DEPARTMENT OF COMPUTER SCIENCE

MAHILA MAHAVIDYALAYA

BANARAS HINDU UNIVERSITY

PROPERTY DEALING WEB APP

Guided by-

- DR. RAKHI GARG
  (Associate Professor)
- DR. SARVESH PANDEY
  (Assistant Professor)

Presented by-

- Shivani Singh
  (18229CMP003)
- Akriti Singh
  (18229CMP008)

Contents

Introduction

Overview of project

ER Schema

ER Diagram

Flask

Routes

ML Model

HOME SWEET HOME

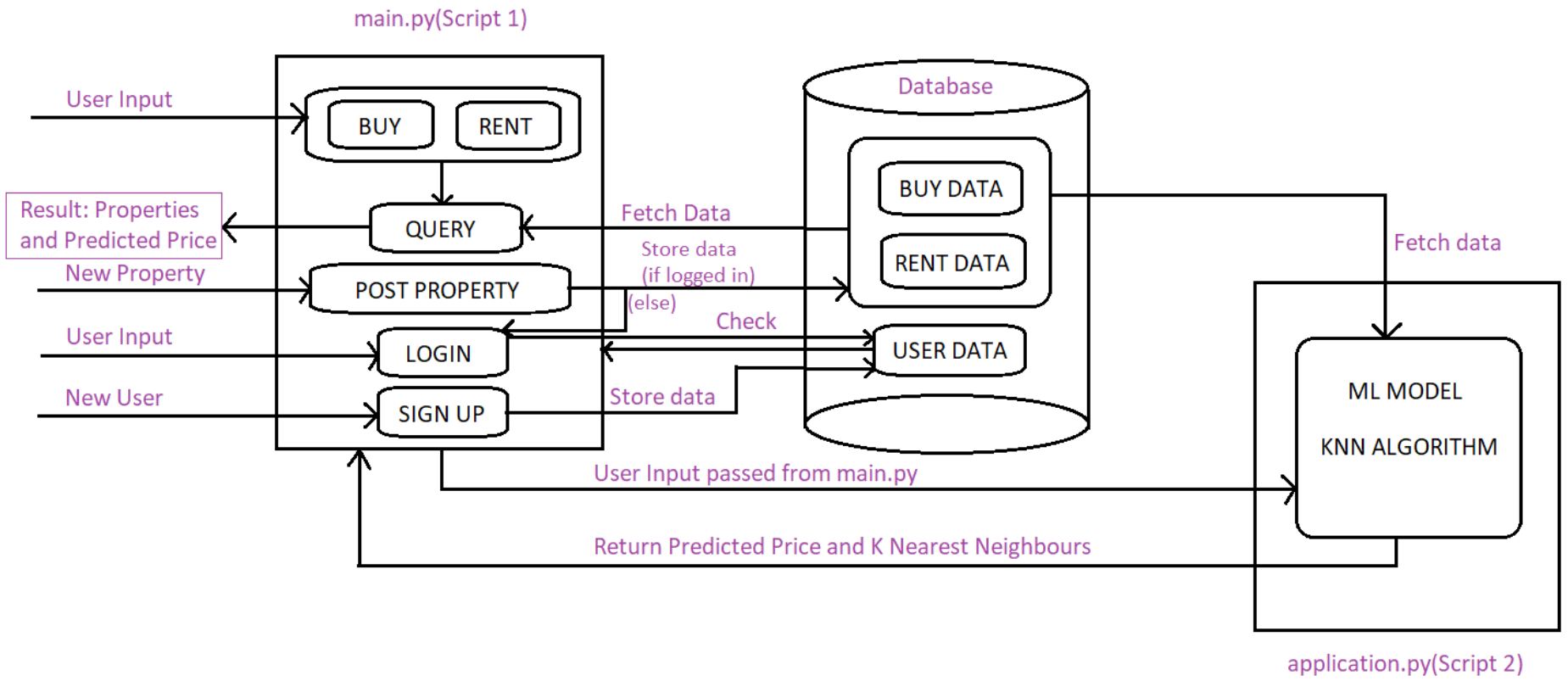
Objective

- To provide a GUI for end-user who is looking for buying/ selling/ renting any housing asset.
- Assist user in selecting the most suitable property based on his/her custom requirements through ML model.

