MARCIN LASKOWSKI — 01434258 mll3917@ic.ac.uk
MSc Human and Biological Robotics

MACHINE LEARNING AND NEURAL COMPUTATION

Coursework 1

1 3D Visualisation

The data that were prepared included information about the locations and rental prices of flats from all around the world. What was needed were properties that are located within Greater london and that is why data clean up was needed where Longitude value was from 51.3° N to 51.7° N and Latitude from 0.51°W to 0.34°E.

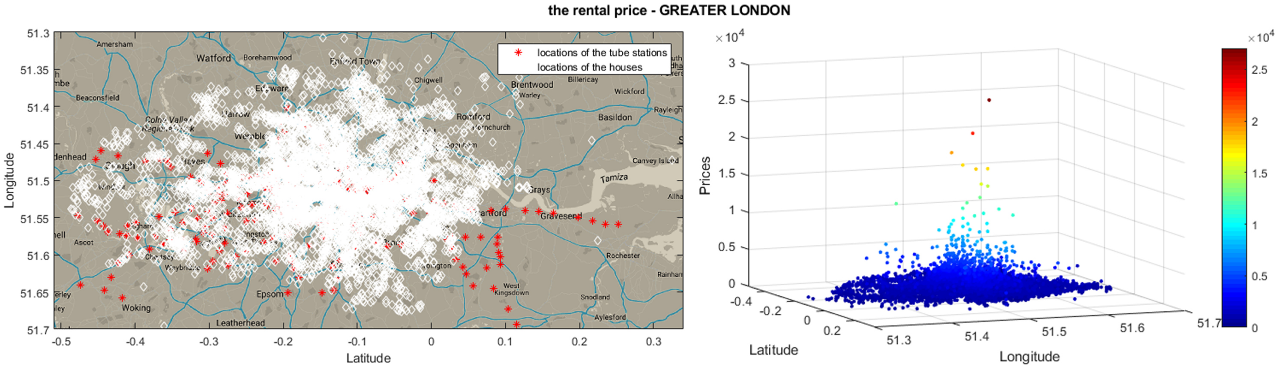

Figure 1: 2D top view and 3D Visualisation of the rental prices within Greater London

2 Training

After cleaning the data, it was necessary to sort the rows in the way to have equally distributed values. That operation was necessary due to obtain correct results at the end. For training, the data were divided into two part (training part: 80% and testing part: 20%). Initially, the operations were performed using the grid method, and the second method (Clustering) was used for compare the results.

2.1 Grid Method

In the first grid method x and y plane (in this case longitude and latitude) was divided into small boxes which created 2D grid. During training in each iteration mean and sigma was obtain calculating longitude and latitude of the all points in the box. However before that operation to obtain the good results it was calculated the mean and sigma of the prices of the points within the box and due to leave only the points which prices are within the range. Second grid method was perform slightly different. Mean of the box was calculated exactly in the same way as during previous approach while the sigma was not calculated from the data only in the box while from the longitude and latitude of all points which were put in to the training. Such operation should gives better distibution. However in this case the result was not so good. Grid method gave really good results, error varies from 700 to 900 pounds. RMSE for each configuration of the grid was calculated and the best result occurs to be 6x5. Using greater grid small gaussian apears in the heatmap and sometimes obtained huge values on the z axis. The reason of that could be the small amount of data in the box which are close to each other and create the very small sigma. To small grid size gives big RMSE, the curvature of the heatmap what very smooth however it was inaccurate.

2.2 Clustering

During first clustering the input data (Longitude and Latitude of all points) were divided into clusters. To perform clustering initially the mean and the standard deviation was calculated to estimate the center points of the cluster. After that operation every point was assigned to some cluster. Mean and the Sigma was calculated in the same way as in the grid method (longitude and latitude of all points inside the box) while here instead of boxes we have clusters. In the Second approach instead of 2D clustering it was performed 3D one. To obtain 3D clusters it was also taken price mean and standard deviation for calculating the center points of the clusters in the 3D space. Then the mean and the value was performed only on the longitude and latitude. For clustering the most optimal number of clusters was 20. Clustering generally gave slightly better heatmap at the end while result are not very good. Due to different positions of clusters, value of the error vary.

