



Semaine Math-Entreprise

ANALYSE PAR IMAGES SATELLITES DE LA DYNAMIQUE D'EUTROPHISATION

APPLICATION AU LAC D'AYDAT

Réjane JOYARD, Alexis ALZURIA, Raphaël BARATEAU, Océanne BOUSQUET,
Grégoire DOAT, Yannis LEBRUN, Aurélien LONCQ, Malik MASRI

Encadré par Eloïse COMTE, Catherine CHOQUET et Michel BERTHIER

TABLE DES MATIÈRES

Contexte	1
Introduction	1
I. Traitement des images satellites	2
1. Analyses des images	2
2. Création et placement du masque	3
3. Le problème des nuages	5
4. Concentration de Chlorophylle-a	8
5. Comparaison avec la bathymétrie	10
II. Études statistiques	13
1. Présentation des données météorologiques	13
2. Tri et étude des données météorologiques	14
2.1. Gestion des valeurs aberrantes	14
2.2. Variables étudiées et conservées	15
3. Recoupement des données observées et mesurées	17
3.1. Mise en relation des données météorologiques	17
3.2. Réduction de dimension et clustering des données météorologiques	20
3.3. Recoupement avec les données observées par satellite	21
Annexes	24
Appendix A Cross-Corrélation	24
Appendix B UMAP	25
Appendix C HDBSCAN	26
Table des figures	28
Références	29

Contexte

La Semaine Maths-Entreprise constitue une composante essentielle du programme du Master MIX de l'Université de La Rochelle, visant à rassembler l'ensemble des étudiants autour d'une problématique présentée par une instance extérieure, en l'occurrence l'Institut National de la Recherche pour l'Agriculture, l'Alimentation et l'Environnement (INRAE). L'objectif est de formuler une réponse à cette problématique au cours d'une semaine d'activités dédiées.

Le but est de s'immerger dans le domaine de la recherche en soumettant un document digne des standards et apportant des réponses pertinentes et réfléchies.

Un aspect tout aussi crucial de cette expérience est l'acquisition de compétences en travail d'équipe, impliquant une organisation efficace, une répartition judicieuse des tâches, une bonne communication et éventuellement une gestion des conflits. Les délais étant particulièrement courts, c'est aussi l'occasion d'apprendre à être efficace et à optimiser au mieux l'organisation des différentes tâches.

Introduction

L'eutrophisation des lacs est le résultat d'un apport conséquent de nutriments conduisant à un développement important d'algues appauvrissant l'écosystème du lac. Ces nutriments (phosphore, azote) proviennent des activités anthropiques (agriculture, industries) des bassins versants et sont drainés par les rivières. Parmi les algues à risques, on distingue les cyanobactéries. En effet, ces dernières peuvent produire des cyanotoxines, affectant la biodiversité des lacs et pouvant être dangereuses pour la santé. La réglementation implique donc une fermeture des plages et des accès aux activités nautiques et de pêche sur les lacs lorsque le taux de cyanotoxines est trop élevé. La prolifération de cyanobactéries toxiques a donc des conséquences environnementales et économiques négatives. La gestion des bassins versants est primordiale pour garantir la qualité de l'eau d'un lac, mais elle peut être complexe selon l'évolution du bassin (croissance de zones habitées, gestion des eaux pluviales et usées, entretien du réseau hydrographique). De plus, les changements climatiques engendrent des conditions favorables à la présence des cyanobactéries. Le défi est de concilier l'équilibre écologique naturel des écosystèmes avec les pressions anthropiques associées.

Le travail que Deguene Diene [1] a effectué durant son stage a permis l'étude de l'eutrophisation du lac d'Aydat et ainsi de dégager des tendances significatives concernant la présence des cyanobactéries. Cependant, ce travail s'appuyait seulement sur des images ne présentant aucune couverture nuageuse. Nous présentons dans ce rapport une approche répondant à ce problème en prenant en compte la couche nuageuse. Dans un premier temps, nous nous intéresserons au traitement des données satellites (comme la création des masques nécessaires) puis nous étudierons les données météorologiques ainsi que leurs potentielles corrélations avec les données observées grâce aux images multispectrales.

I. Traitement des images satellites

1. Analyses des images

Le site Copernicus, associé au satellite Sentinel-2, propose des images prises dans des longueurs d'onde précises. En fonction de celles-ci, Sentinel-2 propose différents niveaux de précision comme on peut le voir sur la figure 1.1 ci-dessous. En abscisse se trouvent les différentes longueurs d'ondes et en ordonnée le nombre de mètres correspondant au côté d'un pixel.

Par exemple, 10 mètres est associé à la longueur d'onde B04 qui signifie que sur une image produite par les capteurs de la longueur d'onde B04, un pixel correspond à une surface de 10×10 mètres au sol.

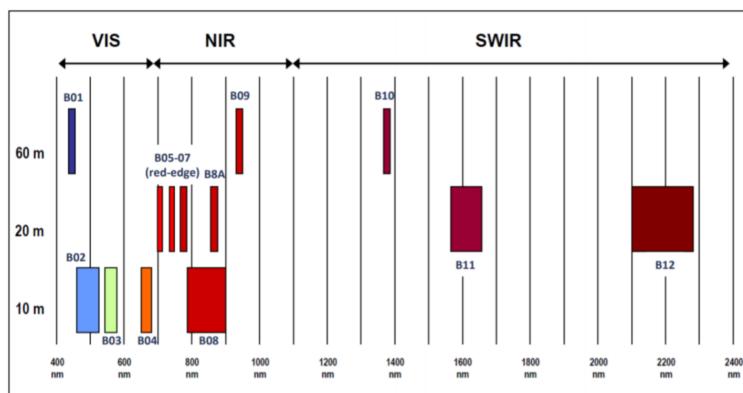


Figure 1.1 — Résolution disponible sur le télescope Sentinel-2 pour chaque longueur d'onde

Dans toute cette section, on utilisera nombre de mètres par pixel comme niveau de résolution.

Dans notre cas, et on détaillera plus loin pourquoi, nous ne nous intéresserons qu'aux longueurs d'onde B04, B05, B8A (de longueurs d'onde respectives 664.6 nm, 704.1 nm, 864.7 nm). On peut voir sur la figure 1.1 qu'elles n'ont pas toutes la même précision : il va donc falloir faire un choix quant au niveau de précision des images que l'on garde.

A priori, pour éviter toute extrapolation de données, il nous faut prendre toutes les images dans la plus faible résolution, ici 20 mètres. Seulement, nous avons eu affaire à trois problèmes majeurs :

- D'une part, à cause d'une mise à jour récente, le code de téléchargement de données satellites fourni ne fonctionne pas et ce n'est que trop tard que l'on a pu obtenir un code fonctionnel. Ainsi, notre seul moyen d'obtenir des images est de les télécharger à la main, un jour à la fois depuis le site Copernicus.
- D'autre part, des images déjà téléchargées nous ont été fournies pour les longueurs d'onde B04, B05, B8A et sont annoncées avec une résolution de 20 mètres. Or, lorsqu'on les place à côté d'images que nous avons téléchargées avec des résolutions de 10 et 20 mètres, il apparaît clairement qu'elles ne sont pas à la résolution annoncée (voir figure 1.2).
- Enfin, même si nous avons téléchargé la plupart des images disponibles à la main pour les années 2022 et 2023, elles sont en format JPEG. Or, celles fournies sont en JPEG2000 et nos codes permettent seulement de traiter ce format. Cela rend inexploitables les images que nous avons téléchargées.

Face à toutes ces complications, nous ne travaillerons qu'avec les images JPEG2000 qui nous ont été fournies. Il est donc important de garder en tête que nous n'avons aucune idée de la fiabilité de ces données, alors que toutes nos analyses reposent dessus.

Cela étant dit, les couleurs globales des images fournies sont cohérentes avec celles que nous avons téléchargées.

