Micropollutants assessment of Vidy Bay backed by machine learning algorithms

Renaud Nasch

Abstract

This report will attempt to assess the micro-pollution of the lake at three different locations, using machine learning algorithms. Three methods will be used, 2 cluster methods, K-means and DBSCAN to visualize the data and a random forest to try to predict the pollutants.

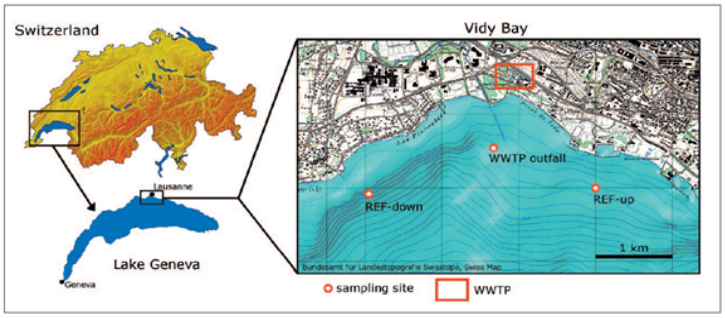

Fig. 1. Situation and map of the Vidy Bay showing the sampling locations WWTP outfall (Swiss coordinates: 534'672/ 151'540), REF-up (Swiss coordinates: 536'000/151'000) and REF down (Swiss coordinates: 533'048/150'920). Coordinates are in Swiss Grid system with datum CH1903.

Figure 1. Situation and map of sampling location,(1 is the ref up, 2 is the WWTP outfall and 3 is the ref down)

1. Intro

Monitoring and maintaining the health of the water is critical to the ecosystem services the lake can provide. The lake is essential for meeting the water needs of a large population. It is also the habitat for many organisms that depend on the lake. Therefore, it is necessary to study it, in order to better protect it. This report uses means still not widespread to approach the environmental questions, and this by the prism of the machine learning, which will be able to bring answers on the environment of the lake.

2. Data

The data for this report were acquired by Bonvin et al. in 2012. The data consist of 11 vertical concentration profiles at 3 different locations on Lake Geneva for each month of the year except February.

There are 11 months of total data, but for this report the data used is for only two months, April, May, because the original data is stored in 11 separate Excel files, all of which are different. For time reasons, it was easier to start with the first two months and see if it works before processing the entire data set.

The data consist of 5 columns, which held information 38 pollutants, on month of sampling, location of sampling, depth of sampling, and the concentration measured.

The data is split into 0.8 train and 0.2. The test set is then split in two, to create the validation set.

The code can be obtained from this link:

shorturl.at/qJPU8

The code can be obtained from this link:

shorturl.at/gDEMS

3. Methodology

Missing values are replace by 0, as it can be estimated that no measurement of pollutant can be simplified as no pollutant.

3.1. KMeans

The hyper parameters search is done by two methods first visually by choosing the K with the elbow technique, and the silhouette score. And secondly a hyper parameters search is done on three chosen parameters: “number of clusters”, which is the number of separate group of data, “max_iter”, which is the number of iteration for a single run, and the

"algorithm" which chooses the intern algorithm. The initial parameters "k-means++" are chosen by default, as this is a better initial start for KMeans.

3.2. DBSCAN

To compare with the first model DBSCAN is chosen. The data are first normalized. For DBSCAN, the most important hyper parameter is the "eps", which is the maximum distance of a sample to be determined as neighbors, this distance is a subjective distance based on the assumptions of the data sets. The "min_samples" is the minimum number of samples in a neighborhood for a point to be considered a center point.

3.3. Random Forest

After finding that the DBSCAn is not usable in this case compared to the KMeans, I decided to change the direction of the report to do a random forest. The random forest classifier is used with "n_estimators" being the number of trees in the forest, "max_depth" which is the maximum depth of trees, and "min_samples_split" which is minimum number of samples required to split an internal node. The last hyper parameter is the "max_features" which is the number of features to consider when looking for the best split.

4. Results

4.1. KMEANS

The silhouette score seems to have a optimum k of 4. The elbow method indicates also a k of 4.

Figure 2 shows the pollutant at various depth and with concentration as factors of marker size, all the data is plotted, the figure 4 shows the data but with the time as y axis.