3 Testing

At the beginning testing was performed on the 20% of the data (which were left after initial division before training) and the Root Mean Square Error (RMSE) of rental pricees was used as a metric to see the quality of prediction. To obtain the best RMSE crossvalidation was performed. The whole data set was divided into 5 sections, each with the same number of data. In this approach all the time 80% data was trained and 20% was tested. Such a ratio was optimal for the big amount of the data. During tests additional optimizations were performed like finding the best grid size which occurs to be 6x5. Table.1 shows the results of the crossvalidation using different methods.

Crossvalidation	Grid Method 01	Grid Method 02	Clustering 2D	Clustering 3D
1 part	876	887	957	867
2 part	860	730	827	810
3 part	709	789	867	966
4 part	840	909	985	899
5 part	863	967	990	953

Table 1: Table of the RMSE results for the different training approaches with crossvalidation

It can be noticed that final RMSE are in every case much lower than the average rental price across all the rental data. Every method gives different results. It is important to notice that the results may vary due to small amount of features. The most stable is the first method.

4 Heatmap

Below it can be seen the heatmap for all described above cases. Fig.2 shows Grid Method while Fig.3 Clustering.

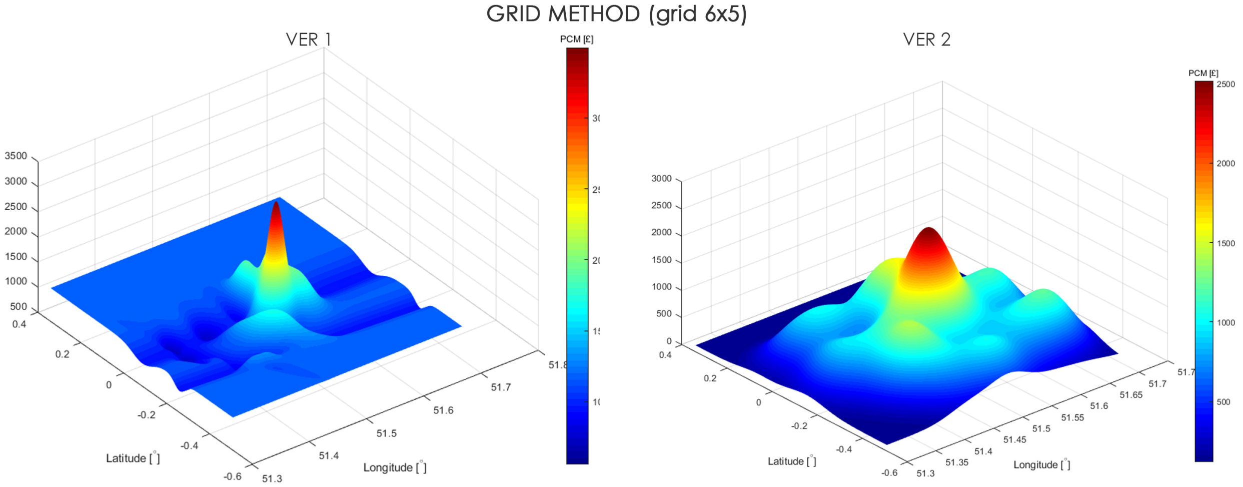

Figure 2: Heatmap of the Grid method with different sigma approach

In the Fig.1 first heatmap do not have any spike, narrow gaussian, there are some valleys which indicates the smaller price for the flat. However the whole gaussian is really good and predicted the prices very precise. The Heatmap on the Fig.2 on the right is fitted very good it occurs with the values around the london. Prices are close to zero which is not true.

