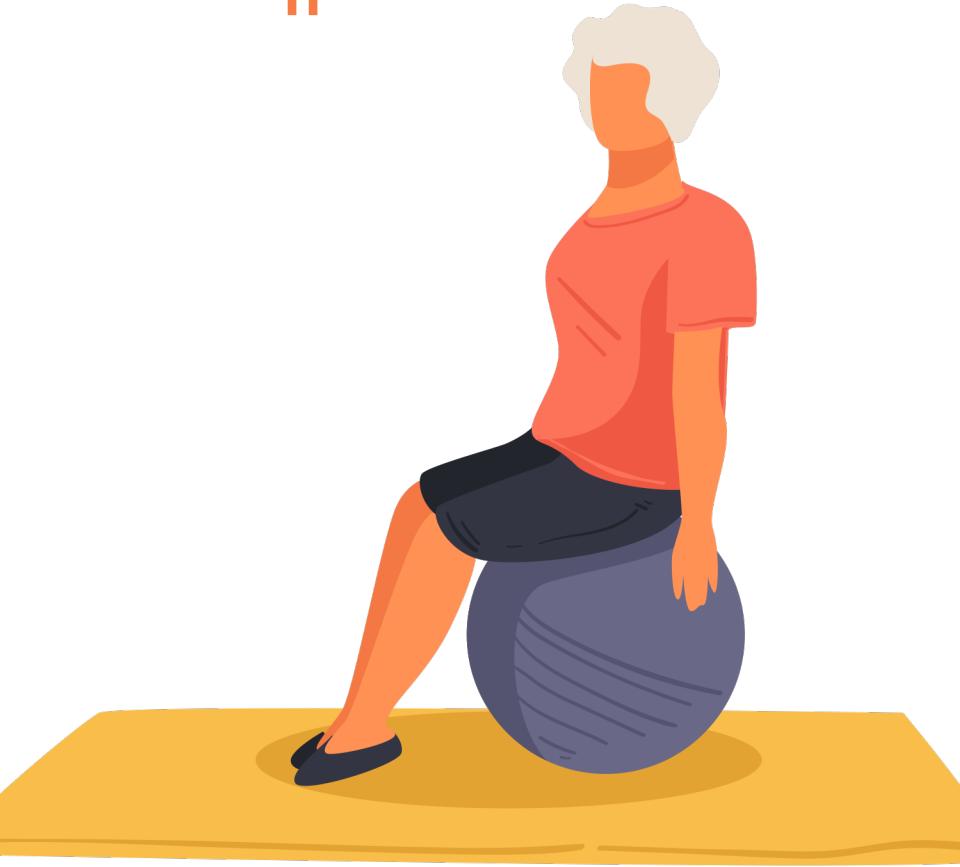
Pandas,
Numpy



