In this section, we will describe the choices made while clustering with DBScan. We considered several subsets of attributes, and, for each of them, we analysed the algorithm’s trend depending on the variations of **epsilon** and **mintpts**. Then, we plotted the number of clusters obtained in a heatmap. The subsets are:

subset\_1 = ['Age', 'DistanceFromHome','Education','JobLevel','MonthlyIncome',

'NumCompaniesWorked','TotalWorkingYears','YearsAtCompany',

'YearsMean']

Subset\_3 = ['Age', 'DistanceFromHome','Education','JobLevel',

'NumCompaniesWorked','TotalWorkingYears','YearsAtCompany',

'YearsMean']

Subset\_4 = ['HourlyRate','DailyRate','MonthlyRate','DistanceFromHome',

'MonthlyIncome','TotalWorkingYears','Age','YearsAtCompany']

We analysed more subsets of attributes, excluding categorical attributes or discrete attributes having a small domain. We therefore chose numeric variables that were not correlated each other. (The redundant attributes were merged to improve efficiency: an example is “YearsMean” as we can see from the x.x matrix).

Since this type of algorithm doesn’t care about the shape of the clusters, instead of using the silhouette index, like in K-Means, we decided to take advantage of a Heatmap. When the hyper parameters varies, therefore for each pair <**epsilon**, **mintpts**>, it gives us the number of clusters found. The algorithm was performed for all the **epsilon** values of the interval [0.1,2.5] and mintpts [5.15].

pasted-image.tiff

pasted-image.tiff

Analysing the heatmap, we immediately realised that for low values of **epsilon** all points are classified as ‘noise points’ (0 clusters) while, for high values, we tend to obtain a single cluster. Our reasoning, looking at the heatmap led us to go deeper in our research for a solution able to provide at most 2/3 clusters. We divided the research in two areas, for **epsilon** values including [1.1,1.2] and N [8,11] and to **epsilon** [1.8,1.9] and N [9,14].

Subsequently, from the distance function (5-NN distances) we can see that about 85% of the points have a distance ranging from 1.0 to 2.0; moreover with an **epsilon** greater than 2.0 the distances grow very fast and we will most likely include in the clusters even the noise points.

The distance graph leads us to choose the hyperparameters from the second zone, not exceeding the **epsilon** limit. In fact, as we can see from the table, most of the points in the first area are noise points, and the cluster identified contains a few relevant values. The second area confirms the presence of a single large cluster and a quite low number of noise (about 10%), so we will choose hyperparameters belonging to this area. Any variation of the hyperparameters increases noise and proportionally decreases points from this relevant cluster.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| epsilon | | minpts | | Noise points | | Value\_counts | |
| 1.1 | | 8 | | 1069 | | [79,12,7,9] | |
| 1.1 | | 9 | | 1097 | | [27,43,9] | |
| 1.1 | | 10 | | 1125 | | [26,8,9,8] | |
| 1.1 | | 11 | | 1143 | | [25,8] | |
| 1.2 | | 8 | | 845 | | [290,5,25,7,4] | |
| 1.2 | | 9 | | 892 | | [260,24] | |
| 1.2 | | 10 | | 927 | | [179,52,18] | |
| 1.2 | | 11 | | 957 | | [162,45,12] | |
| epsilon | minpts | | Noise points | | Value\_counts | |
| 1.8 | 9 | | 183 | | [977,16] | |
| 1.8 | 10 | | 195 | | [968,13] | |
| 1.8 | 11 | | 209 | | [954,13] | |
| 1.8 | 12 | | 209 | | [945,13] | |
| 1.9 | 9 | | 128 | | [1048] | |
| 1.9 | 10 | | 134 | | [1042] | |
| 1.9 | 11 | | 158 | | [1003,15] | |
| 1.9 | 12 | | 166 | | [994,16] | |

We tested the algorithm using metrics different from the Euclidean one, such as Minkowski, cosine and Cityblock. Unfortunately the result remained the same: a big cluster is prevailing.

Even considering other subsets, the result doesn’t change.

The following scatter-plots describes the distribution as a function of pairs of attributes chosen from ‘subset 4’, with an **epsilon** = 1.8 and **minpts** = 11.

Considering the density parameter, it’s clear to see that does’t provide any further information on dataset’s structure. Regardless of the possible subsets of attributes considered, this won’t change. The algorithm identifies the points as a single large cluster.