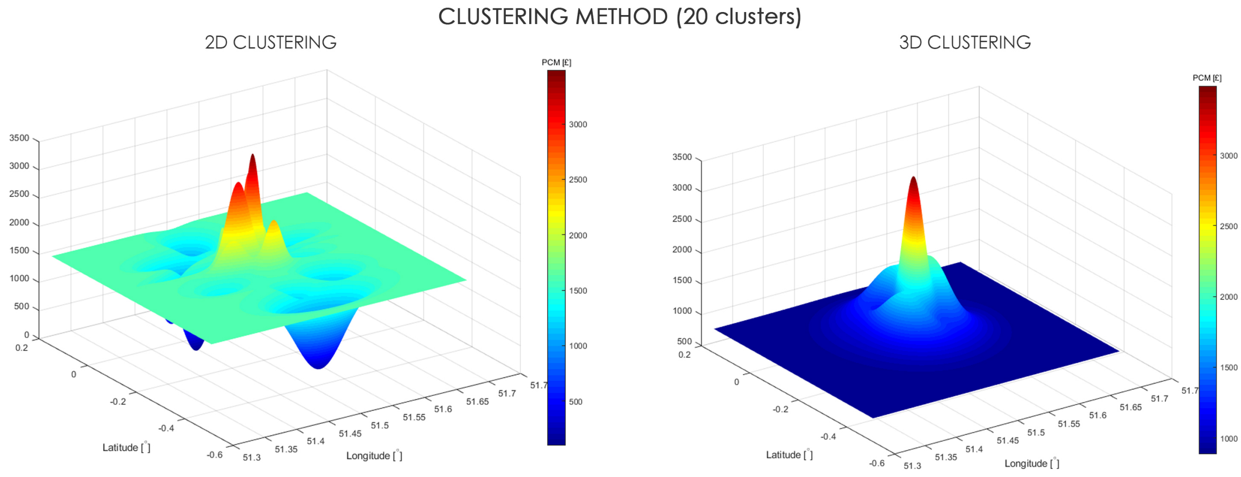

Figure 3: Heatmap of the 2D and 3D clustering

The heatmap in the Fig.3 on the left shows 2D clustering. It can be seen that there are some values, the heatmap is not really precise and in some places is close to zero which could be treated as some overfitting. While the the 3D clustering heatmap is really good, shape is very good, with the bigger pick in the middle and slight small around, however the RMSE is high comparing to other methods.
Finally the best method was Grid method due to pretty good RMSE oscillating around 800 pounds. Heatmap really reflects the Greater London. The curvature is not perfect and definitely could be improved adding some additional optimisation. However the choosing the best model it was important to take into the consideration the best RMSE.

5 Personalised tube

Thank to CID number it was possible to obtain name of the home tube station and location. That location was used to get the price for the flat in the same location. For CID: 01434258 the home station was 'Liverpool Street' with the location values [51.5177, -0.082458]. The rental price for single bedroom room / studio in the Liverpool Street is around 1,964 which is very probable because Liverpool Street is one of London's main railway stations, as well as a tube station in the City of London that is why price is above the mean. Checking the real value near the station occurs that the prices vary around 1,600 to 2,500 pounds per flat. Which means that the predictions occurs to be close to the real one.

6 Conclusions

Summing up it is hard to estimate which method gives the better results. Choosing the best method it was very important to think about this problem as the real live cases, trying to find the best suited model with small amount of disturbances in the heatmap, not only small RMSE. It is also important to notice that the results obtained from two methods vary due to small amount of features. In the real live to obtain more precise results it would be better to apply more features like size of the flat (in square meters) or floor number. To improve the results sometimes the gradient descent is used, but in this case the improvement will be slightly, not big because current method is already calculating maximum likelihood.

