NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
AN AUTONOMOUS INSTITUTION, AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY,
BELGAUM, APPROVED BY AICTE & GOVT.OF KARNATAKA


# DATA MINNING

on

PROPOSAL DOCUMENT FOR DB SCAN

Submitted in partial fulfilment of the requirement for the award of Degree of
Bachelor of Engineering

in

Computer Science and Engineering

Submitted by:

| | |
|---|---|
| ADITI GORDE | 1NT19CS012 |
| BHOOMIKA | 1NT19CS091 |
| SHARVIN PINTO | 1NT19CS174 |
| TEJASWINI MURARI | 1NT19CS201 |

Under the Guidance of
Dr. Vani Vasudevan
Assistant Professor, Dept. of CS&E, NMIT


Department of Computer Science and Engineering

(Accredited by NBA Tie)

# Introduction

**DATA MINING**

The process of extracting information to identify patterns, trends, and useful data that would allow the business to take the data-driven decision from huge sets of data is called Data Mining.

In other words, we can say that Data Mining is the process of investigating hidden patterns of information to various perspectives for categorization into useful data, which is collected and assembled in particular areas such as data warehouses, efficient analysis, data mining algorithm, helping decision making and other data requirement to eventually cost-cutting and generating revenue.
Clustering analysis is an unsupervised learning method that separates the data points into several specific bunches or groups, such that the data points in the same groups have similar properties and data points in different groups have different properties in some sense.

Centrally, all clustering methods use the same approach i.e. first we calculate similarities and then we use it to cluster the data points into groups or batches. Here we will focus on the **Density-based spatial clustering of applications with noise** (**DBSCAN**) clustering method.

What's nice about DBSCAN is that you don't have to specify the number of clusters to use it. All you need is a function to calculate the distance between values and some guidance for what amount of distance is considered "close". DBSCAN also

produces more reasonable results than *k*-means across a variety of different distributions.

**Data mining task**

The planned approach involves most of the standard data mining steps which include:

- Data Understanding: Taking a closer look at the dataset available, particularly understanding the attributes available and the quality of the data. Based on the understanding, planning and modifying the approach to be taken for reaching the end goal.
- Data Preparation: Involving multiple actions to convert the existing raw data into final data that can be used for the analysis, which includes cleaning the data, data reduction based on the requirement, and data transformation. The data is also normalized in this process.
- Training the model: Based on the identified training dataset and the method adopted, the model is trained.
- Evaluating the model: The model trained is then used to predict the values using the test dataset.

## Data Set

To see one realistic example of DBSCAN algorithm, we have used Canada Weather data for the year 2014 to cluster weather stations. The data-frame consists of 1341 rows and 25 columns.

So there are 25 columns and now we need to learn more about the Columns, The names that are difficult to guess

- Stn_Name === Station Name
- Prov === Province
- Tm === Mean Temperature (°C)
- Tn === Lowest Monthly Minimum Temperature
- Tx === Highest Monthly Maximum Temperature
- DwTm === Days Without Valid Mean Temperature
- DwTx === Days Without Valid Maximum Temperature
- DwTn === Days Without Valid Minimum Temperature
- D === Mean Temperature Difference from Normal
- S === Snowfall (cm)
- DwS === Days Without Snowfall
- S%N === Percent of Normal Snowfall
- P === Total Precipitation (mm)
- DwP === Days Without Valid Precipitation
- P%N === Percent of Normal Precipitation
- Pd === No. of days with precipitation 1mm or More
- BS === Bright Sunshine days
- DwBS === Days Without valid Bright Sunshine
- BS% === Percent of Normal Bright Sunshine
- HDD === Degree Days Below 18°C
- CDD === Degree Days Above 18°C
- Stn_No === Station Number; Climate Station Identifier (1st 3 Digits==Indicate drainage basin, Last 4 Digits Sorting Alphabetically)

## Methods and models

**Clustering** analysis is an unsupervised learning method that separates the data points into several specific bunches or groups, such that the data points in the same groups have similar properties and data points in different groups have different properties in some sense.

It comprises of many different methods based on different distance measures.

Centrally, all clustering methods use the same approach i.e. first we calculate similarities and then we use it to cluster the data points into groups or batches. Here we will focus on the **Density-based spatial clustering of applications with noise** (**DBSCAN**) clustering method.

