

CARDIOVASCULAR STROKE PREDICTION SYSTEM

PROJECT GUIDE BY :-

MR.ABHAY MANE

MEMBERS:-

ANKIT BHOGE ROHIT GOYAL

ACHARYA SARANG ABHISHEK PANDEY

SHEIKH MOHSIN

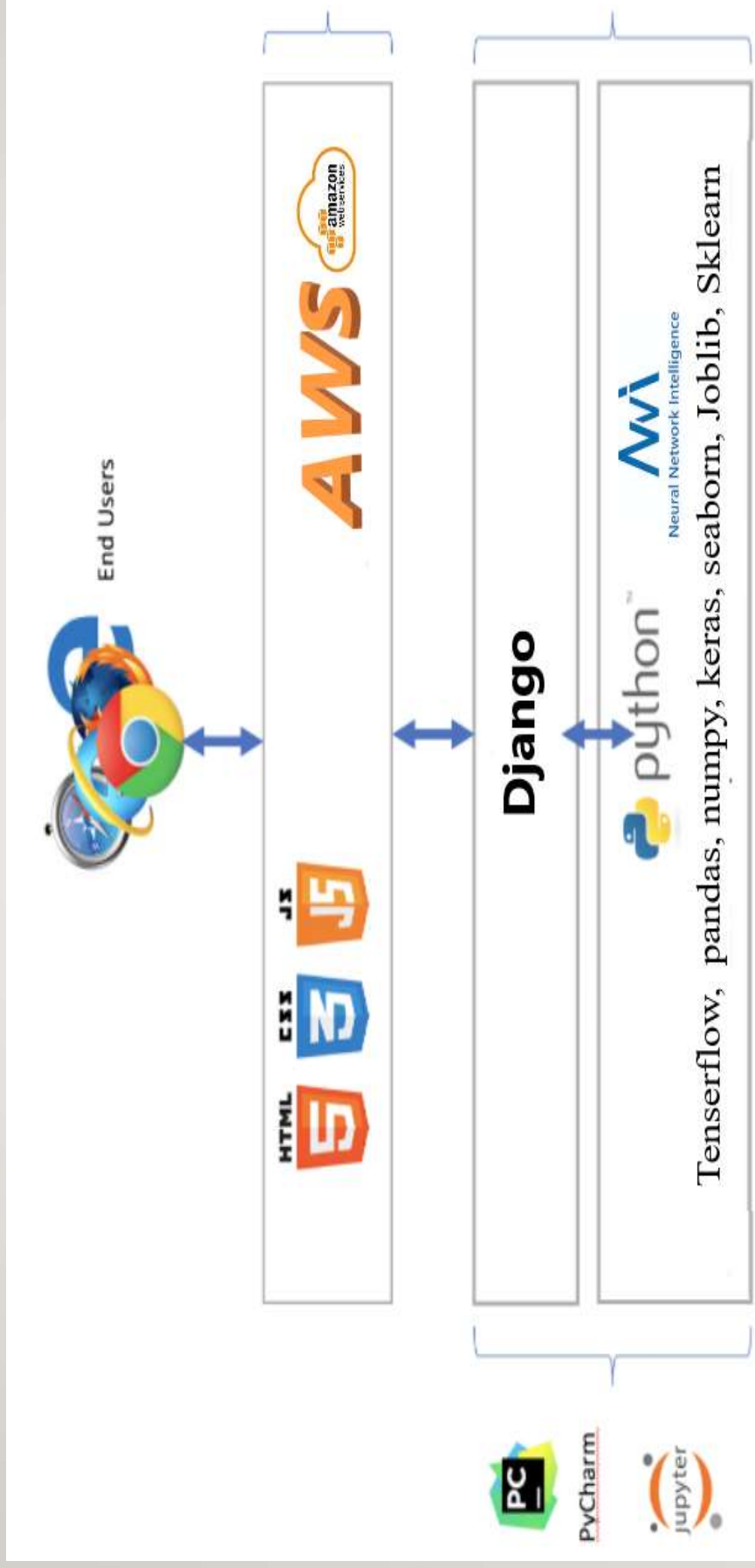


HOW DO WE SOLVE PROBLEM?