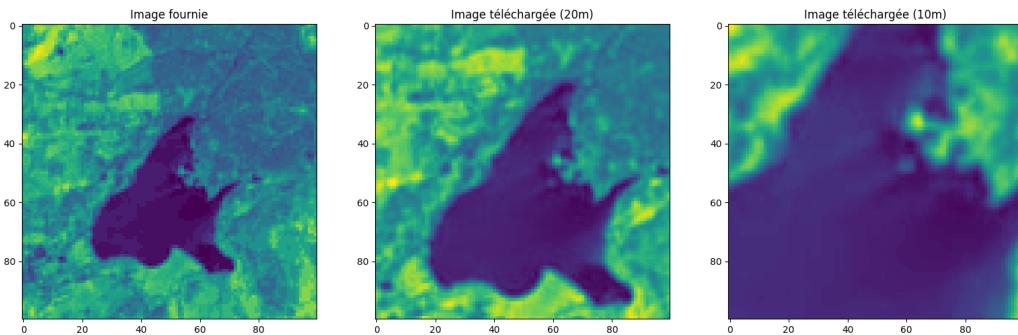


Figure 1.2 — Comparaison entre une image fournie et deux images téléchargées de résolutions respectives 20 et 10 mètres. Toutes ont été prises à la date du 24/06/2023

En particulier sur la bande B8A, les nuages apparaissent toujours en blanc et le lac toujours en noir. Le traitement potentiel des données fournies n'affecte donc pas la pertinence et l'efficacité de notre méthode.

Pour ce qui est du contenu des images, une analyse grossière a suffit à nous convaincre qu'elles ont toutes été prises avec un même cadrage. C'est-à-dire que le lac a toujours la même orientation sur les images et qu'il est toujours placé au même endroit. Par sécurité, on admettra simplement qu'elles ont été prises avec la même orientation et on verra que cela suffit pour que notre méthode soit efficace.

Enfin, comme nous pouvons le voir sur les figures 1.3 et 1.4, la bande B8A permet de mieux distinguer le lac du reste de l'image. Il apparaît toujours sombre tandis que le reste apparaît en nuances de gris : c'est sur cette information que se base la prochaine section.

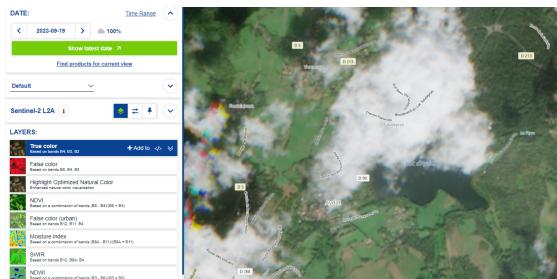


Figure 1.3 — Copie d'écran d'une image du lac depuis Copernicus



Figure 1.4 — Copie d'écran d'une image du lac pour la longueur d'onde B8A depuis Copernicus

2. Crédit et placement du masque

Ici, nous cherchons à produire un masque binaire pour chaque jour (1 sur le lac et 0 ailleurs) de même taille que les images fournies, à savoir 100×100 pixels.

L'approche est la suivante :

- Produire un premier masque du lac ajusté à la taille du lac. Dans notre cas, 56×50 pixels.
- Agrandir ce masque pour qu'il soit de même taille que les images (100×100 pixels).
- Calculer, par une méthode que nous allons détailler, la translation à effectuer sur ce nouveau masque pour que les lacs sur l'image B8A et le masque correspondent.
- Avec une fonction, faire ce calcul et enregistrer ce masque pour tous les jours où nous avons des images. Puis, les ranger dans des dossiers triés par années, mois et jours.

Pour créer le premier masque qui épouse la forme du lac, nous choisissons une journée dégagée pour que le lac soit bien visible puis, aux moyens d'un filtre médian et de méthodes de morphologie mathématique,

nous avons obtenu le résultat de la figure 1.5 :

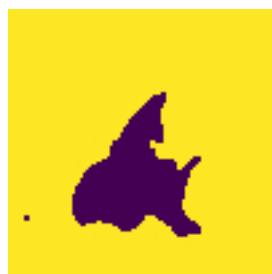


Figure 1.5 — Masque obtenu après modification
de l'image B8A à la date du 08/02/2015



Figure 1.6 — Masque tronqué

Cela demande un traitement spécifique à l'image choisie mais ce n'est pas un problème puisqu'il n'y a qu'un seul masque à réaliser. Il ne reste alors plus qu'à tronquer l'image pour avoir uniquement le lac, ce qui nous donne l'image 1.6 (*voir Python/MakeMask.py sur GitHub pour le code exact*)¹. Pour mettre ce masque à la taille de l'autre image, il suffit d'ajouter des zéros (voir ligne 4-9 du code 1).

Pour déterminer la translation qui place le masque du lac sur les images, nous avons considéré trois méthodes.

D'abord, la recherche de points d'intérêt. Il existe des algorithmes (type Harris corner detector) permettant de trouver ces points sur une image en se basant sur des fortes variations dans les valeurs de l'image. Comme nous pouvons le voir sur la bande B8A, le lac est bien distinct du sol : l'idée est donc de calculer les points d'intérêt sur une image B8A et sur le masque redimensionné, puis de calculer la translation qui superpose les points d'intérêt des deux images.

Le problème d'une telle méthode est qu'elle est très peu robuste à la présence de nuages. En effet, les nuages créent des contrastes que l'algorithme détecte comme de nouveaux points d'intérêt et en cache d'autres, ce qui pose problème pour le calcul de la translation.

Ensuite, nous avons tenté d'utiliser des méthodes de flux optique pour recadrer toutes les images par rapport à une image propre de référence – sans passer par le masque donc – avant d'ajuster à la main le masque à l'image de référence.

Le problème de ces méthodes et qu'elles calculent en chaque pixel un vecteur suivant lequel il faut le déplacer. Cela a tendance à rendre flou les images et il n'était pas envisageable de faire de telles modifications aux images satellites. Nous avons tenté de faire une moyenne de ces vecteurs pour n'avoir qu'une translation mais nous n'avons pas pu trouver des paramètres donnant des résultats satisfaisants.

Enfin, nous avons utilisé la méthode dite de cross-corrélation. Elle permet de détecter des ressemblances entre deux images et détermine la translation qui superpose au mieux ces ressemblances (voir Annexe A plus de détail). En l'appliquant à une image et à notre masque, nous obtenons le vecteur de translation **shift** (l. 24 du code 1) qui place notre masque sur le lac de l'image. Le résultat donne le masque à la taille de l'image (l. 25) et l'image avec le masque appliqué (l. 26) :

```

1 def NettoyageImg(img, mask, fullinfo=False):
2
3     # recuperation des tailles
4     h, w = img.shape
5     n, p = mask.shape
6
7     # ajustement du masque a l'image
8     mask = np.concatenate((mask, np.zeros((h - n, p))), axis=0) # (h,p)
9     mask = np.concatenate((mask, np.zeros((h, w - p))), axis=1) # (h,w)
10
11    # Obtention de la translation par Fourier
12
13    # passage dans Fourier

```

¹https://github.com/rejanej/INRAE_MIX_2023

```

14     imgFT = np.fft.fft2(img)
15     maskFT = np.fft.fft2(mask)
16
17     # Cross Power Spectrum et Fourier inverse
18     Cross = (imgFT * maskFT.conjugate()) / np.abs(imgFT * maskFT.conjugate())
19     invCross = np.fft.ifft2(Cross)
20
21     # Recuperation du shift
22
23     # translation (shift) pour placer le masque au bon endroit
24     shift = np.unravel_index(np.argmax(np.abs(invCross)), invCross.shape)
25     # masque apres translation
26     translated = np.roll(mask, shift=shift, axis=(0, 1))
27     # application du masque a l'image
28     clear = img * translated
29
30     # Returns en fonction des besoins
31
32     if fullinfo == False:
33         return clear
34     else:
35         return shift, translated, clear

```

Code 1 — Fonction de génération des masques du fichier Python/CalculMask.py

Il est à noter que cette méthode ne permet pas de tenir compte des potentielles rotations et homothéties. Ce n'est pas un problème sachant les hypothèses faites sur les images (cf. fin de la section précédente). De par son fonctionnement, la cross-corrélation sera d'autant plus efficace si le lac est bien distinct de l'image, ce qui est le cas pour les images B8A.

Cette fonction est avant tout pensée pour renvoyer le masque `mask` appliqué à l'image `img`, mais il s'est avéré plus pertinent de conserver le masque ajusté plutôt que de modifier l'image. Dans cette optique nous avons ajouté le paramètre `fullinfo` qui permet de retourner `shift` la translation à effectuer, `translated` le masque correctement placé et `clear` l'image masquée.

Dans la figure 1.7 ci-dessous se trouvent quatre exemples d'images de plus en plus nuageuses avec en dessous la position du masque calculée par la fonction en transparence et le résultat une fois le masque appliquée.

