DBSCAn

In questa sezione, descriviamo le scelte adottate durante il clustring con DBScan. Abbiamo preso in considerazione più sottoinsiemi di attributi e per ognuno di loro analizzato l’andamento dell’algoritmo al variare di epsilon e mintpts, plottando in due heatmap le la distanza media con i 5-th noise Point più vicini e il numero di cluster ottenuti.

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']

Come possiamo leggere, sono stati analizzati più sottinsiemi di attributi non prendendo in considerazione attributi categorici ne attributi discreti aventi un dominio molto contenuto. Abbiamo scelto quindi variabili numeriche non correlati tra loro (gli attributi ridondanti sono stati uniti. i.e. YearsMean come possiamo leggere dalla matrice x.x).

Dato che questo tipo di algoritmo si preoccupa poco della forma dei cluster, invece di utilizzare l’indice di silhouette come in K-Means, abbiamo plottato una heatmap con cui, al variare degli iperparametri, quindi per ogni coppia <epsilon,minpts>, ci restituisce il numero di cluster individuati. L’algoritmo è stato eseguito per tutti i valori di epsilon dell’intervallo [0.1,2.5] e minipts [5,15]. 



Analizzando la heatmap ci accorgiamo immediatamente che per valori bassi di epsilon tutti i punti vengono classificati come noise points (0 cluster) mentre, invece, per alti valori tendiamo ad avere un unico cluster. Il buon senso, leggendo la heatmap ci ha portato a restringere la ricerca nel trovare una soluzione con al massimo 2-3 cluster. Abbiamo ristretto la ricerca in due zone, per valori di eps compresi [1.1,1.2] e N [8,11] e in eps[1.8,1.9] e N[9,14].

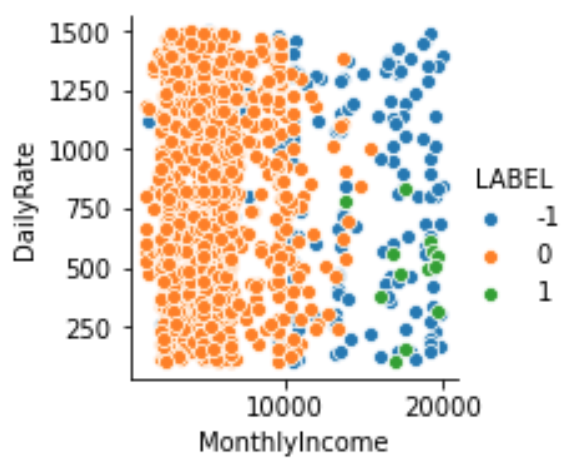
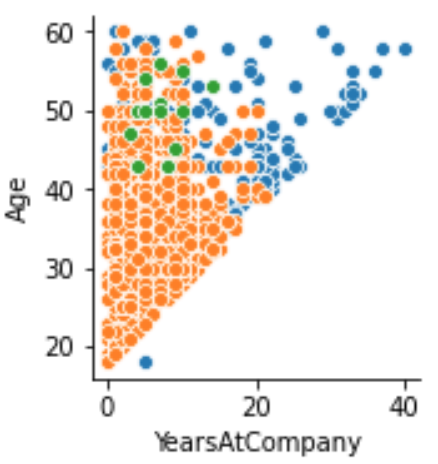
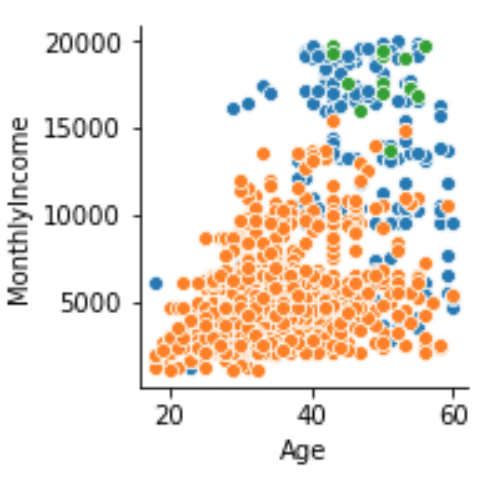
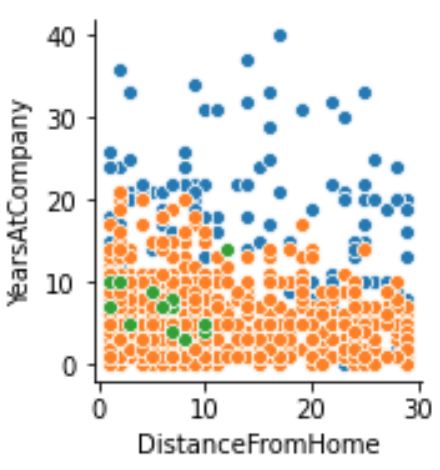
Successivamente, come possiamo leggere meglio dalla funzione delle distanze (5-NN distances) l’85% circa dei punti hanno una distanza compresa tra 1.0 e 2.0, inoltre con un epsilon superiore a 2.0 circa le distanze crescono molto velocemente e clusterizzeremo molto probabilmente anche i noise points.

Il grafico delle distanze ci porta a scegliere gli iperparametri preferibilmente della seconda zona, non superando l’epsilon limite. Infatti, come possiamo notare dalla tabella, la maggior parte dei punti nella prima zona sono tutti noise points e il cluster che vengono identificati contengono un numero irrisorio di punti. La seconda zona ci conferma la presenza di un unico grande cluster ed un numero relativamente basso di noise Point (circa il 10%), quindi sceglieremo una coppia di iperparametri appartenenti a quest’ultima zona. Qualsiasi variazione degli iperparametri porta l’aumento di rumore e il proporzionale decremento di punti da questo grande 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] |

Abbiamo provato ad eseguire l’algoritmo utilizzando metriche differenti da quella euclidea, come minkowski, cosine e cityblock ma il risultato rimane lo stesso, viene identificato sempre un big cluster. Anche considerando gli altri subsets non abbiamo nessun cambiamento. I seguenti scatters mostrano la distribuzione in funzione di alcune coppie di attributi del sottoinsieme scelto per l’analisi (subset 4), prendendo epsilon = 1.8 e minpts=11.

Considerando l’alta densità dei punti, la clusterizzazione density based non ha portato nessun tipo di informazione sulla struttura del dataset, indipendntemente dai possibili sottoinsiemi di attributi considerati. L’algoritmo identifica i punti come un unico grande cluster.