Lecture 10:
Modeling Data Surrogate Based Modeling, Hybrid Piecewise Polynomial Modeling, and Random Forest

COSC 526: Introduction to Data Mining
Spring 2021

THE UNIVERSITY OF TENNESSEE KNOXVILLE

BIG ORANGE. BIG IDEAS.®

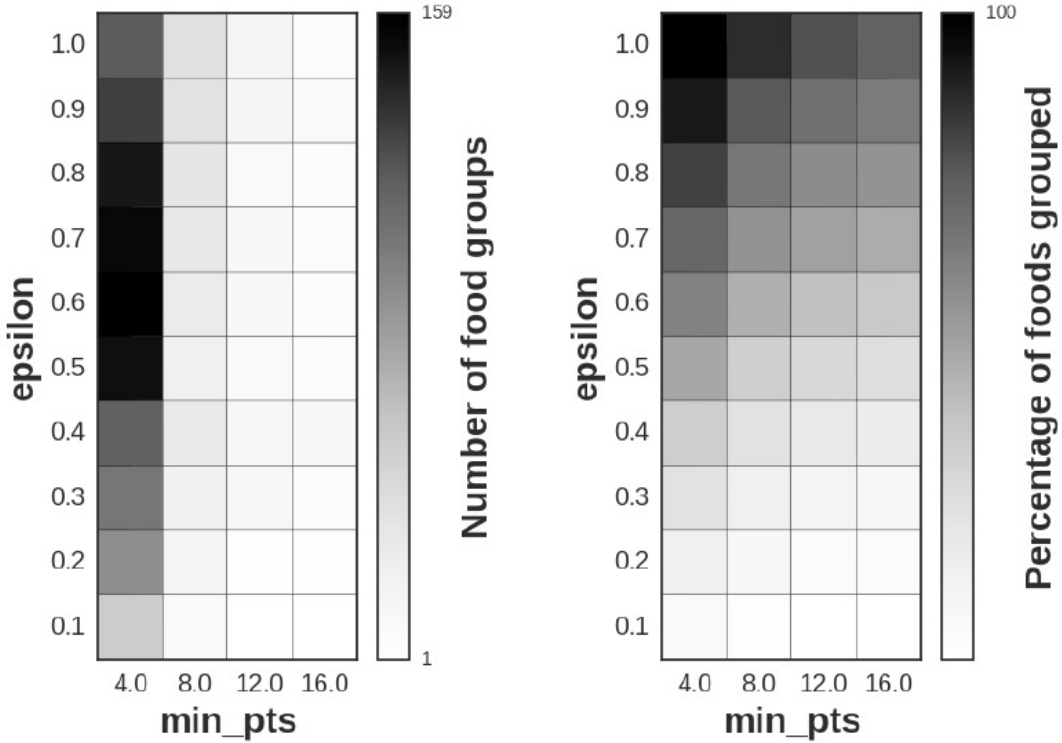
Assignment 9

Assignment 9

- We use Density-based spatial clustering of applications with noise (DBSCAN) to cluster food items based on macronutrient and micronutrient content.
  - We learned about DBSCAN in the last lecture.
- We will reference the paper *Development of a Scalable Method for Creating Food Groups Using the NHANES Dataset and MapReduce
  - The paper can be downloaded from the ACM Digital Library -- available through the University
  - The paper is also in the course repo
- We use the data preprocessing code and the DBSCAN code provided in this paper's git repo.
  - https://github.com/TauferLab/NHANES-Analytics

Assignment 9

- We preprocess the raw NHANES dietary data
  - Data location: ./data/NHANES-20**-dietary.csv
- We cluster the data with DBSCAN
- We reproduce Figure 6 from the
  - Your figures may be slightly different because you are only using 2 years of NHANES data, where the paper used 7 years of data



(a) Number of clusters (b) Clustered items

Figure 6: Heat maps showing the number of clusters (a) and the percentage of clustered food items (b) when the Euclidean distance is used.