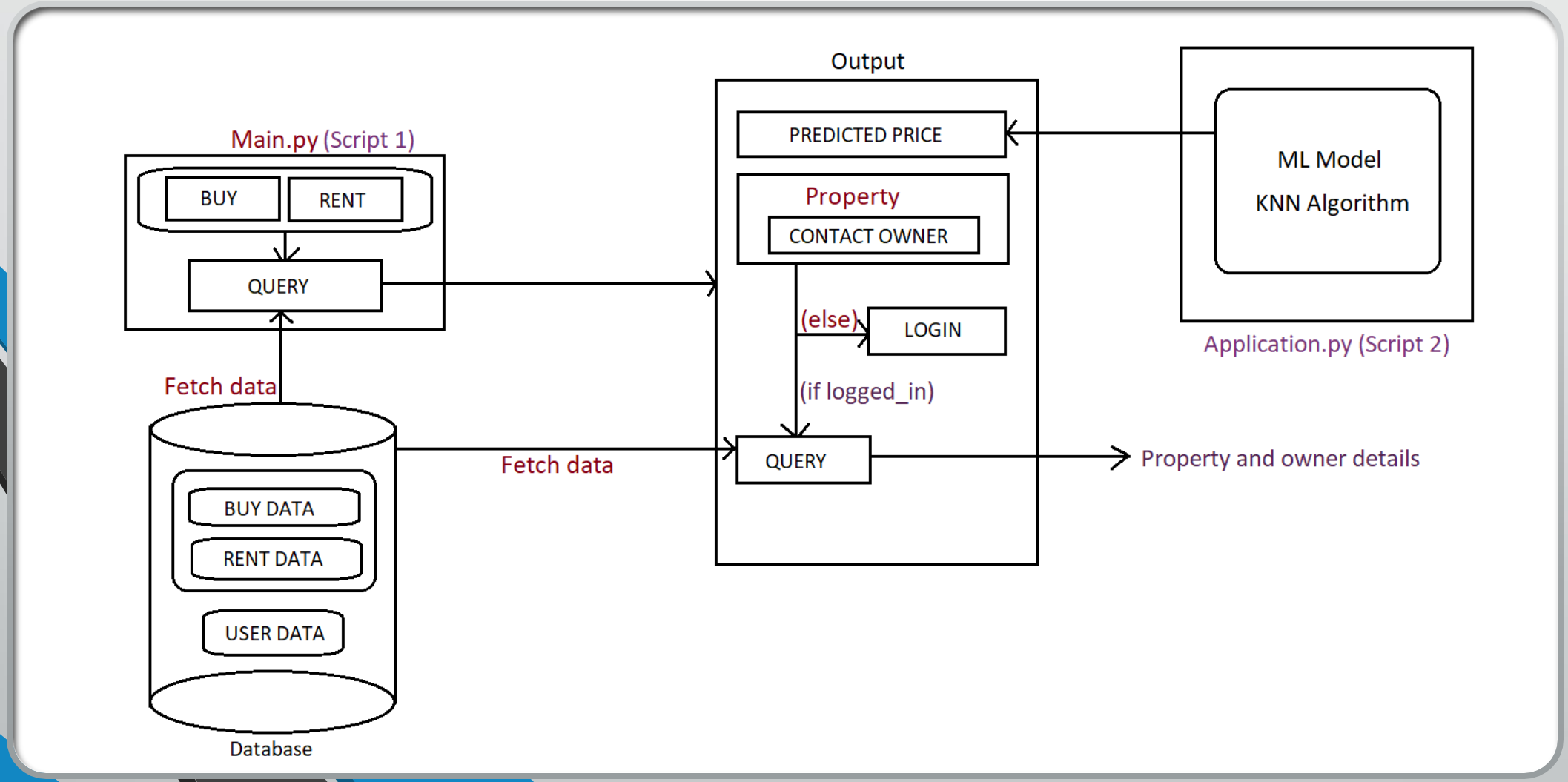
Additional Features

- To predict the price of the property so that the user can verify that the price asked by the owner is appreciable or not.
- Suggesting the best property matches if there is no such property which has the exact same features as asked by the user.