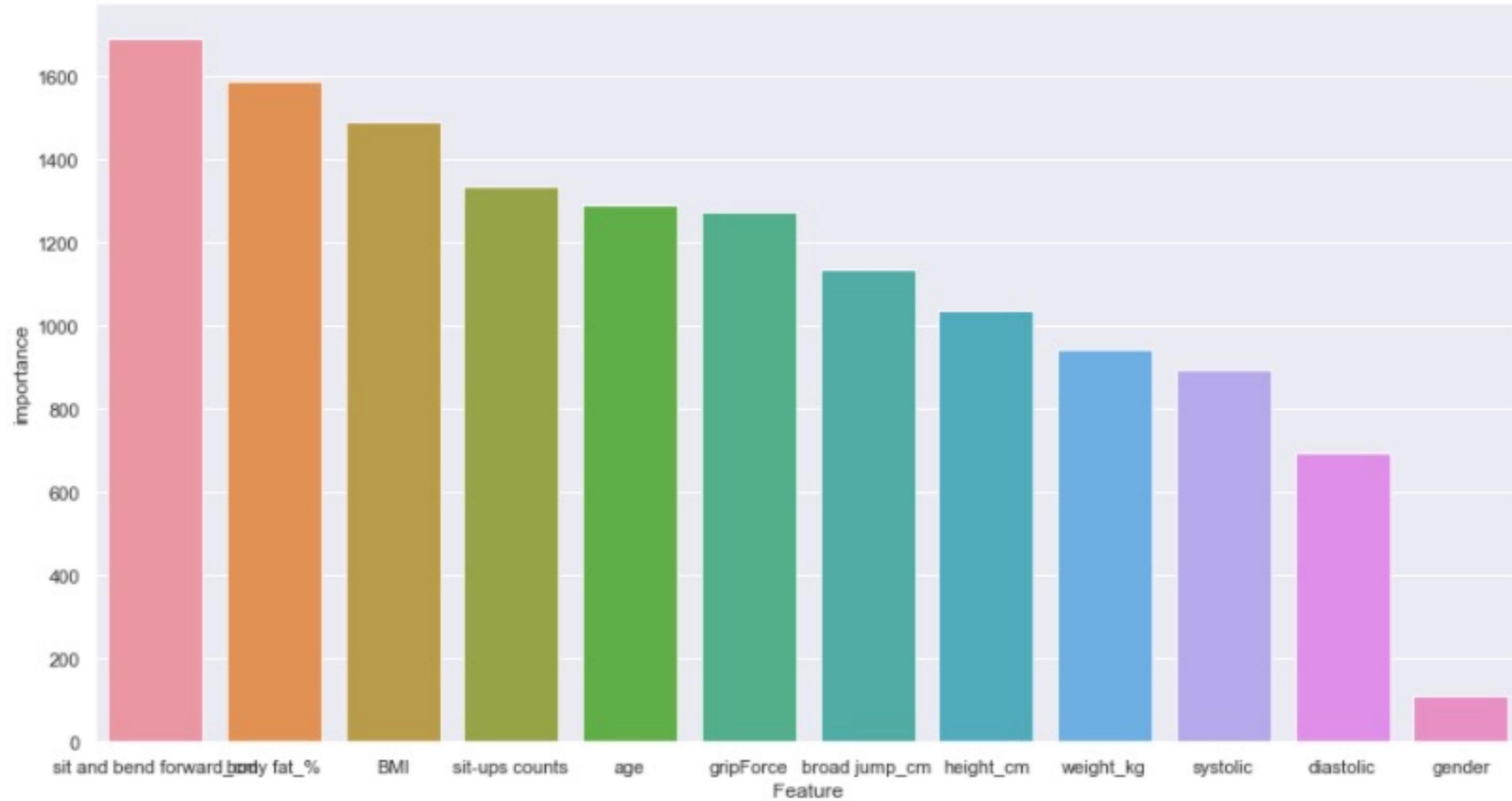
Seaborn,Matplotlib
b



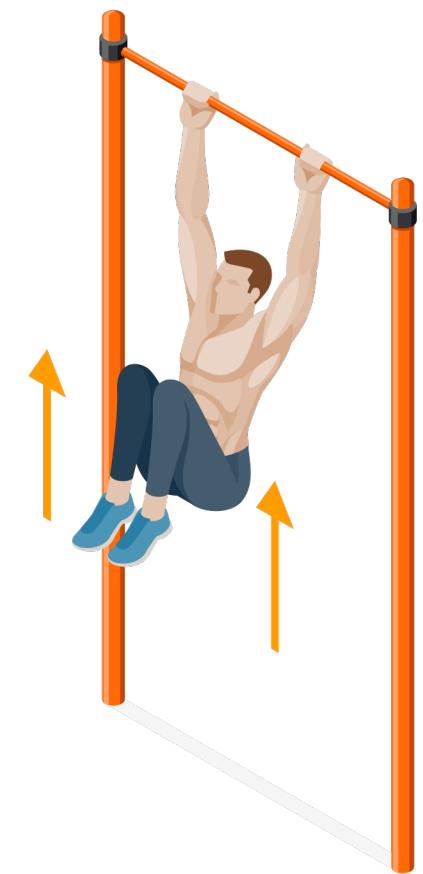