## Density-Based Clustering Algorithms

**Density**-Based **Clustering** refers to unsupervised learning methods that identify distinctive groups/clusters in the data, based on the idea that a cluster in data space is a contiguous region of high point density, separated from other such clusters by contiguous regions of low point density.
Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is a base algorithm for density-based clustering. It can discover clusters of different shapes and sizes from a large amount of data, which is containing noise and outliers.

The DBSCAN algorithm uses two parameters:

- **minPts:** The minimum number of points (a threshold) clustered together for a region to be considered dense.
- **eps (ε):** A distance measure that will be used to locate the points in the neighbourhood of any point.

These parameters can be understood if we explore two concepts called Density Reachability and Density Connectivity.

**Reachability** in terms of density establishes a point to be reachable from another if it lies within a particular distance (eps) from it.
**Connectivity**, on the other hand, involves a transitivity based chaining-approach to determine whether points are located in a particular cluster. For example, p and q points could be connected if p->r->s->t->q, where a->b means b is in the neighbourhood of a.

## ASSESMENTS

Main reasons for using the algorithm according to Ester et.al. are

*1. It requires minimum domain knowledge.*

*2. It can discover clusters of arbitrary shape.*

*3. Efficient for large database, i.e. sample size more than few thousands.*

**The main concept of DBSCAN algorithm is to locate regions of high density that are separated from one another by regions of low density.** So, how do we measure density of a region? Below are the 2 steps —

- Density at a point P: Number of points within a circle of Radius Eps (ϵ) from point P.
- Dense Region: For each point in the cluster, the circle with radius ϵ contains at least minimum number of points (MinPts).

The Epsilon neighbourhood of a point P in the database D is defined as (following the definition from Ester et.al.)

$$N(p) = \{q \in D \mid dist(p, q) \leq \epsilon\} \ldots. (1)$$

Following the definition of dense region, a point can be classified as a **Core Point** if $|N(p)| \geq$ MinPts. The Core Points, as the name suggests, lie usually within the interior of a cluster. A **Border Point** has fewer than MinPts within its ϵ-neighbourhood (N), but it lies in the neighbourhood of another core point. **Noise** is any data point that is neither core nor border point.

Steps of DBSCAN Algorithm:

With the definitions above, we can go through the steps of DBSCAN algorithm as below —

1. The algorithm starts with an arbitrary point which has not been visited and its neighbourhood information is retrieved from the ϵ parameter.

2. If this point contains *MinPts* within ϵ neighbourhood, cluster formation starts. Otherwise, the point is labelled as noise. This point can be later found within the ϵ neighbourhood of a different point and, thus can be made a part of the cluster. Concept of density reachable and density connected points are important here.

3. If a point is found to be a core point, then the points within the ϵ neighbourhood is also part of the cluster. So all the points found within ϵ neighbourhood are added, along with their own ϵ neighbourhood, if they are also core points.

4. The above process continues until the density-connected cluster is completely found.

5. The process restarts with a new point which can be a part of a new cluster or labelled as noise.

## PRESENTATION AND VISUALIZATION

- Tool-Basemap for plotting 2D data on maps using python. Basemaps serve as a reference map on which you overlay data from layers and visualize geographic information. An individual Basemap can be made of multiple feature, raster, or web layers.
- Map projection coordinates will be used as features to cluster the data points spatially along with the temperatures

## ROLES

| STUDENT | ROLE |
|---|---|
| ADITI GORDE | Drafting the project proposal. Work on the initial data understanding, preparation, selection of the clustering algorithm, and assist with training of the model |
| BHOOMIKA | Drafting the project proposal. Focusing on dividing the dataset for training & testing, |

| | |
|---|---|
| | implementing the algorithm, and training the model. |
| SHARVIN PINTO | Training the model, working on the aggregation of the results for the other stated goals of the project, and working on the data visualization. Drafting the project report |
| TEJASWINI MUARI | Working on the data visualization, presentation, assessment of the model trained, and identifying possible ways to improve the model. Drafting the project report. |

## SCHEDULE:

| | |
|---|---|
| 7-01-2022 | Data pre-processing and Training of the model |
| 10-01-2022 | Visualisation and Conclusion |
| 17-01-2022 | Report finalization |

## BIBLIOGRAPHY


- *"A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise"*; Martin Ester et.al. KDD-96 Proceedings.

- Density Based Clustering Methods; Gao,J., Associate Professor Buffalo University. Presentation Link.

- https://towardsdatascience.com/

- TutorialsPoint

- Github