Overview of Project



Execution of Query



ER Schema

USERS

user_id	name	Email_id	Phone_no	password

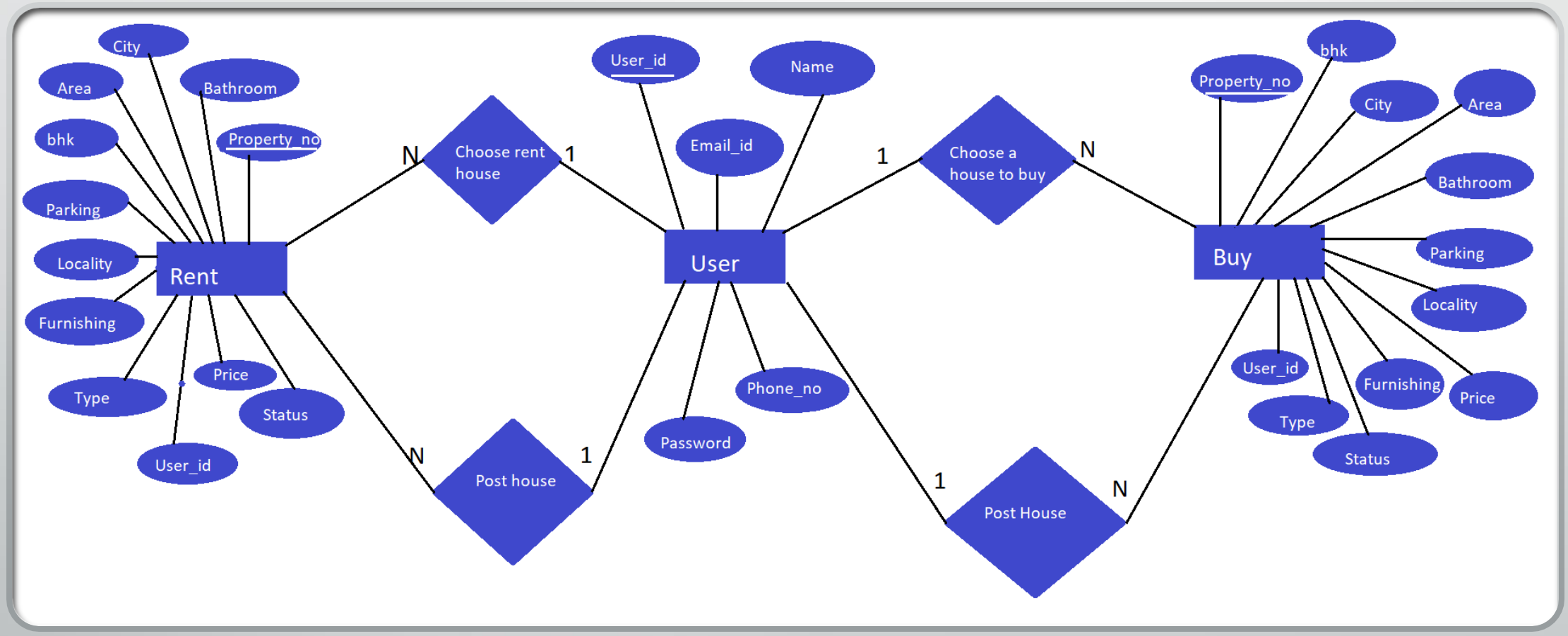
BUY_HOUSE

property_no	city	bhk	bathroom	parking	area	locality	furnishing	price	status	type	user_id

RENT_HOUSE

property_no	city	bhk	bathroom	parking	area	locality	furnishing	price	status	type	user-id

ER Diagram


City
Area
Bathroom
bhk
Property_no
Parking
Locality
Furnishing
Type
Price
Status
User_id
Rent
N
Choose rent house
1
N
Post house
1
User_id
Name
Email_id
Phone_no
Password
User
1
Choose a house to buy
N
1
Post House
N
Property_no
bhk
City
Area
Bathroom
Parking
Locality
Furnishing
Price
Status
Type
User_id
Buy

FLASK

Flask is an API of **Python** that allows us to build up web-applications.
A Web-Application Framework or Web Framework is the collection of modules and libraries that helps the developer to write applications without writing the low-level codes such as protocols, thread management, etc.