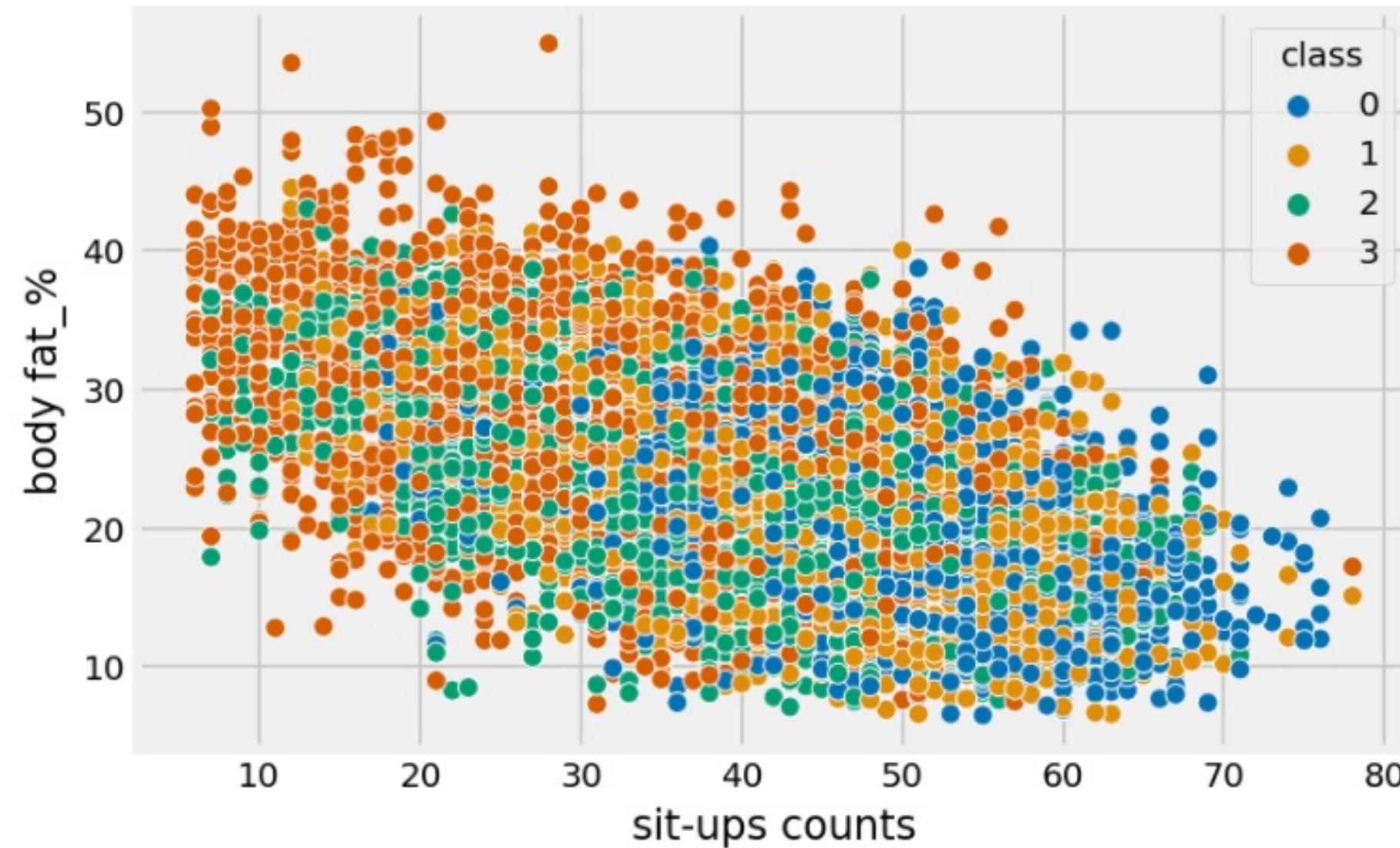
Sklear
n