On peut voir que le procédé est robuste à la présence de nuage, même lorsque le lac est partiellement couvert. Bien entendu, la méthode échoue lorsqu'il y a trop de nuage, ce qui n'est pas problématique puisque notre méthode de détection de nuage discrimine très bien ces cas pathologiques. Méthode, que nous allons maintenant détailler.

3. Le problème des nuages

Dans les images fournies, nous disposons de trois bandes spectrales 1.1. La bande B04 (664.6 nm) correspond au rouge : cette longueur d'onde est intéressante car les végétaux, étant principalement verts (présence de chlorophylle), absorbent le rouge. La deuxième bande est B05 (704.1 nm) qui se situe dans une partie du spectre proche des infrarouges appelée "red edge". Cette portion a la particularité de présenter un changement rapide de réflectance de la chlorophylle. En effet, la végétation absorbe la plupart de la lumière dans le spectre visible mais réfléchit fortement dans les longueurs d'onde supérieures à 700 nm. Cette bande est donc très pertinente dans la recherche de végétation au sein d'images satellites. Enfin, la dernière bande est B8A (864.7 nm) qui correspond au proche infrarouge . Cette longueur d'onde présente la particularité d'être très fortement réfléchie par les couches nuageuses et peu par la végétation, d'où son utilité dans nos travaux.

Notre but est de créer un masque permettant de supprimer les potentiels nuages présents dans les dif-

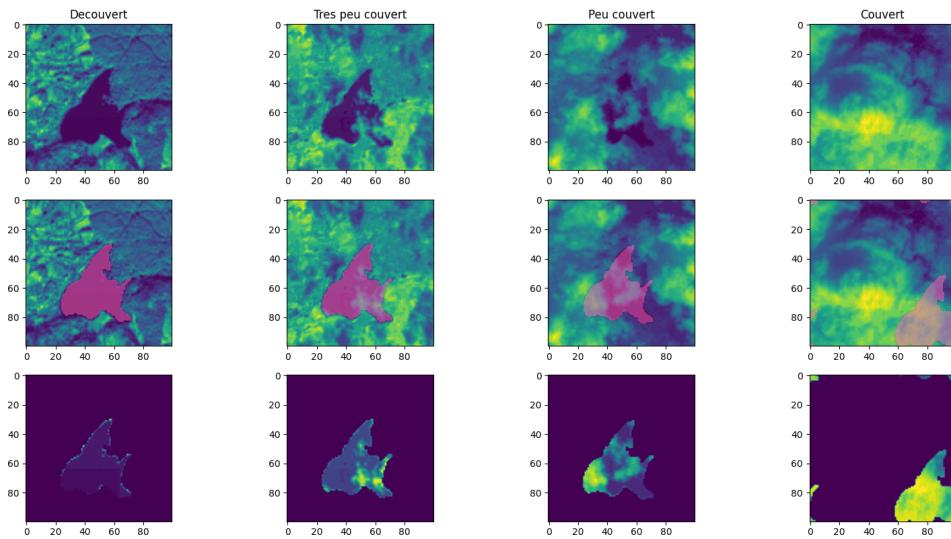


Figure 1.7 — Exemple de détection du lac sur différents niveaux de couverture nuageuse

férentes bandes spectrales, déjà filtrées par le masque précédemment établi et délimitant les contours du lac.

Après la visualisation des différentes images fournies, nous avons pu remarquer que les nuages sont principalement détectés dans la bande B8A pour les raisons décrites ci-dessus. Cependant, préférant garder toutes les informations présentes dans les images par précaution, nous avons choisi de prendre en compte les trois bandes en les sommant. Ici, il est important de noter que les images fournies doivent être au format JPEG2000. En effet, pour éviter des soucis de clipping avec des valeurs dépassant 255 avec une image JPEG ou PNG, nous préférons travailler avec des formats JPEG2000.

Par la suite, nous avons entamé un travail de visualisation d'histogrammes sur cette somme d'images masquées par le masque délimitant le lac pour comprendre ce que nous visualisons. C'est alors que nous avons pu constater que les histogrammes pouvaient généralement être distingués en trois catégories :

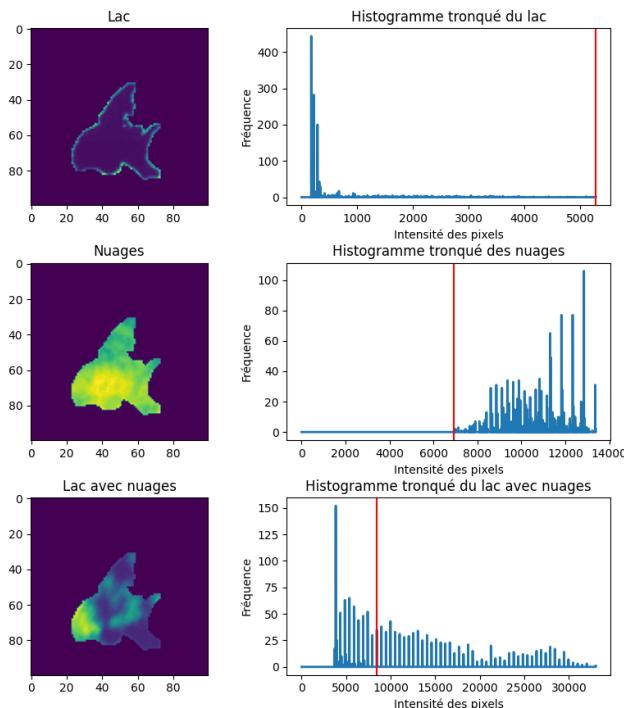


Figure 1.8 — Représentation du lac plus ou moins couvert associé à son histogramme tronqué pour supprimer les potentiels nuages

Dates : lac (15/10/2017), nuages (28/08/2023), lac avec les nuages (25/08/2023)

La partie de l'image représentant le lac ne dépasse jamais la valeur 10 000. De plus, pour les nuages très épais (qui réfléchissent beaucoup d'infrarouges) comme les cumulonimbus, les valeurs dépassent les 10 000. Dans le cas où l'on peut observer la présence du lac et des nuages, les histogrammes se présentent toujours sous forme bimodale. Nous avons donc le lac sur la gauche et les nuages sur la droite. Cette approche ne permettra donc pas le traitement d'images affectées par un fin voile nuageux ou par de la brume car les valeurs des différentes bandes en sont impactées. On définit alors une fonction *seuil* qui, à partir de l'image somme masquée, établit la disjonction de cas et coupe l'histogramme à la valeur adéquate.

```

1 def seuil(image):
2     """
3     :param
4         image (ndarray): image correspondant a la somme des bandes B04, B05, B8A
5     :return
6         seuil (int): seuillage pour le troncage de l'histogramme
7         mask_sans_nuage (ndarray): masque de l'image sans les nuages
8     """
9     # Calculer l'histogramme
10    hist, bins_center = exposure.histogram(image)
11    # Enlever la premiere valeur (0) de l'histogramme qui correspond au background
12    hist = hist[1:]
13    bins_center = bins_center[1:]
14
15    # Disjonction de cas
16    sup = len(hist)
17    if sup > 10000:
18        # Presence de nuages
19        max = np.argmax(hist)
20        if max > sup / 2:
21            # Seulement des nuages
22            seuil = np.nonzero(hist)[0][0] # Premiere valeur non-nulle
23            print(f'Seulement des nuages : {seuil}')
24        else:
25            # Nuage et lac : histogramme bimodal
26            seuil = threshold_otsu(image) # Application de la methode d'Otsu (seuillage
27            automatique)
28            if seuil < 10:
29                # Cas ou les nuages sont trop fins pour etre detectes par Otsu qui seuille a
30                zero
31                seuil = sup # Garder tous les nuages car ils sont negligables
32            print(f'Nuages et lac visibles : {seuil}')
33        else:
34            # Seulement le lac Aydat
35            seuil = sup # Garder toute l'image : pas de troncage
36            print(f'Uniquement lac : {seuil}')
37
38    # Appliquer le masque sans nuages a notre image : garder l'image sans le background
39    # intersectee avec l'image tronquée
40    image_ss_nuage = np.where((image > 0) & (image <= seuil + 1), image, 0)
41
42    return seuil, mask_sans_nuage

```

Code 2 — Fonction qui récupère le seuil et le masque sans nuages

A la ligne 20, nous utilisons l'hypothèse (après observations des différents histogrammes) que dans le cas bimodal (nuage et lac), le nombre maximal de pixels représentant le lac est toujours plus important que le nombre maximal de pixels représentant les nuages. En effet, la couche nuageuse étant beaucoup plus nuancée que la surface du lac, la partie “nuage” dans l'histogramme est donc plus étalée que la partie “lac”. On utilise alors la méthode d’Otsu (seuillage automatique) pour déterminer le seuil optimal qui sépare les deux classes “nuage” et “lac”.