Routing:

The web frameworks provide routing technique so that user can remember the URLs. It is useful to access the web page directly without navigating from the Home page. It is done through the following route() decorator, to bind the URL to a function.

Routes in our Project

- **"/ "and "/rent"** - These URLs are associated with the buy() and rent() function which is the home page of our website where user can search their required properties.
- **"/signup" -** This URL is associated with the signup() function which will render the signup.html template where user can enter his/her details to join as a user.
- **"/login" -** This URL is associated with the login() function which will render the login.html template where user can enter his/her details which is further verified from the database. If the user has already signed up than a secret key is generated and the user successfully get logged_in in that session.
- **"/logout" -** This URL will release the session variable and the user will be redirected to the home page.
- **"/post" -** This URL is associated with the post() function which will render the post.html template where user can enter the details of the property he/she wanted to post.
- **"/profile" -** This URL is associated with the profile() function which will render the profile.html template where user can edit his/her details.

- **"/user_saleprop" and "/user_rentprop"** - This URL is associated with the user_saleprop() and user_rentprop() function which will render the userprop.html template where user can view the properties he posted for sale or rent. He can also edit the details of the property or delete it.
- **"/edit_prop/<pro_for>/<user_id>/<prop_no>"** - This is a dynamic URL associated with the editprop() function which will render the editprop.html template where user can edit the details of his/her property.
- **"/delete_prop/<pro_for>/<prop_no>"** - This is a dynamic URL which will delete the property and will redirect the user to "/user_saleprop".
- **"/buy_own_pro"** - This URL is associated with the buy_own_pro() function which will render the cardview.html template where user can view the properties for sale with a particular filter that it is a owner property.

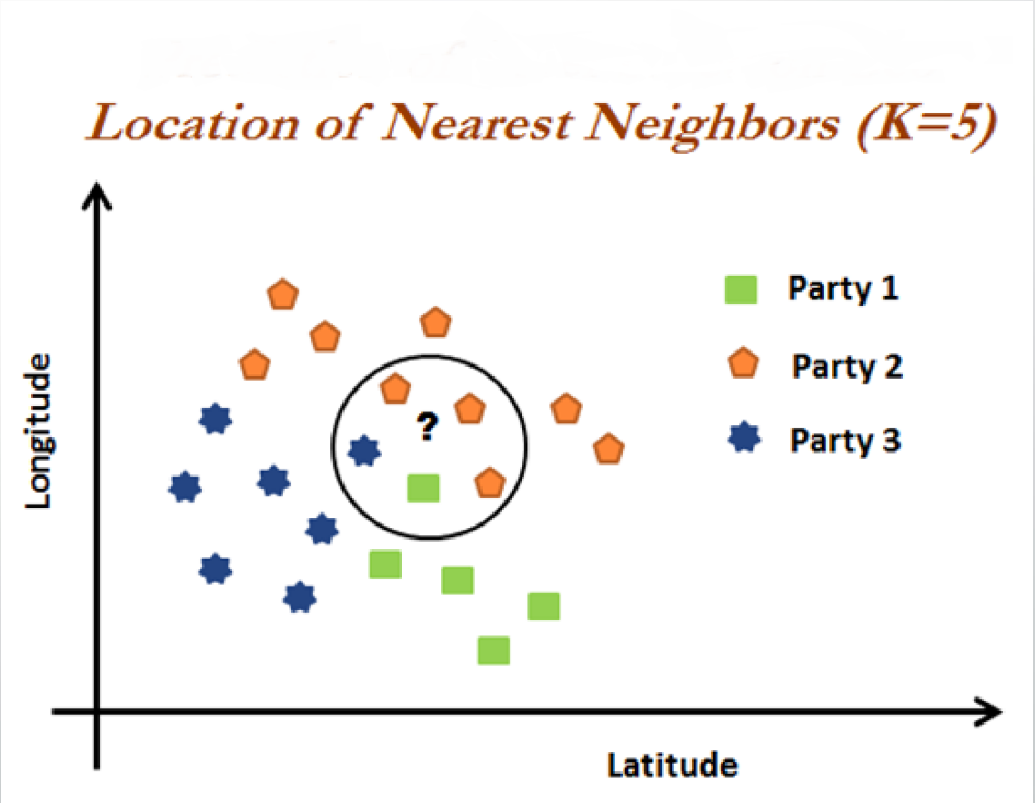
  (There are more such routes available which will filter the properties with some particular condirions.)
- **"/propview/<pro_for>/<user_id>/<prop_no>"** - This is a dynamic URL associated with the propview() function which will render the propview.html template when the user is logged_in otherwise redirect the user to "/login". Here user can view the details of property and it's owner.