Important Features

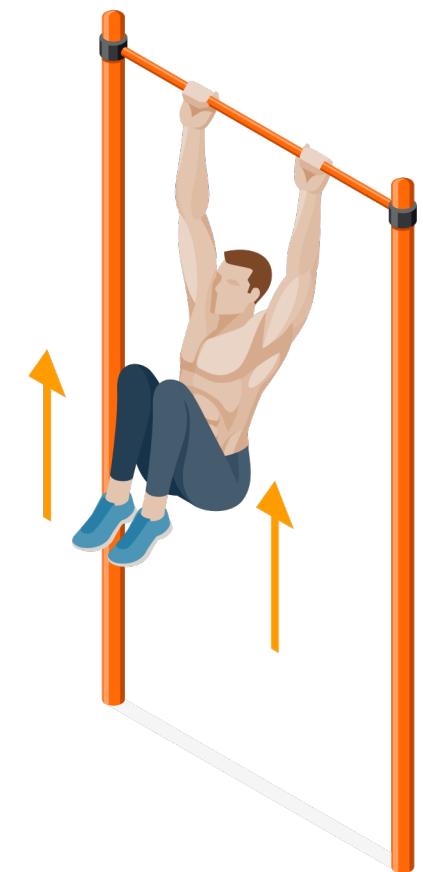
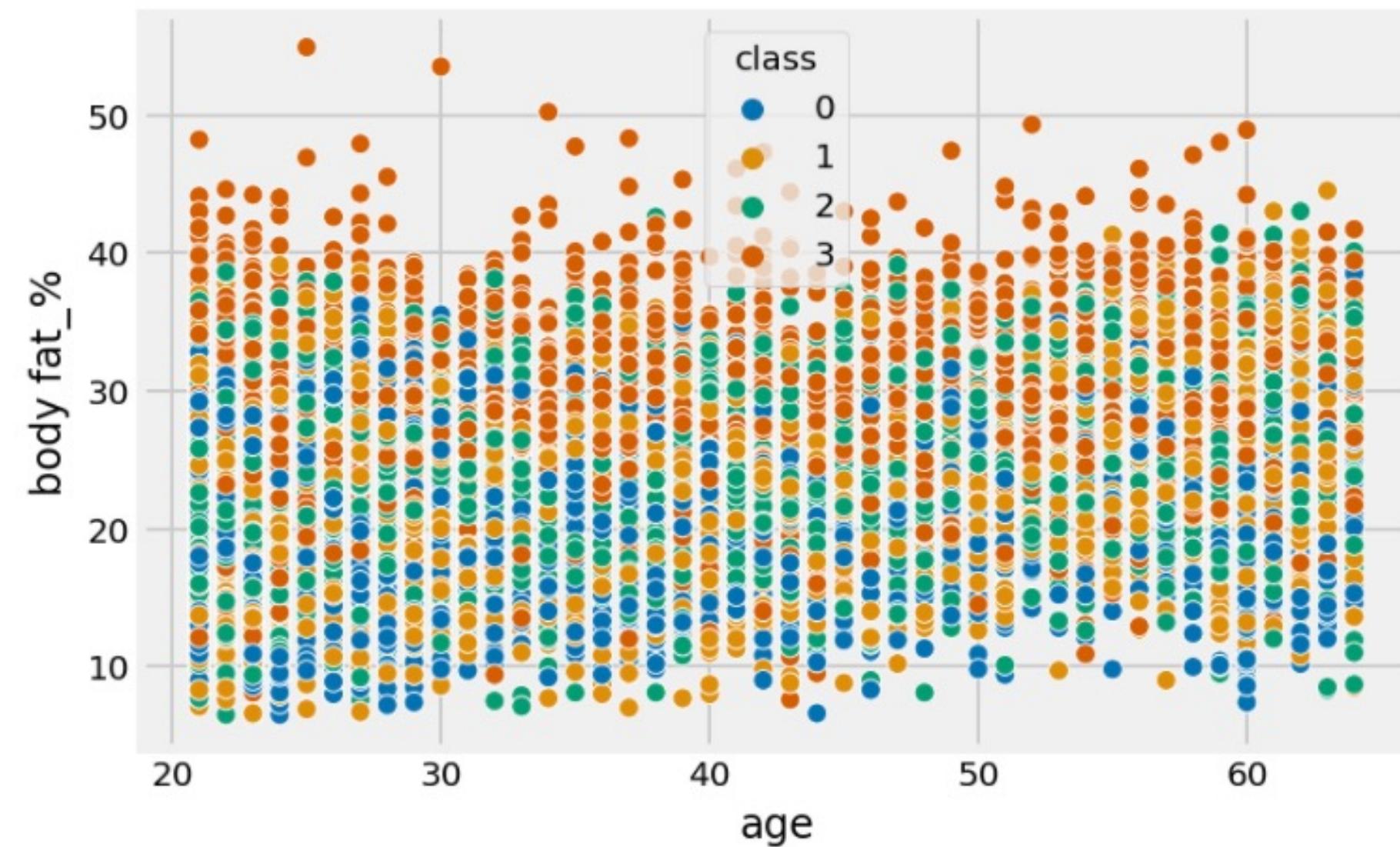