- For solving this problem we have used machine learning techniques.
- Using this we are trying to predict the possibility of a person getting cardio stroke
- For prediction we have consider that parameter which we can measure at home
- For making this system user-friendly we have prepared the web interface through which we can predict.



SYSTEM DESIGN



HOW WE BUILD MACHINE LEARNING MODEL?

For building machine learning model.

- Finding out proper dataset which will help us addressing and solving problem better.
- Doing data preprocessing.
- Trying out different machine learning algorithm.
- Hyperparameter tuning.
- Building front-end using Django.
- Deploying using AWS.



FINDING DATASET

- We have found the dataset on kaggle.

Cardiovascular Disease dataset

817

New Notebook

Download (760 kB)

Unknown

Expected upload
Not specified

Data	Code (159)	Discussion (8)	Metadata
There are 3 types of input features:			
<ul style="list-style-type: none">• <i>Objective</i>: factual information;• <i>Examination</i>: results of medical examination;• <i>Subjective</i>: information given by the patient.			
Features:			
1. Age Objective Feature age int (days)			
2. Height Objective Feature height int (cm)			
3. Weight Objective Feature weight float (kg)			
4. Gender Objective Feature gender categorical code			
5. Systolic blood pressure Examination Feature ap_hi int			
6. Diastolic blood pressure Examination Feature ap_lo int			
7. Cholesterol Examination Feature cholesterol 1: normal, 2: above normal, 3: well above normal			
8. Glucose Examination Feature gluc 1: normal, 2: above normal, 3: well above normal			
9. Smoking Subjective Feature smoke binary			
10. Alcohol intake Subjective Feature alco binary			
11. Physical activity Subjective Feature active binary			
12. Presence or absence of cardiovascular disease Target Variable cardio binary			
All of the dataset values were collected at the moment of medical examination.			



DATA PREPROCESSING

	Type 1 error (False Positive)	Type 2 error (False Negative)	Accuracy	F1 - Score	Precision	Recall	Important Feature		
Basic Implementation directly on data	2378	2606	71.52	71.3497	72.2973	70.4267	ap_hi	weight	ap_lo
Treating numerical columns in details	2340	2557	71.9643	71.471	72.3861	70.5788	weight	ap_hi	height
Adding some more columns	2466	2546	71.3059	70.992	71.3222	70.6648	BMI	ap_hi	weight
standardScaler - LeaveOneOutEncoder	2254	2660	71.8669	71.4535	73.1794	69.807			
dropping record	2388	2577	71.09	70.3387	71.142	69.5534	BMI	weight	height
dropping features	2328	2584	71.3986	70.748	71.8433	69.6856	BMI	weight	height



TRYING WITH DIFFERENT ALGORITHM

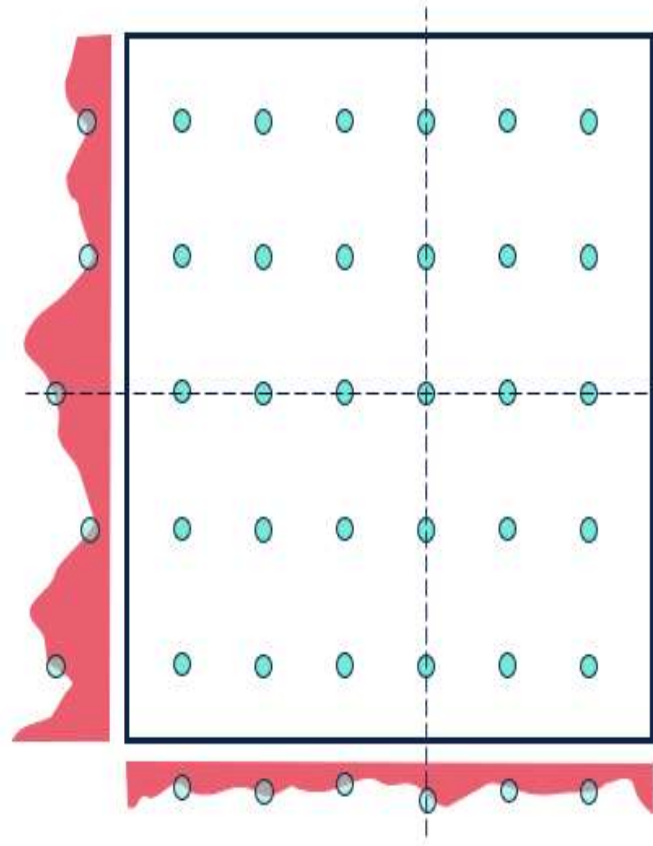
Algorithm Name	Accuracy	Precision	Recall	F-1 Score	False Positive	False Negative
Random forest	74.1341	76.9348	69.1921	72.8583	1818	2
AdaBoost	73.4757	76.6334	66.5468	71.2361	1749	2
Multi-Layer Percerptron	72.8803	73.107	72.508	72.843	2337	2
Decision Tree	73.24	73.19	73.56	73.37	3450	3
NB	73.25	73.19	73.56	73.37	3450	2
XGBoost	73.64	71	78	75	2255	3
SVM	73	71	79	75	2170	3
Bagging	72.87	75	69	72	2448	3
Gradient Boosting	73.11	75	69	72	2375	3
Logistic Regression	69	67	77	72	2054	3