Il est important de noter que ce seuillage ne coupe pas d'informations relatives à la présence de chlorophylle. En effet, nous avons observé plusieurs images et avons remarqué que l'intensité des pixels représentant de la chlorophylle ne dépasse jamais la barrière des 10 000.

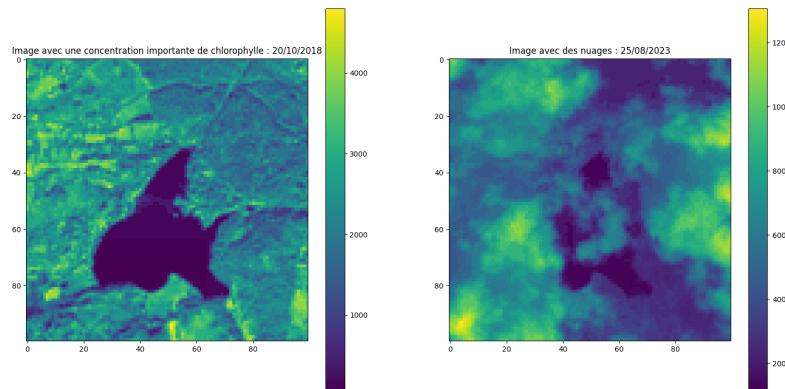


Figure 1.9 — Comparaison de l'intensité des pixels entre une image avec un taux de chlorophylle très important et une image nuageuse

On extraie donc un masque grâce au seuillage qui représente la partie de l'image ne présentant pas ou peu de nuage.

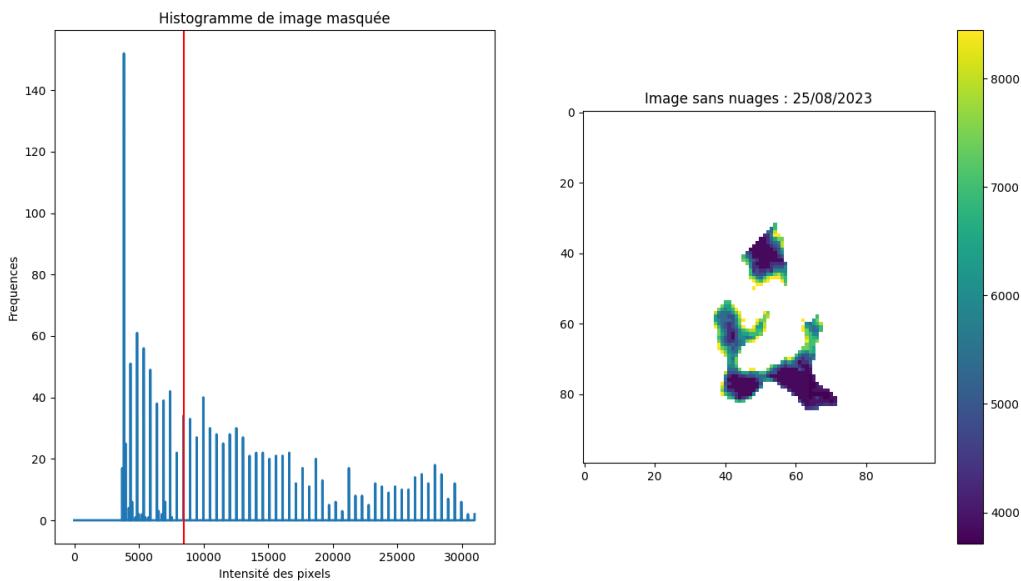


Figure 1.10 — Masquage d'une image après seuillage pour supprimer les nuages

4. Concentration de Chlorophylle-a

Comme précédemment expliqué ci-dessus, la végétation absorbe le rouge (B04) et les nuages réfléchissent les infrarouges (B8A). Ils laissent donc passer la bande B04.

On peut alors calculer plusieurs indices de réflectance. Ces indices permettent de décrire l'état d'un milieu observé à travers plusieurs bandes spectrales. Ils se basent généralement sur des combinaisons (différence, rapport, etc...) de réflectance dans les différentes longueurs d'onde. Ils traduisent alors des surfaces de

natures différentes mais l'indice en lui-même ne permet cependant pas de représenter des mesures concrètes comme la concentration de chlorophylle-a dans notre cas. Ils sont malgré tout très utiles car on peut les comparer avec les mesures in-situ (bouée, balise) et effectuer une régression pour déterminer des formules afin d'estimer des mesures et des paramètres biophysiques comme la concentration de chlorophylle-a ou la profondeur d'une étendue d'eau.

On essaie alors plusieurs indicateurs de réflectance : le Maximum Peak Height (MPH) spécialisé dans la détection de la végétation qui est utilisé dans le rapport de stage fourni, et le Normalized Difference Chlorophyll Index (NDCI) spécialisé dans la détection de chlorophylle-a dans des étendues d'eau :

$$MPH = B5 - B4 - \frac{(B8A - B4) \times (\lambda_{B5} - \lambda_{B4})}{\lambda_{B8A} - \lambda_{B4}}$$

$$NDCI = \frac{B5 - B4}{B5 + B4}$$

où $B4$, $B5$ et $B8A$ sont les réflectances des bandes considérées et où λ_{B4} , λ_{B5} et λ_{B8A} en sont les longueurs d'onde centrales.

Pour le calcul de la concentration de chlorophylle-a, nous avons utilisé la formule empirique basée sur le MPH utilisée dans le code du stage. Cette formule a été déterminée par régression sur des lacs en Lithuanie [5]. Elle a l'avantage d'être basée sur les données de plusieurs lacs conférant ainsi une généralisabilité plus large par rapport à une formule basée sur un seul lac. Elle ne garantit cependant pas la fiabilité des résultats que l'on peut obtenir sur le lac d'Aydat. Dans l'optique de se rapprocher des résultats présentés dans le rapport de stage (avec l'indice MPH), nous avons donc déterminé une formule empirique de concentration de chlorophylle-a à partir du NDCI par régression polynomiale en prenant cette concentration selon l'indice MPH (Chla-mph) sur l'image du 20/10/2018 comme valeur de référence. Ce jour est intéressant car le lac est assez nuancé en terme de concentration; la régression en sera d'autant plus efficace. On ne pouvait pas prendre les valeurs mesurées par la bouée comme référence par manque d'information (panne de la bouée en 2019 et présence d'une seule bouée dans le lac). Il aurait également été judicieux d'appliquer cette régression sur l'ensemble des images masquées sans nuage pour obtenir une formule bien généralisable au lac d'Aydat.

On peut aussi calculer l'absorption de la chlorophylle-a uniquement grâce à la bande B04 ($IndiceB4 = 1/B4$) en utilisant la formule (11) de l'article [6] où le coefficient B a été déterminé par régression sur des mesures in-situ dans une partie d'un lac en Amérique du Nord. On retrouve ainsi un résultat cohérent qui correspond bien à l'absorption de celle-ci.

Un dernier indice, Ocean Chlorophyll 2-band algorithm (OC2V4) [8], évalue la concentration de chlorophylle-a en se basant sur les bandes B04 et B05 :

$$OC2V4 = \log \left(\frac{B5}{B4} \right)$$

Dans sa construction, il se base également sur une évaluation de la luminosité. On s'en servira donc plus bas pour estimer la quantité de lumière. On peut remarquer sur la figure ci-dessous qu'il est relativement efficace. Nous l'avons également recalibré (par régression) sur les données du lac d'Aydat pour obtenir les bonnes échelles de concentration.