PairPlot



PairPlot (Selected
Features)

PairPlot



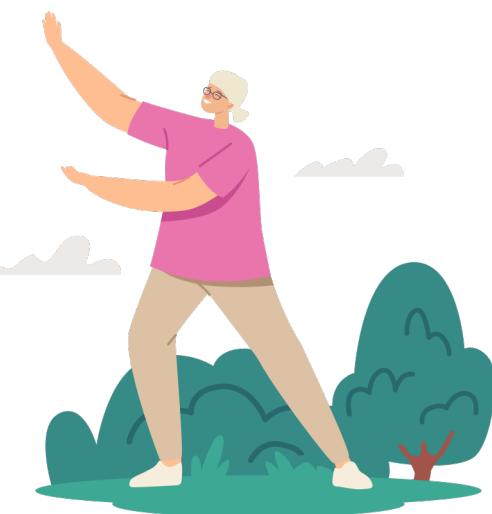
PairPlot (Selected
Features)

Model Experiments



Experiments Name	
0	LogisiticRegression
1	KNN
2	DecisionTree
3	RandomForest
4	Xgboost
5	SVM
6	MultinomialNB

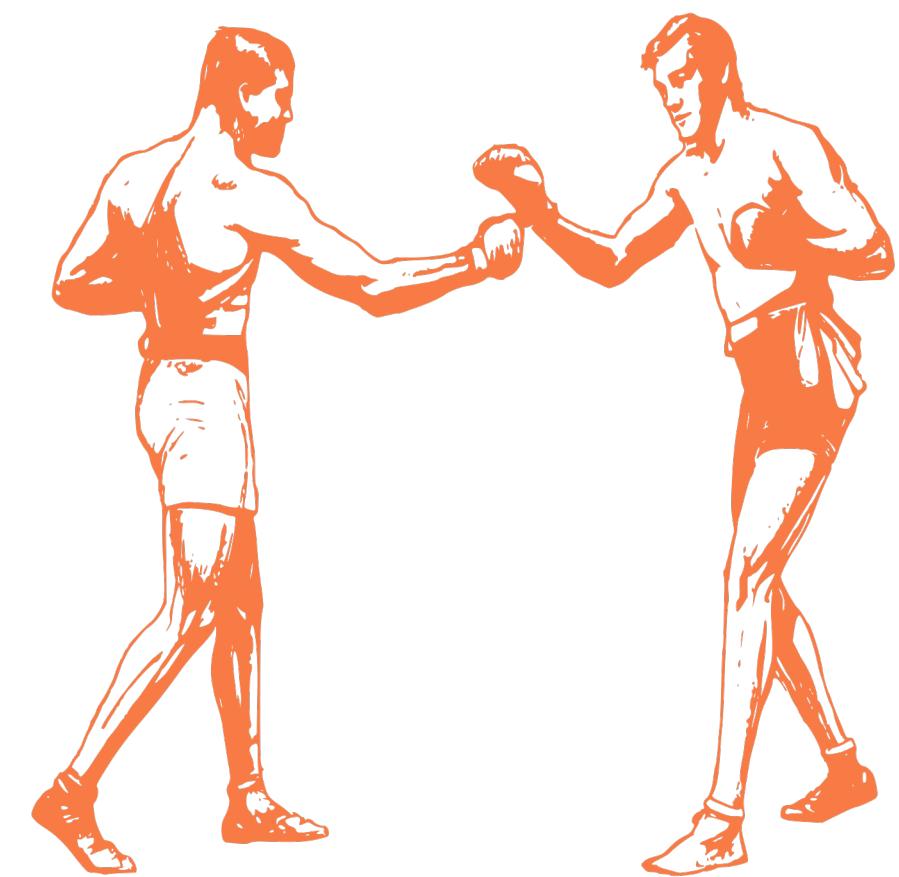
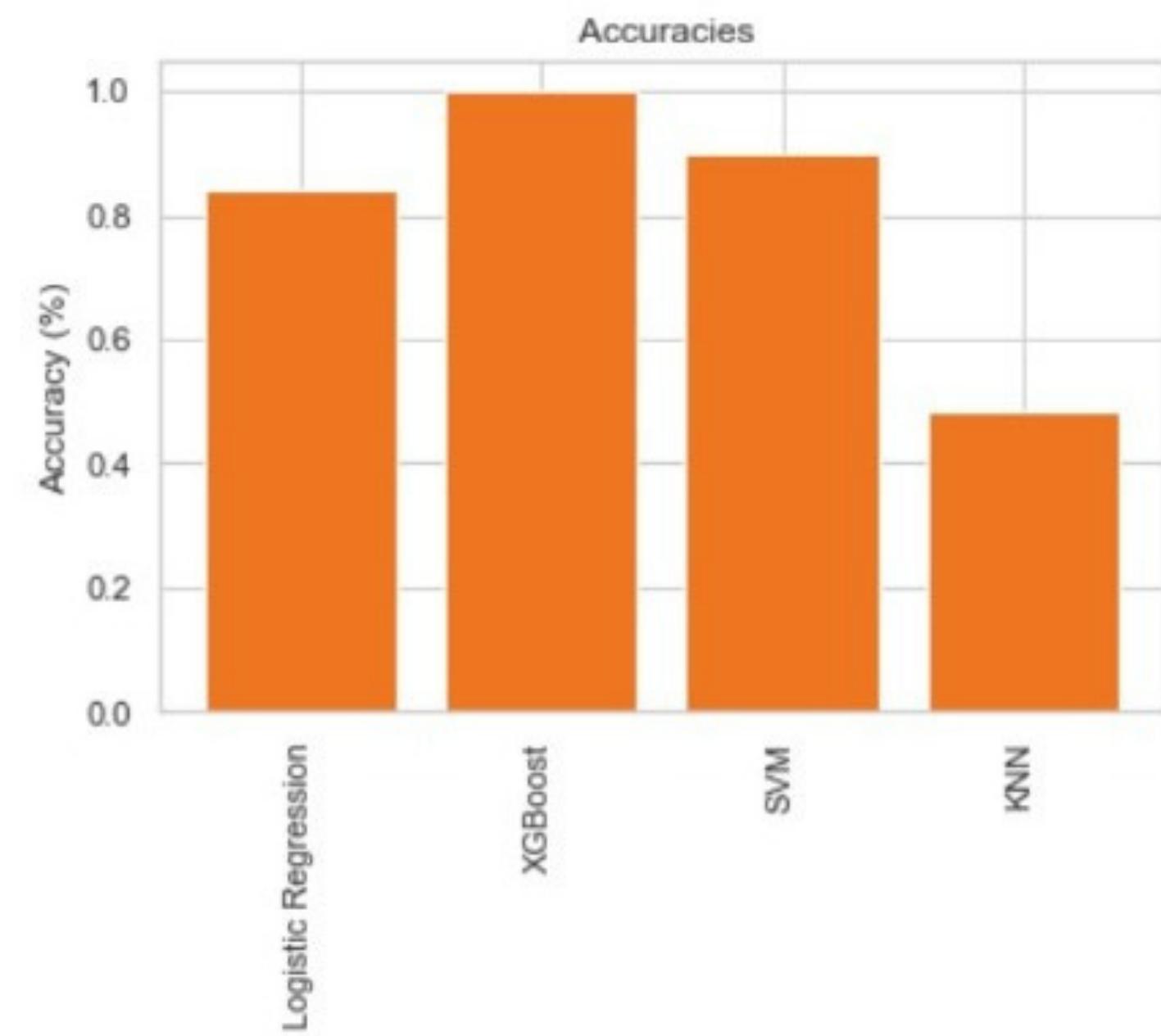
The worst model