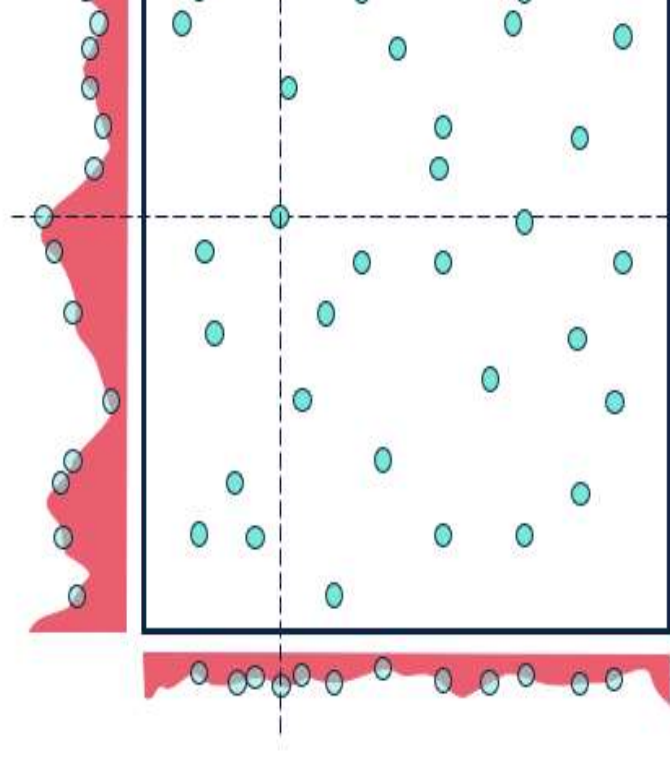
HYPERMETER TUNING

- Random Search
- Grid Search
- TPE (Tree Parzen Estimator)