In [10]:
```
# Note: If you are having trouble loading Spark, try uncommenting the following two lines
#import findspark
#findspark.init()

from pyspark import SparkContext
sc = SparkContext.getOrCreate()

# Load code from paper
from src.preprocess import cleanNHANESData
from src.DBSCAN import DBSCAN
```

We are giving you the DBSCAN sw
Running on spark

In [12]:
```
# Define DBSCAN parameters
epsilon = 1
min_pts = 4
metric = 'euclidean'

# Cluster the food items with DBSCAN
# This may take a few minutes, depending on your machine!
food_clusters = DBSCAN(sc, clean_data, epsilon=epsilon, minpts=min_pts, metric=metric)

# Look at the cluster results
# Elements in the RDD are key-value pairs of (food ID, cluster ID)
print('Clustered food items: {}'.format(food_clusters.count()))
print(food_clusters.take(1))
```

```
DBSCAN completed in 10 iterations
Clustered food items: 1376
[('92511010', '11100000')]
```

Keep in mind that you have the DBSCAN setting hardcoded in one of the prepared cells
In your solutions, you may have to reset the parameters

Assignment 9

- Problem 1: Measure metrics to characterize clusters
  - *Define the functions which intake food clusters (i.e., the output of DBSCAN) and returns **percentage of food items clustered** and **number of food clusters found**.*
- Problem 2: Plot a heatmap
  - *Define the function that intakes a 2D array and plots a heatmap*
- Problem 3: Recreate Figure 6 from the paper
  - *Cluster food items with the **Euclidean distance metric** and epsilon and min_pts values in ranges [2,4] and [4,7], respectively*

Assignment 9

- Problem 4: Visualize n-dimension spaces
  - *Project the data down to 2 dimensions and visualize the clusters*
  - *Use t-Distributed Stochastic Neighbor Embedding (t-SNE) for dimensionality reduction*
- Problem 5: Euclidian, or not Euclidian, “that is the question”
  - *Use the given code to examine the clusters found using DBSCAN and Cosine Similarity and compare with clusters found using the Euclidian distance*

Assignment 10

Introducing Pandas

- Walk through some early stages of data downscaling
- Introduce you to a powerful Python package for data analysis: Pandas
- Use Pandas to executed these stages
  - data processing,
  - modeling to generate fine-scale predictions, and
  - visualization

Introducing Pandas

- In previous assignments, we used
  - csv library to read data files
  - numpy and pyspark libraries to deal with missing values
- Both functionalities are readily available with Pandas