	Model Name	Accuracy score	MSE Before	After tuning	MSE After
3	XGBoost	0.81103	0.242759	0.81517	0.244828
1	Random Forest	0.80621	0.260000	0.80483	0.255172
4	SVM	0.76207	0.289655	0.78207	0.265517
5	Decision Tree	0.73862	0.386897	0.76345	0.346897
6	MultinomialNB	0.73862	0.386897	0.76345	0.346897
0	Logistic Regression	0.72966	0.351034	0.73655	0.321379
2	KNN	0.70690	0.424828	0.71586	0.386897



Compare Models

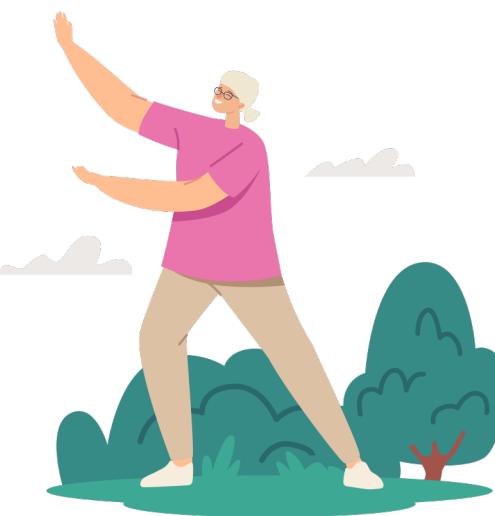


Model Selection



Model Name	KNN	F1	Precision	Recall	Accuracy
Class A	0.72	0.88	0.81	0.72	
Class B	0.51	0.49	0.50		
Class C	0.61	0.60	0.61		
Class D	0.98	0.83	0.90		

Model Selection



Model Name	RandomForest	F1	Precision	Recall	Accuracy
		0.84	0.90	0.87	0.80
Class A	0.65	0.63	0.64		
Class C	0.72	0.70	0.71		
Class D	0.97	0.94	0.96		