Nous avons également trouvé des formules de régression pour déterminer la population de cyanobactéries (en cellules par millilitre) [7] qui utilisaient le Normalized difference water index (NDWI) et le Normalized Soil Moisture Index (NSMI). Cependant ces indices nécessitaient les bandes B02, B03 et B08 que nous n'avions pas. Nous avons tenté de les télécharger nous-mêmes mais nous ne sommes pas parvenus à obtenir des résultats cohérents à cause d'une part des coefficients polynomiaux peu généralisables à notre lac et d'autre part d'une différence de résolution entre les nouvelles images téléchargées (en JPG) et les images fournies (en JPG2000). Il serait là aussi intéressant d'effectuer une régression propre au lac d'Aydat pour obtenir des indices de population convenables.

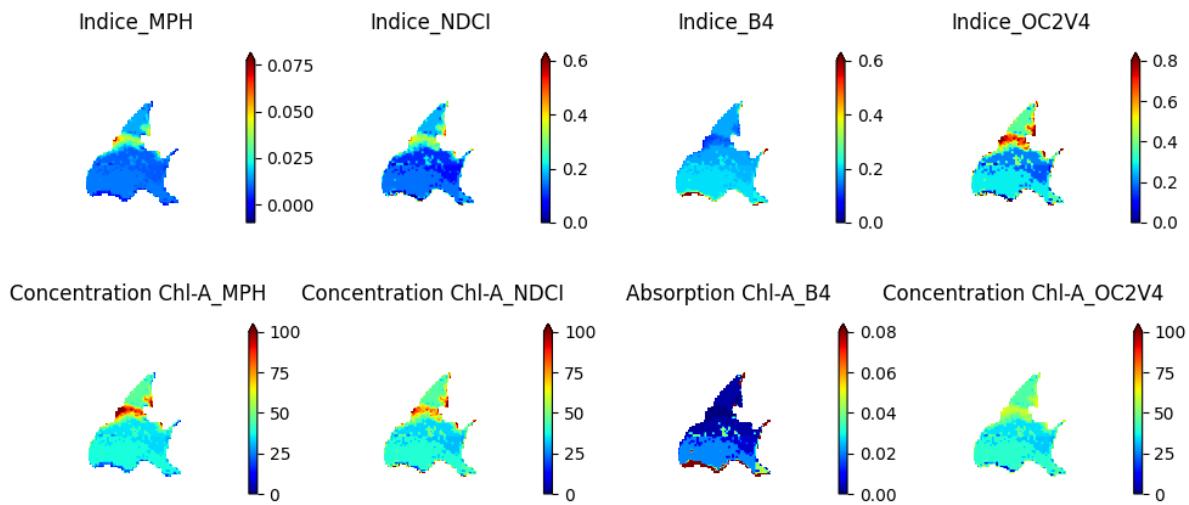


Figure 1.11 — Comparaison des différents indices de réflectance et de leur concentration en Chlorophylle-a associée

Avec nos résultats, nous pouvons remarquer que la détermination des coefficients est une étape importante pour garantir des résultats cohérents et proches des mesures in-situ. Si l'on voulait généraliser nos résultats, il faudrait effectuer une régression pour chaque zone étudiée en fonction des valeurs mesurées sur place.

5. Comparaison avec la bathymétrie

Dans le cadre de notre recherche, nous examinons l'hypothèse d'une corrélation entre la concentration en chlorophylle-a et la profondeur du lac. Pour étudier cette relation nous utilisons comme seule source de données bathymétriques, une image illustrant les lignes de niveaux (voir figure 1.12 ci-dessous).



Figure 1.12 — Lignes de niveaux de la bathymétrie du lac

Figure 1.13 — Color Map de la bathymétrie calculée

N'étant pas en mesure d'identifier l'origine de cette image, nous avons trouvé une alternative permettant de calculer la bathymétrie à partir d'images satellites (voir [2] pour plus de détail). Pour l'obtenir, nous avons utilisée la formule présente dans le script en au haut de la figure 1.13 ci-dessus, conformément à l'article [2]. Les coefficients m_0 et m_1 ont été déterminés à l'aide d'une régression, en exploitant la condition que la profondeur est nulle au bord du lac et en tenant compte, comme le montrent les lignes de niveau de la figure 1.12, de la profondeur maximale estimée du lac qui est de 16 mètres environ. Comme nous pouvons

le voir dans le script de l'image 1.13, le calcul nécessite une information sur les longueurs d'onde B02 et B03. Cependant, nous n'avons pas accès à ces données (voir section 1.) et c'est pour cette raison que nous n'avons pas poursuivi les recherches, bien que les résultats soient prometteurs.

Par la suite, nous avons repris notre masque du lac 1.5 et nous y avons ajouté trois niveaux de couleur correspondant à trois niveaux de profondeurs. Le masque obtenu (figure 1.14 ci-dessous) présente en noir l'extérieur du lac, en gris foncé les zones peu profondes (0-7 mètres), en gris clair les zones intermédiaires (7-12 mètres) et en blanc les zones les plus profondes (12-16 mètres) :



Figure 1.14 — Masque avec les trois niveaux de bathymétrie

Ce masque est approximatif car, d'une part, il est de très faible résolution (56×50 pixels) et d'autre part car il a été réalisé à la main en suivant au mieux l'image 1.12. Pour des analyses plus précises, un découpage aussi grossier n'est pas suffisant, cependant il reste convenable pour une étude préliminaire.

Afin de visualiser les résultats, nous avons initialement développé un programme générant une color map qui représente la concentration en chlorophylle-a. Celle-ci est agrandie de manière à permettre la superposition des lignes de niveau de la bathymétrie. Cette affichage est disponible sur GitHub dans le fichier `Python/Bathymetrie.py` et nous obtenons des résultats de la forme :

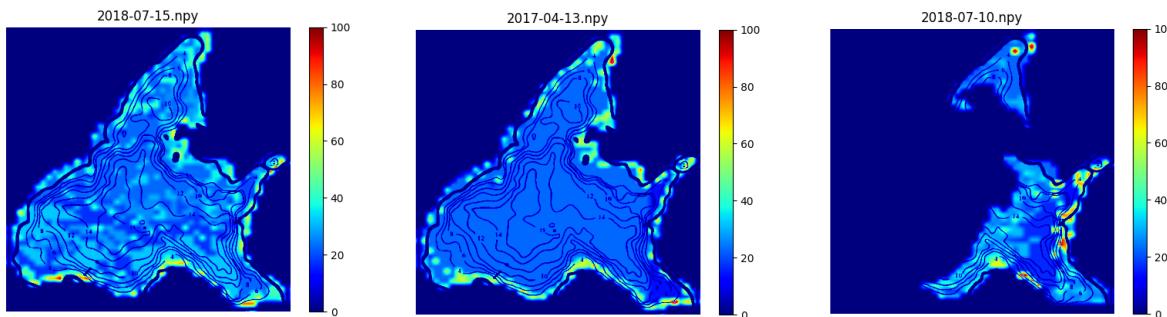


Figure 1.15 — Carte bathymétrique superposée à la Color map de concentration Chl-a

Pour obtenir de tels résultats, nous avons dû traiter l'image de la bathymétrie 1.12 et les images montrant le taux de chlorophylle-a.

Nous avons modifié l'image de la bathymétrie pour la transformer en une image de masque binaire, où les lignes de niveau sont représentées en noir (0) et le fond en blanc (1). Pour se faire, nous avons appliqué une technique de seuillage binaire à l'aide de la bibliothèque OpenCV :

```

1 #lecture de l'image bathy
2 img_bathy = cv2.imread(path_bathy) #utilisation de CV2 uniquement pour la convertir en
3 binaire
4
5 #mise en binaire de la bathymetrie
6 _, img_bathy = cv2.threshold(img_bathy,127,255,cv2.THRESH_BINARY)
7 img_bathy = img_bathy[:, :, 0]/np.max(img_bathy[:, :, 0]) # normalisation et prise de l'
8 image importante

```

Code 3 — Seuillage binaire de la bathymétrie

Puis, nous avons ajusté les dimensions des images filtrées pour qu'elles correspondent à celles de l'image de la bathymétrie. En effet, les images masquées que nous avons générées, ont une résolution de 50×56 pixels tandis que notre image de bathymétrie a une résolution de 230×219 pixels.

L'objectif est d'étendre la taille de nos images masquées en utilisant la fonction de redimensionnement `resize` de la bibliothèque `skimage.transform`. Cela permettra d'adapter l'image de la bathymétrie sur nos images masquées. Il est important de noter que cette méthode entraîne un effet de flou sur notre image masquée, mais cette décision a été prise afin de superposer la bathymétrie et d'observer les résultats. Étant donné que nos images masquées présentent des imperfections par rapport à la forme exacte du lac, nous les avons alignées manuellement. Cela a été réalisé de manière à ce que la position de l'entrée et de la sortie soient la plus cohérente possible, comme illustré dans le résultat ci-dessus.