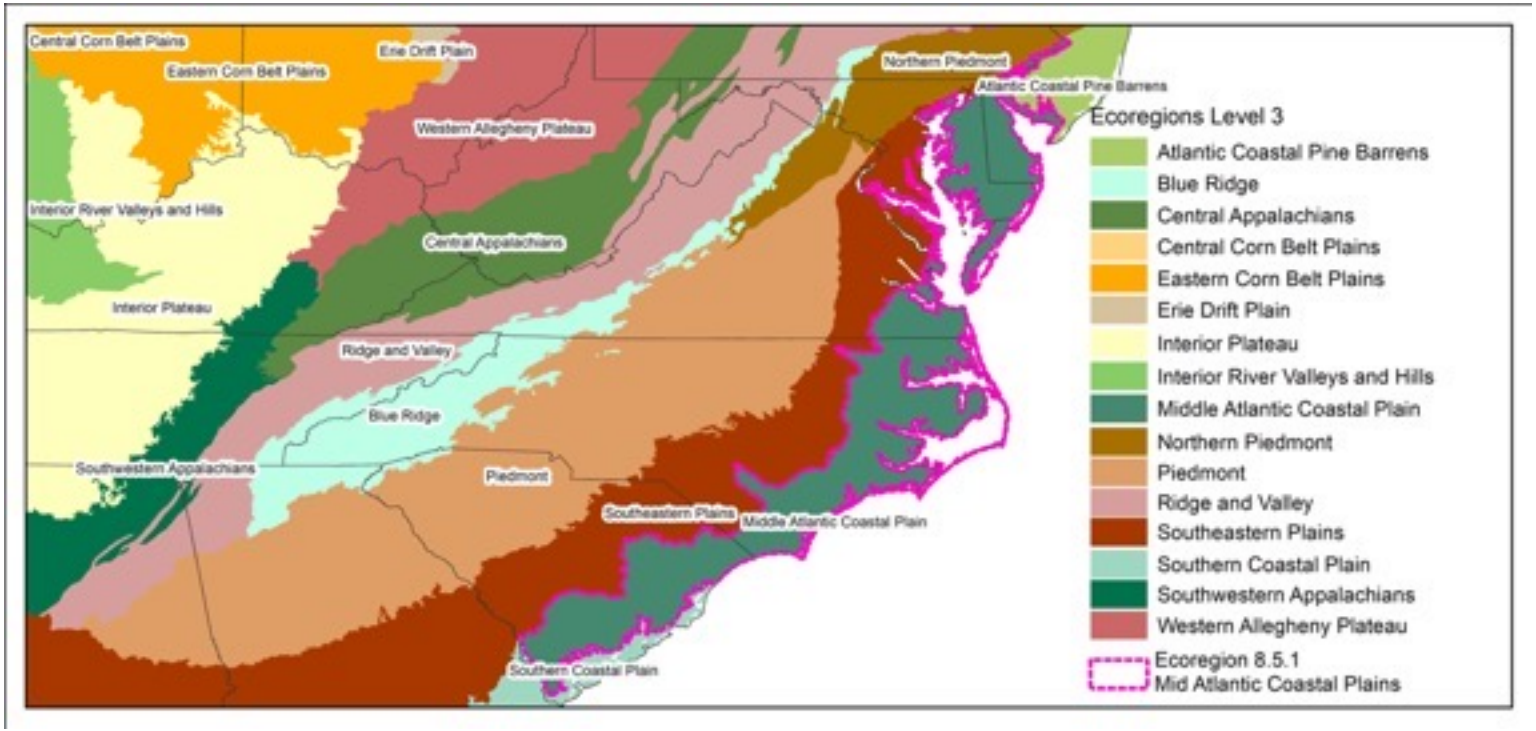
Assignment 10

- Preprocessing data (different from dietary data):
- Problem 1: Remove all monthly columns from a given data frame except for a specified month
- Problem 2: Drop the rows that have an NA in that column
- Problem 3: Use KNN to predict data on the region of interest
- Problem 4: Create heatmaps for the training and prediction soil moisture values
    - Course-grained map: Original Soil Moisture Heatmap
    - Fine-grained map: Predicted Soil Moisture Heatmap

Data features and preparation

- Process data in Pandas DataFrames to prepare it for modeling
- Data for the Mid Atlantic Coastal Plains, a North American ecoregion containing the state of Delaware
- Subset of data for a single year (2016)
- Two comma-separated text files:
  - "Delaware_train.csv" with 29 columns
  - "Delaware_eval.csv" with 17 columns
- Initial two columns: x (longitude) and y (latitude) coordinates
- Final fifteen columns: 15 topographic parameters.
- Twelve columns (of the "train" file ONLY): monthly soil moisture averages for 2016 for the approximately 27 km x 27 km pixel with centroid at the given coordinates

Middle Atlantic Coastal Plains



Level III ecoregion 8.5.1 Region with a broad range of moisture ratios

Data predictions

- Feed the processed data into a pre-prepared modeling script to produce downscaled soil moisture predictions
- Coarse-resolution "train" data is used to generate a model
- Model is evaluated on the fine-resolution "eval" data

Data visualization

- Take advantage of Pandas' integration with matplotlib to create heatmaps for visually comparing your downscaled soil moisture product to the original data

Project

Project

Motivation Describe the motivation of your work. To build the motivation, you can answer these questions:

- What is the problem you are tackling?
- How is the problem solved today?
- Write a paragraph of 200 - 300 words

Contributions List between 2 and 4 contributions of your work. Contributions are bullet points that define your solution. E.g.,

- We build a system that ....
- We validate the system accuracy by ....
- We measure the performance of the system by ...
- Write a section of 150 - 200 words

Project

Tests List the type of tests (measurements) you will perform. E.g.,

- What are your metrics of success?
- Where do you run your tests?
- What tests do you perform?
- How many times do you run each test?
- What do you measure?
- Write a section of 250 - 350 words.

Milestones

- March 26: **Define your project**
  - Describe the motivation of your work
  - List between 2 and 4 contributions of your work
  - List the type of tests (measurements) you will perform
- April 2: No lecture

Milestones

- April 9: **Create a new notebook with your solution**
  - Write down the steps of your solution in distinct text cells; add one or multiple cells (as needed) to hold your code for each step. You can leave these software cells empty for the moment. Expand the text cells describing your solution.
  - Add visualization cells that allow you to visualize results. You can leave these software cells empty for the moment.
  - Add software to the code cells that upload data from source and pre-process data.
  - Push your notebook into your GitHub repository as frequently as needed

Milestones

- April 16: **Finalize software and complete the test run within your notebook**
- April 23: **Create your poster and get feedback, submit draft**
- **April 30: Submit your final notebook and poster in GitHub**
- **May 7: Submit your 2-page abstract in GitHub**