7 Appendices

7.1 Matlab code with the main file

Listing 1: Matlab code with the main file

```
% MACHINE LEARNING AND NEURAL COMPUTATION
%% Coursework 1

clear all
close all
clc
load london.mat

tic;
%% <<<<<<<<<<<<<<<<<<<<<<<<<<<< INITIAL DATA >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

% Tube.location      291x2
% Tube.station       291x1
% Prices.location    14807x2
% Prices.rent        14807x1

% Approximate Latitude/Longitude limits for London:
% LATITUDE:  -0.51 to 0.34
x1 = -0.51; x2 = 0.34;
% LONGITUDE:  51.3 to 51.7
y1 = 51.3; y2 = 51.7;


%% <<<<<<<<<<<<<<<<<<<<<<<<<<<< GRETER LONDON MAP >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% LondonMap = imread('LondonMap5.jpg');

figure(1)
% image(LondonMap,'XData',[x1 x2],'YData',[y1 y2])
hold on;
plot(Tube.location(:,2),Tube.location(:,1),'r*')
hold on;
plot(Prices.location(:,2),Prices.location(:,1),'wd')
title('GREATER LONDON')
xlabel('Latitude')
ylabel('Longitude')
xlim([x1 x2])
ylim([y1 y2])
legend('locations of the tube stations','locations of the houses')
pbaspect([abs(x2-x1) abs(y2-y1) 1])



%% TASK 1
% Visualise in a 3D plot (plot3) the rental price raw data for Greater London

Studios = [Prices.location(:,2),Prices.location(:,1), Prices.rent];

sizeStudios = size(Studios);
number_of_locations = sizeStudios(1,1);

k = 1;
% z = 5500;
% calculating the data which are only whithin greater London
for i = 1:number_of_locations
    if  Studios(i,1)>x1 && Studios(i,1)<x2 && Studios(i,2)<y2 && Studios(i,2)>y1
        LondonStudios(k,:) = Studios(i,:);
        k = k + 1;
        i;
    else
        i;
    end
end

figure(2)
% image(LondonMap,'XData',[x1 x2],'YData',[y1 y2])
hold on;
scatter3(LondonStudios(:,1), LondonStudios(:,2), LondonStudios(:,3), 5, LondonStudios(:,3), 'filled')
title('the rental price - GREATER LONDON')
xlabel('Latitude')
ylabel('Longitude')
zlabel('Prices')
xlim([x1 x2])
ylim([y1 y2])
% pbaspect([abs(x2-x1) abs(y2-y1) 1])
view(45, 45);
grid on;
colormap(jet);
colorbar;
```

```


%% Prepare data

% k-fold crossvalidation division. n - number of parts
% for small amount of data more than 80% of data should go for training
% but in this case the most optimal is 80%
n = 5;

% sorting the rows
% LondonStudios = LondonStudios(randperm(length(LondonStudios)),:);

% Building the model
numberOfStudios = length(LondonStudios);

% division data into n parts
OnePartition = numberOfStudios / n;
% Partition = [];
new_error = [];
Error = [];
k = 0;
Results = [];

averageError = [];
W = [];

for c = 1:n

    disp('<<<<<<<<<<<<<<<<< Crossvalidation >>>>>>>>>>>>>>>>>'); c

    Partition = [];

    %%%%%%%%%% TRAINING %%%%%%%%%%
    % Preparing the matrix with the 80% of data for training
    for j = 1:n
        if (j==c)
        j;
        else
        new_Partition = LondonStudios(j:n:end,:);
        Partition = [Partition; new_Partition];
        end
    end

    % Preparing above data for the train regreressor function
    in = Partition(:,1:2);
    out = Partition(:,3);
    % calculating the parameters
    param = trainRegressor(in, out);


    %%%%%%%%%% TESTING %%%%%%%%%%
    % Preparing the matrix with the 20% of data for testing
    testIn = LondonStudios(c:n:end,:);

    % Calculating the predicted prices
    new_results = testRegressor( testIn(:,1:2) , param );

    %%%%%%%%%% RMSE %%%%%%%%%%
    % Calculating RMSE for each part
    Results = [Results; new_results];
    RMSE = sqrt(sum((testIn(:,3)-new_results).^2)/length(new_results))


    % Check the results
    disp('sanityCheck')
    sanityCheck(@trainRegressor,@testRegressor)

%     figure(c)
%     hold on;
%     scatter3(LondonStudios(c:n:end,2), LondonStudios(c:n:end,1), new_results, 'rx');
%     hold on;
%     scatter3(LondonStudios(c:n:end,2), LondonStudios(c:n:end,1), LondonStudios(c:n:end,3), 'gx'); hold off;
%     hold on;
%     heatmapRent(@testRegressor, param)

end

toc;
```

7.2 Matlab code for the heatmap

Listing 2: Matlab code for calculating the heatmap

```matlab
function heatmapRent(testRegressor, params)

% LATITUDE: -0.51 to 0.34
% x1 = -0.51; x2 = 0.34;
% LONGITUDE: 51.3 to 51.7
% y1 = 51.3; y2 = 51.7;
x1 = -0.51; x2 = 0.34;
y1 = 51.3; y2 = 51.7;

lat = linspace(x1, x2, 900)';
long = linspace(y1, y2, 900)';



[LAT,LON] = meshgrid(lat,long);
rent = testRegressor([reshape(LAT, 900^2, 1),reshape(LON, 900^2, 1)], params);
% colormap('hot');
% imagesc(reshape(rent, 30, 30));
% set(gca, 'XTick', linspace(1,900), 'XTickLabel', lat)
% set(gca, 'YTick', linspace(1,900), 'YTickLabel', long)
xlabel('Latitude')
ylabel('Longtitude')
rent = reshape(rent, 900, 900);
surf(LON,LAT,rent)

colormap('jet');
surf(LON,LAT,rent);
shading interp
view
xlabel('Longitude [^\circ]')
ylabel('Latitude [^\circ]')
view(0,90)
hcb=colorbar
title(hcb,'PCM [ ]')
end
```