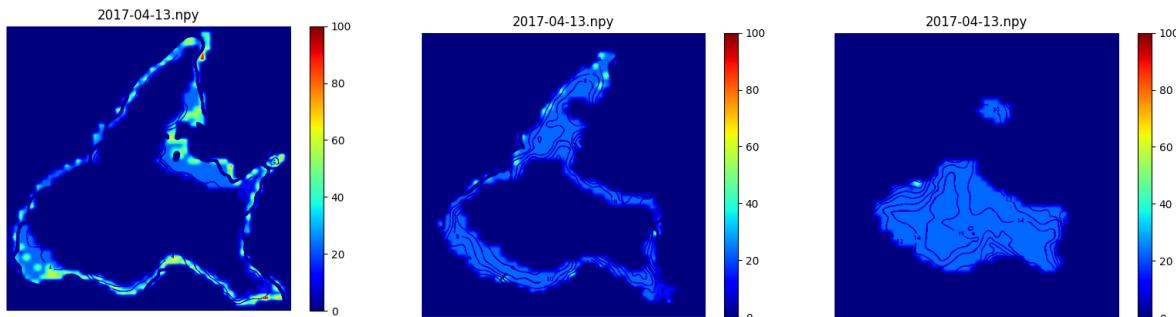


Figure 1.16 — Carte bathymétrique superposée à la color map de concentration Chl-a en fonction des niveaux de profondeurs

Après observations sur plusieurs images correspondant à différentes dates, il semble que la concentration de chlorophylle-a soit plus importante dans les zones les moins profondes du lac (ie. aux bords). On analysera plus bas ces données sur l'ensemble des dates fournies afin de déterminer une potentielle corrélation entre la profondeur du lac et la concentration.

II. Études statistiques

La gestion et l'analyse des données météorologiques occupent une place prépondérante dans le domaine de la recherche scientifique et de la prise de décision. En effet, les données météorologiques fournissent des informations cruciales pour comprendre les conditions atmosphériques, anticiper les changements climatiques, et adapter nos sociétés aux phénomènes météorologiques extrêmes. Dans le cadre de ce rapport, nous explorerons les différentes étapes du traitement des données météorologiques, mettant en lumière les méthodologies, les outils, et les résultats obtenus.

1. Présentation des données météorologiques

Nous disposons d'un fichier Excel contenant 21 feuilles, chacune contenant des données météorologiques. Dans le cadre du traitement de ces données, notre attention s'est concentrée sur les feuilles B_I, ST-GENES et VERNINES.

- **Feuille B_I:** Répertorie les données météorologiques collectées par une bouée dans le lac sur une plage de dates allant du 30/05/2018 au 24/11/2021. Les paramètres examinés incluent :

- Date (DATE : mm/jj/aaaa hh:mm),
- Température (Temp (°C)),
- Quantité de chlorophylle-a (ChloroA (µg/l) et ChloroA (V)),
- Quantité d'oxygène (O2 (%) et O2 (Mg/l)),
- Quantité de phycoérythrine (Phyco (µg/l) et Phyco (V)),
- Conductivité (SpConductivité (µS/cm)),
- Tension (Tension (V)),
- Tension de la sonde (TensionSonde (V)).

- **Feuille ST-GENES:** Répertorie les données météorologiques collectées dans la commune de Saint-Génès-Champanelle (située à 8 km du lac d'Aydat) sur une plage de dates allant du 01/01/2010 au 25/03/2022. Les paramètres examinés incluent :

- Date (DATE : mm/jj/aaaa hh:mm),
- Hauteur des précipitations horaire en millimètres (RR1),
- Température sous abri horaire (T : °C),
- Température du point de rosée horaire (TD : °C),
- Vitesse du vent à 2 mètres horaire (FF2 : m/s),
- Humidité relative horaire (U : %),
- Rayonnement global horaire (GLO : Joules/cm²).

- **Feuille VERNINES:** Répertorie les données météorologiques collectées dans la commune de Vernines (située à 11 km du lac d'Aydat) sur une plage de dates allant du 01/01/2010 au 25/03/2022. Les paramètres examinés incluent :

- Date (DATE : mm/jj/aaaa hh:mm),
- Hauteur des précipitations horaire en millimètres(RR1),
- Épaisseur de neige totale horaire (NEIGETOT : cm),
- Température sous abri horaire (T : °C),

- Température du point de rosée horaire (TD : °C),
- Température minimale sous abri horaire (TN : °C),
- Température maximale sous abri horaire (TX : °C),
- Durée du gel horaire (DG : minutes),
- Vitesse du vent horaire (FF : m/s),
- Vitesse du vent instantané maxi horaire (FXI : m/s),
- Direction du vent maxi instantané horaire (DXI : ROSE de 360),
- Humidité relative mini horaire (UN : %)
- Heure de l'humidité relative mnimale horaire (HUN)
- Humidité relative maxi horaire (UX: %)
- Heure de l'humidité relative maximale horaire (HUX : m/s)

De plus, toutes les feuilles ne présentent pas des données sur la même plage de date, et certaines sont manquantes sur plusieurs mois pour l'année 2018 de la **Feuille B_I**, ainsi que pour l'année 2021 de cette même feuille. La mise en correspondance des données de toutes ces feuilles nous permet donc de visualiser les informations météorologiques sur la plage de date du 30/05/2018 au 24/11/2021.

2. Tri et étude des données météorologiques

2.1. Gestion des valeurs aberrantes

Le traitement des données météorologiques est une étape cruciale dans l'analyse climatique, permettant de révéler des tendances, des variations et des anomalies. Dans le cadre de cette étude, nous avons appliqué la méthode des Z-scores pour normaliser et identifier les valeurs atypiques au sein de notre ensemble de données météorologiques. Cette approche statistique offre une perspective essentielle pour évaluer la variabilité des mesures climatiques et détecter les événements exceptionnels.

La Méthode des Z-Scores

La méthode des **Z-scores**, également connue sous le nom de score Z ou écart-type, est une technique statistique utilisée pour évaluer la position d'une observation par rapport à la moyenne d'une distribution et la variabilité autour de cette moyenne. Elle est particulièrement utile pour détecter les valeurs atypiques, définies comme des observations qui s'écartent significativement de la moyenne de la distribution.

Le Z-score d'une observation x dans une distribution avec une moyenne μ et un écart-type σ est calculé avec la formule suivante :

$$Z = \frac{(x - \mu)}{\sigma}$$

Le Z-score mesure le nombre d'écart-types par lesquels une observation est éloignée de la moyenne. Un Z-score positif indique que l'observation est au-dessus de la moyenne, tandis qu'un Z-score négatif indique qu'elle est en dessous. Cette méthode nous a permis de standardiser les données, les ramenant à une échelle commune et facilitant ainsi la comparaison entre différentes variables. Les observations avec des Z-scores au-delà d'un seuil prédéfini (par exemple, $|Z| > 3$) sont considérées comme des valeurs atypiques.

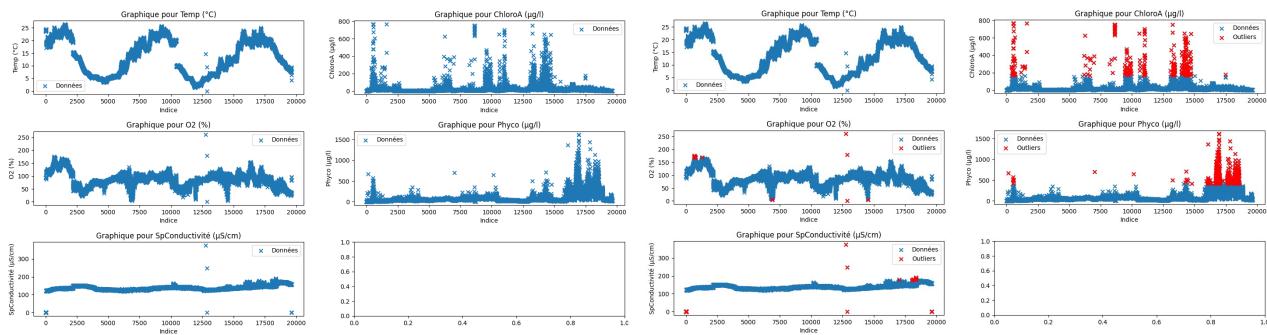


Figure 2.1 — Données représentées avec et sans observation des outliers pour les paramètres Temp, ChloroA, O2, Phyco, SpConductivité avec $Z=3$

En ce qui concerne la pertinence du choix de méthode permettant d'identifier les outliers, d'autres approches existent et ont été expérimentées (comme la méthode de l'écart inter-quartile). Les résultats obtenus étant similaires, nous avons détaillé ici une unique méthode.