In table 1 the hyper parameters search can be found

Table 1. best hyper parameter KMeans

HYPER PARAMETER	BEST
ALGORITHM	'ELKAN'
MAX_ITER	1000
N_CLUSTER	4

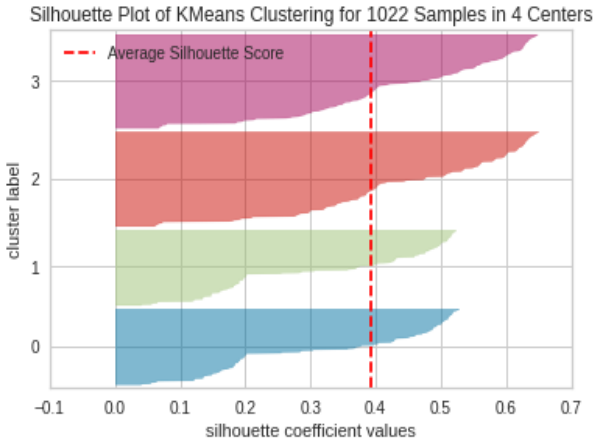

Figure 2. Silhouette score

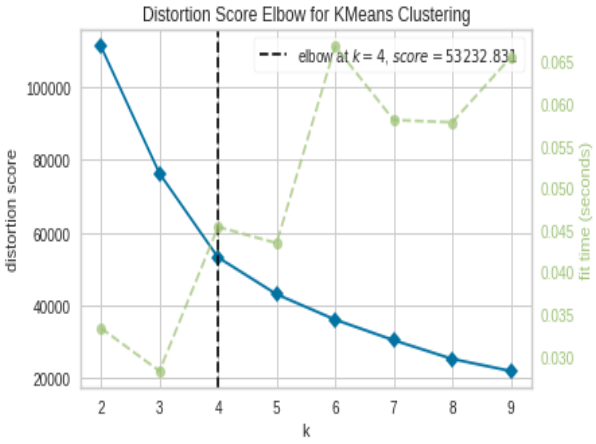

Figure 3. K elbow technique

Table 2. best hyper parameter DBSCAN

HYPER PARAMETER	BEST
EPS	50E-3
MIN_SAMPLES	1000
N_'EUCLIDEAN'	4
N_CLUSTER	13

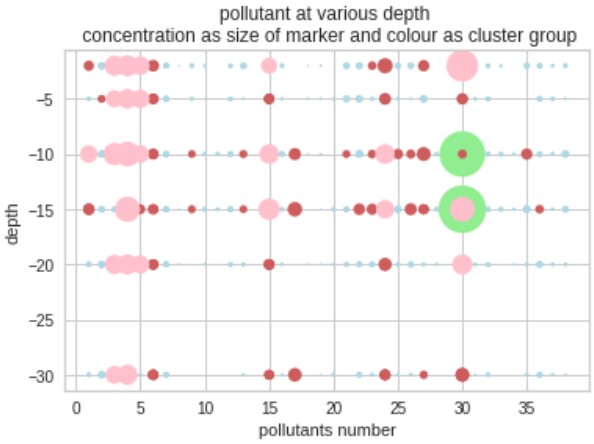

Figure 4. pollutant at various depth concentration as size of marker and colour as cluster group

Figure 4 shows the concentration at different depths for each of the 38 pollutants and the color indicates the groups to which they belong.

4.2. DBSCAN

The DBSCAN search for hyper parameters as shown in table 2 shows that it estimate 13 clusters and the other important hyper parameters

Figure 5 shows that DBSCAN calculated all the data, I failed to retain only the useful information. (You can see that the "diagonal" shape of the dispersion is probably the number of pollutants, and you can see the elbow shape which is probably the depth and two main groups "up,down" which are probably the two months of data).

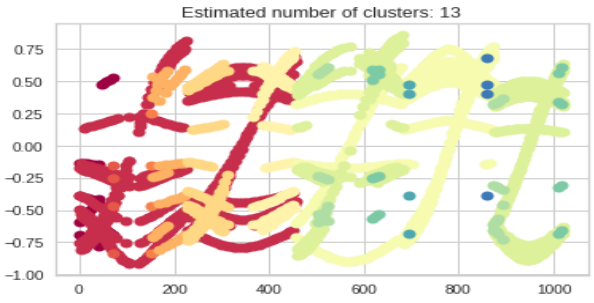

Figure 5. DBSCAN visualization

4.3. Random Forest

The table 3 shows the accuracy score o0ver the test set and the validation set The random forest seems to have the best usable result on my data, the difference between the other 2 models and this one is that the objective is a bit different, the random forest predicts the concentration for a specific point in time.

Table 3. accuracy score random forest

SCORE	VALUE
ACCURACY SCORE(Y TEST, Y PRED)	0.58
ACCURACY SCORE(Y VAL, Y PRED VAL)	0.57

5. Discussion

After struggling to implement the data in order to detect pollution regime by doing clusterization, it has been decided to use the random forest model. To have some results to discuss. The clustering models gave some output based on the computer simplifying the data, as a lot of data shared some intricate similarities, the models grouped by time, location and depth as it is simpler. The KMeans and DBSCAN "k" results are quite different 4 and 13 respectively, which contribute to the difficulty to compare the two models. Figure 5 clearly shows a code problem, it was decided to keep it in this report.

For the random forest the results can be more useful, as it is clearer what is the results. But the accuracy is not really high, and the main problem with this model is that it can only predict a concentration that is already existing, so in fact this is just searching the nearest concentration, there is no "in-between" concentration

What I learned in this report is that having a good start with a clear scientific question and a clear goal is primoridial; knowing what should the data look like in terms of arrange-

ment is also a big part of this project, as I tried at leas 3-4 arrangement, some decreased the dataset, other were not adapted for the models used. And then having strong coding skills can be useful, I'm starting to understand more of the python language than at the beginning of the semester.

6. Conclusion

Although my results are far from what I expected when I started this report, perhaps I took something too complicated to work with, knowing my understanding of machine learning and my coding skills. The conclusion is that the concentration of pollutants varies with time, location, and depth, and as expected the plume at the treatment plant is highly concentrated in micropollutants than upstream and downstream.Improving the code is possible and other means and scientific questions could shed light on the phenomenon.

7. Reference

Bonvin, F., Chevre, N., Rutler, R., Kohn, T. (2012). Occurrence, fate and ecotoxicological relevance. ARCHIVES DES SCIENCES, 13.

8. acknowledgment

Thanks to my colleague Joel for his help Thanks to Tom Beucler and Milton Gomez. Thanks to StackOverflow for bits and pieces.