RANDOM AND GRID SEARCH

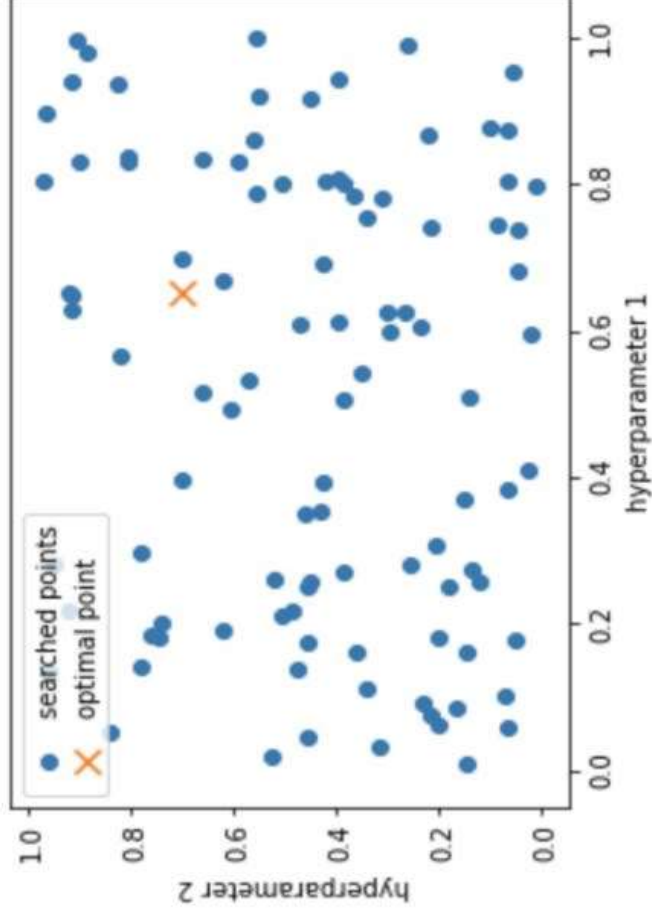


Grid Search

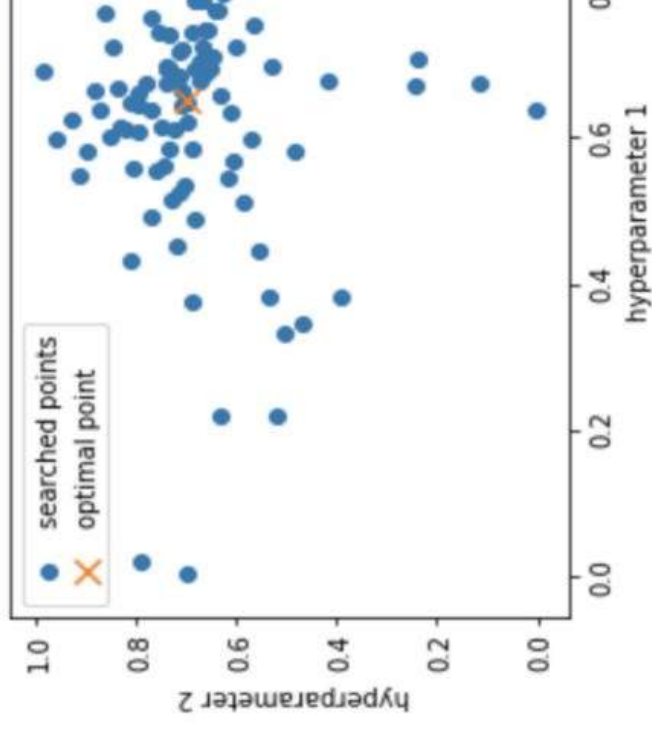


Random Search

TPE



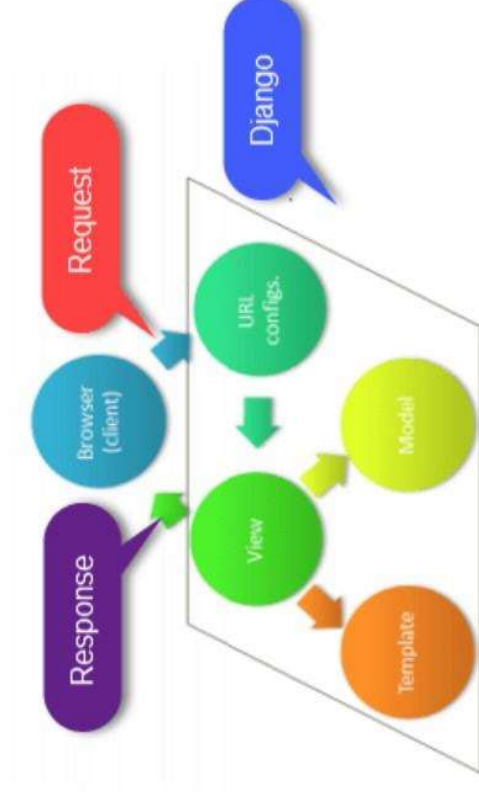
Random search



Sequential Model-based
Optimization (TPE)

DJANGO

Django Architecture



Django Architecture

AWS DEPLOYMENT