2.2. Variables étudiées et conservées

Après avoir méticuleusement éliminé les valeurs aberrantes de notre ensemble de données météorologiques, la prochaine étape consiste à réduire la dimension du problème en éliminant les variables dites “inutiles”. Cette démarche vise à simplifier l’analyse tout en préservant l’essentiel de l’information. Ci-dessous, nous explorons des méthodes efficaces pour atteindre cet objectif.

-Étape 1 : Analyse de la Corrélation

L’analyse de corrélation entre les différentes variables permet de déterminer les relations linéaires ou non linéaires entre elles. Les variables fortement corrélées peuvent fournir une information redondante. En éliminant l’une des variables fortement corrélées, nous réduisons la dimension sans sacrifier significativement l’information contenue dans les données.

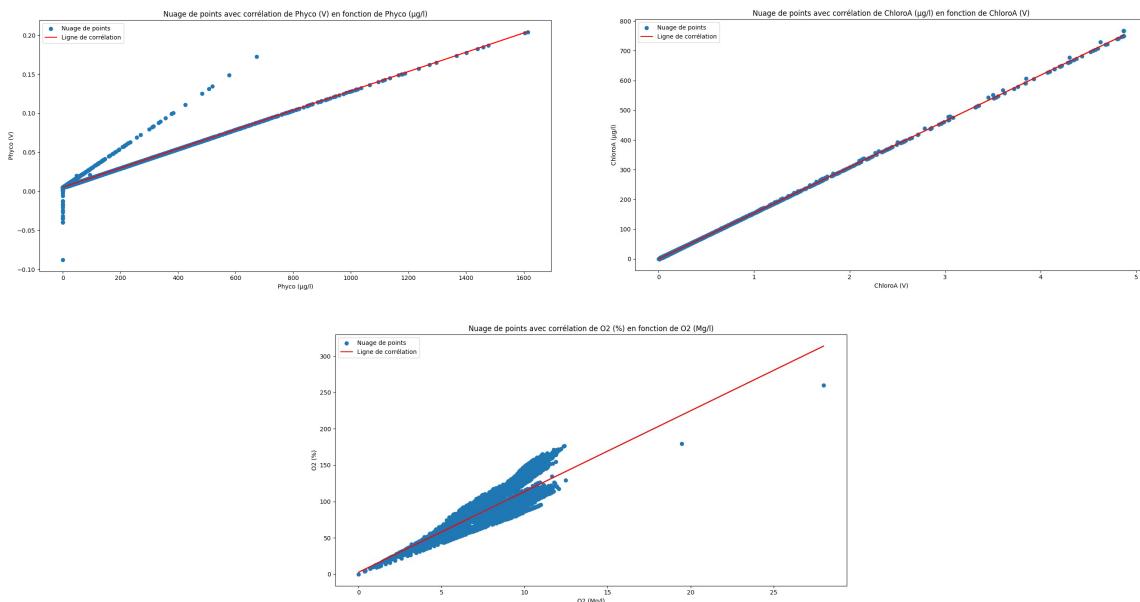


Figure 2.2 — Étude des corrélations croisant les variables redondantes

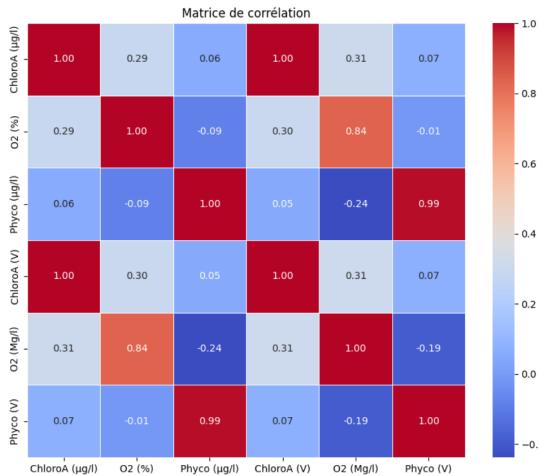


Figure 2.3 — Matrice de corrélation croisant les variables redondantes

Les figures 2.2 et 2.3 nous montrent une forte corrélation entre les variables Phyco ($\mu\text{g/l}$) et Phyco (V), ChloroA ($\mu\text{g/l}$) et ChloroA (V) et O2 (%) et O2 (Mg/l). Pour simplifier l'étude du problème et éliminer les informations redondantes, nous avons donc décidé de ne pas étudier les variables Phyco (V), ChloroA (V), O2 (Mg/l).

-Étape 2 : Analyse des Composantes Principales (PCA)

La PCA est une technique statistique et linéaire qui transforme les variables originales en un nouvel ensemble de variables, appelées composantes principales. Ces composantes sont ordonnées par ordre d'importance, la première capturant le plus grand écart de variance, la deuxième le deuxième plus grand, et ainsi de suite. En éliminant les composantes de faible variance, nous conservons les informations les plus significatives tout en réduisant la dimension du problème.

La formule pour calculer la première composante principale (PC_1) est donnée par :

$$PC_1 = w_{1,1}X_1 + w_{1,2}X_2 + \dots + w_{1,p}X_p$$

où $w_{1,1}, w_{1,2}, \dots, w_{1,p}$ sont les poids attribués à chaque variable et X_1, X_2, \dots, X_p sont les variables originales.

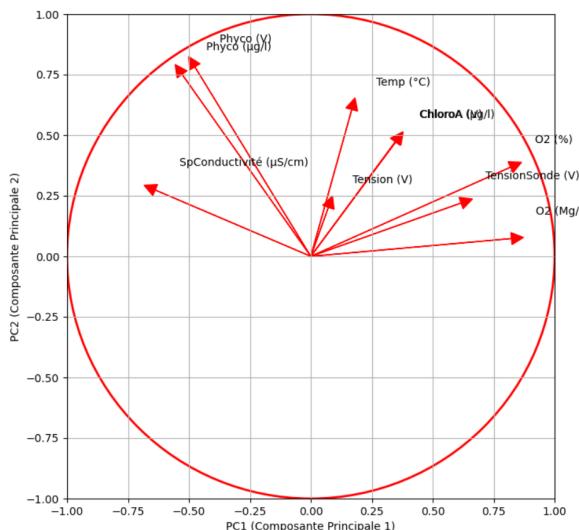


Figure 2.4 — Cercle de corrélation généré à partir de l'analyse en composantes principales (PCA)

Le cercle de corrélations ci-dessus représenté nous montre l'importance de chaque variable dans l'étude du problème. Les flèches les plus courtes représentent les variables qui contribuent le moins à la variance totale, tandis que les plus grandes (proches du cercle) sont celles qui contribuent le plus à la variance totale. On observe ainsi que les variables **TensionSonde** (V) et **Tension** (V) contribuent peu à la variance totale par rapport aux autres variables et peuvent donc être considérées comme moins importantes.

3. Recouplement des données observées et mesurées

Dans le cadre de notre étude visant à prédire la quantité de cyanobactéries présente dans le lac, une phase cruciale de l'analyse consiste à explorer les corrélations entre différentes variables. Cette approche statistique nous permet de comprendre les relations entre les divers facteurs environnementaux et la prolifération des cyanobactéries dans le lac.

3.1. Mise en relation des données météorologiques

La première étape de notre approche consistait à explorer d'éventuelles dépendances linéaires entre les variables en examinant la matrice de corrélation. Nous avons cherché à déterminer si des relations linéaires significatives pouvaient être identifiées entre chaque paire de variables, ce qui aurait permis de mettre en lumière des associations directes entre les différentes dimensions de nos données.