Backend (Machine Learning Model)

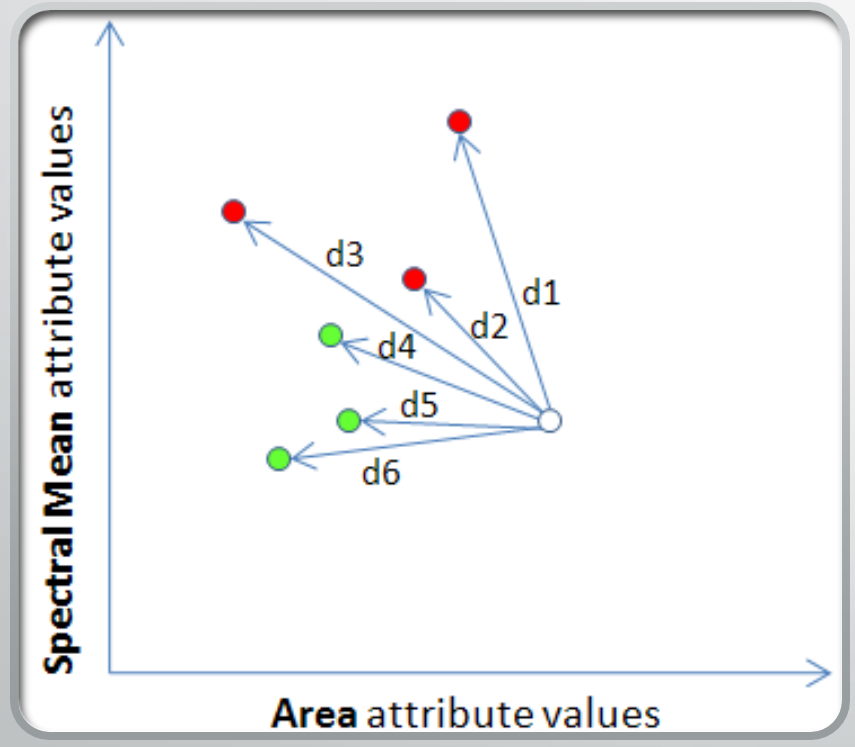
KNN Algorithm (K Nearest Neighbour)

Two types of problems are solved by KNN algorithm

- ❑ Classification problem
- ❑ Regression problem



Working of KNN Algorithm



- Receive an unclassified data.
- Measure the distance from the new data to all others data that is already classified.
- Gets the K smaller distances. (K is a parameter that you define)
- Check the list of classes had the shortest distance and count the amount of each class that appears.

USE OF ML MODEL IN OUR PROJECT

User input works as new data point and it consists of all the details needed to calculate price of the property according to users requirements. Since we have to find value (PRICE) of this new data point (PROPERTY DETAILS) we are facing regression problem and to solve this we are using KNN algorithm.

This is a sorted distance table and from here we have to collect property id of first k distances and mean price of these k properties will be the predicted price of new data point.

DISTANCE Between new data point and previous data point
110.027269
210.009524
260.013461
410.007317
600.499792 ------


THANK YOU