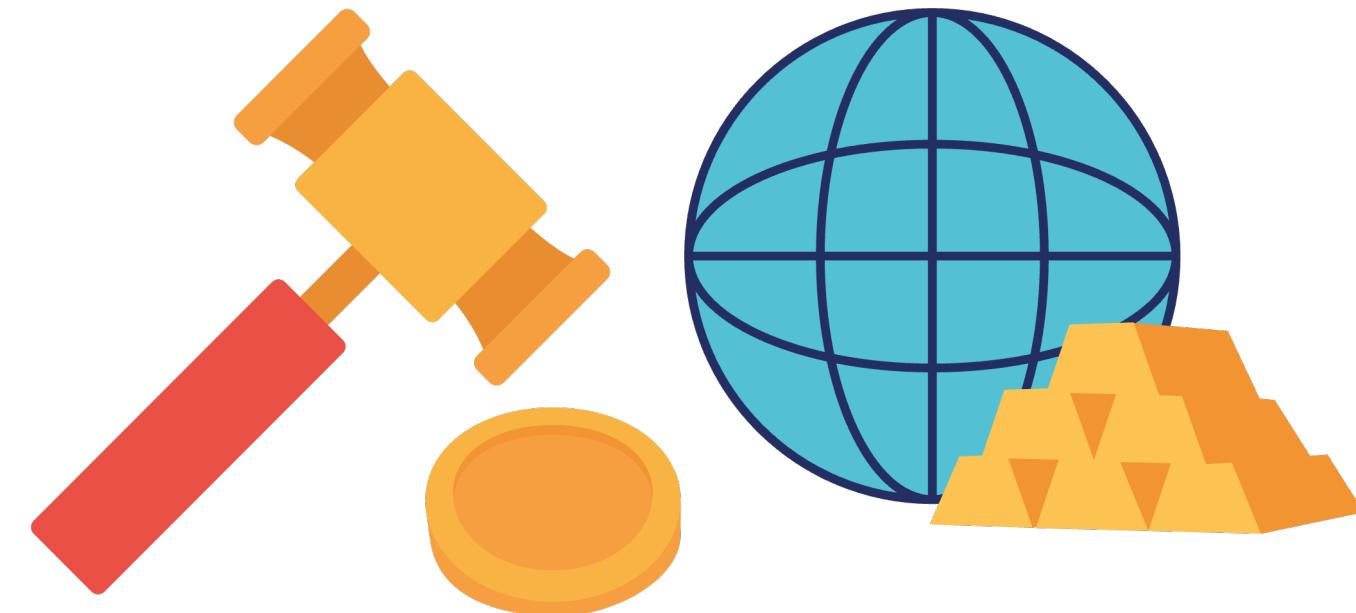
Model Selection



Model Name	F1	Precision	Recall	Accuracy
Xgboost-Tun				
Class A	0.85	0.90	0.88	0.82
Class B	0.66	0.67	0.67	
Class C	0.74	0.70	0.72	
Class D	0.97	0.95	0.96	

Golden Model

XGboost-tunning



Model Name	Xgboost-Tun	Precision	Recall	F1	Accuracy	Weighted avg
Class A	0.89	0.90	0.90	0.90	0.85	0.85
Class B	0.65	0.61	0.63			0.84
Class C	0.72	0.75	0.73			0.84
Class D	0.99	0.99	0.99			0.84



Example Prediction

Prediction

```
In [39]: input_data = (24,0,62,171,20,65,95,20,100,100,200,21.1)
# now input_data is in the form of tuples.
# we need to change it to numpy array
input_data_array = np.asarray(input_data)

# reshape the numpy array as we are predicting for only one instance
input_data_reshape = input_data_array.reshape(1,-1)
prediction = model_xgb_tuned.predict(input_data_reshape)
print(prediction)

['D']
```

Deliverable



01

Classify the
classes
based on BMI
measrments



02

Xgboost Model
is the Best Model
for this problem

03

GridSearch Tunning
take a long time to run

04

KNN is very
fast to excute

Future Plan

01

- What type of Exercise best for**
- Class
 - Personal Final Goals
 - Adding videos

02

- What type of Food**
- Best item
 - Nutrition schedule



Conclusion

Finding the best model



GridSearchCV for Tuning

Random forest is a very powerful model



Thanks for Listening

Any Questions?