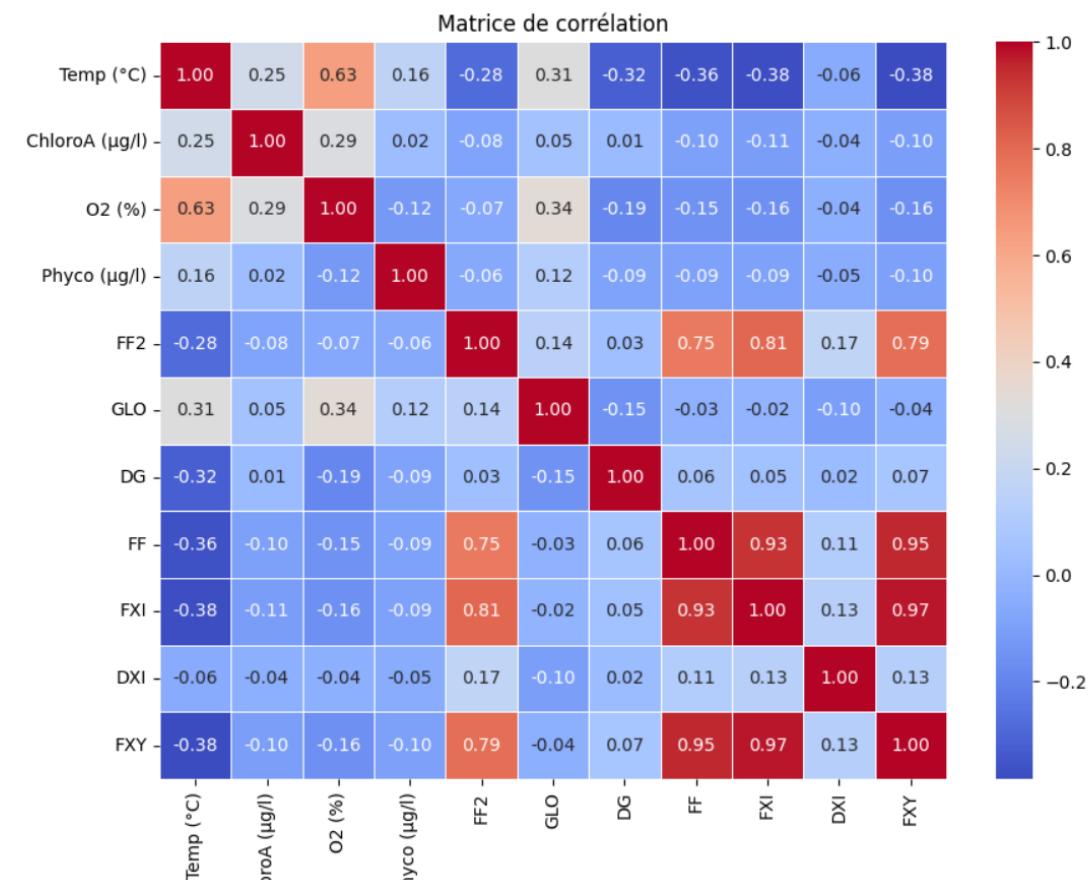


Figure 2.5 — Matrice de corrélation croisant les variables

La matrice de corrélations montre qu'il n'y a aucune corrélation significative entre les variables deux à deux, hormis quelques dépendances attendues et logiques :Temp (°C) avec O2 (%), FXY avec FF, FF2 avec FXY...

Un premier résultat inattendu concerne les observations sur les variables (ChloroA ($\mu\text{g/l}$) et (Phyco ($\mu\text{g/l}$)). La matrice de corrélation montre un score très faible de 0.02. En croisant ces deux variables sur un graphique, on se rend effectivement compte de cette indépendance, comme le montre la figure 2.6 ci-dessous :

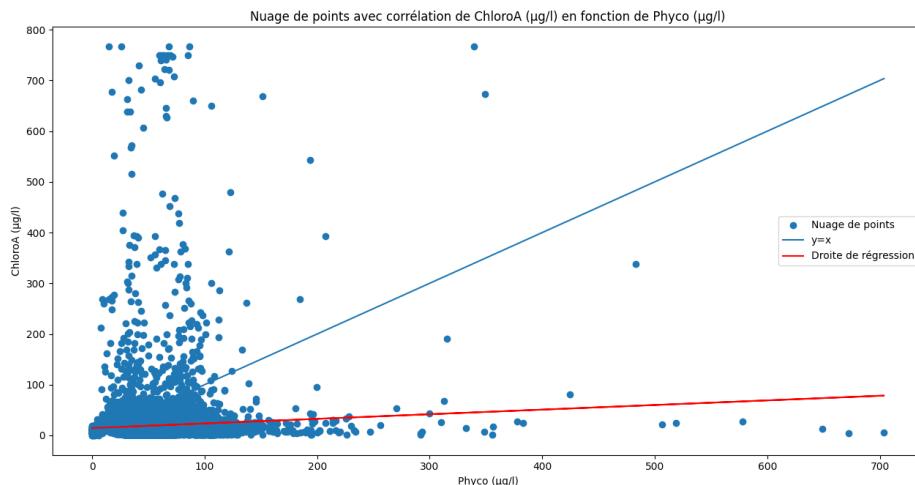


Figure 2.6 — Croisement des concentrations de ChloroA et Phyco

Les données sont très éparpillées, ce qui va à l'encontre de l'idée que les quantités de phycoérythrine et de chlorophylle-a seraient corrélées, étant donné qu'elles constituent toutes deux des pigments révélateurs de la présence de cyanobactéries.

Face à l'observation de l'absence de corrélations significatives entre les variables, nous avons pris la décision de diversifier nos approches d'analyse. L'absence de dépendances linéaires évidentes entre chaque paire de variables suggère que d'autres mécanismes et relations, potentiellement plus complexes, pourraient influencer la structure de nos données.

Dans cette optique, nous avons exploré deux approches complémentaires :

Régression OLS (Ordinary Least Squares)

Conscients que la corrélation linéaire n'est pas toujours le reflet complet des relations entre les variables, nous avons opté pour une Analyse en Moindres Carrés Ordinaires (OLS). Cette technique de régression linéaire permet d'ajuster un modèle linéaire aux données, tout en cherchant à minimiser la somme des carrés des erreurs résiduelles. La régression OLS nous offre ainsi la possibilité d'identifier des relations linéaires potentielles qui pourraient ne pas être immédiatement apparentes dans une simple analyse de corrélation.

Cross-Corrélation et Corrélations Temporelles

Étant donné que nos données météorologiques sont chronologiques, nous avons également exploré la Cross-Corrélation pour examiner les corrélations temporelles entre les variables. Cette méthode nous permet de déterminer si des motifs de corrélation se manifestent avec un décalage temporel, révélant ainsi des relations dépendantes du temps. Cette approche est particulièrement pertinente pour capturer des interactions dynamiques qui pourraient ne pas être évidentes dans une analyse statique de corrélation. Malheureusement, ces explorations n'ont pas abouti à la découverte de relations significatives ou de modèles prédictifs robustes. Cela est notamment du à la sensibilité des données. En effet, en fonction des variables conservées dans l'observation des données, les résultats d'ACP diffèrent complètement comme on peut le voir sur les figures 2.7, 2.8 et 2.9.

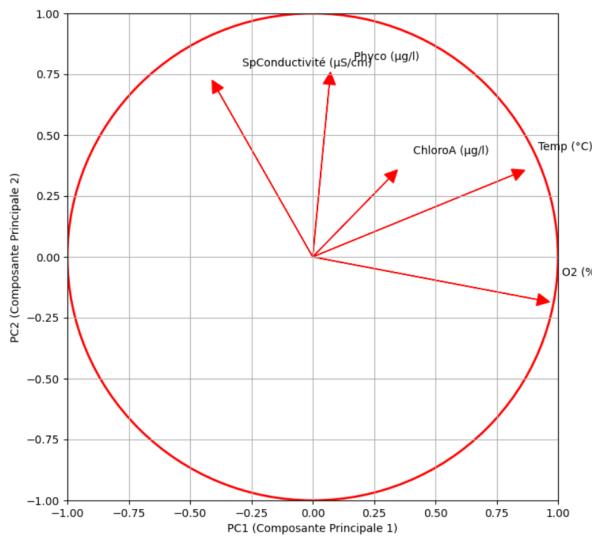


Figure 2.7 — Cercle de corrélation avec les variables SpConductivité, Phyco, ChloroA, Temp, O2

En considérant ces données, aucune dépendance ne ressort. L'ajout progressif de variables fournit des résultats diverses :

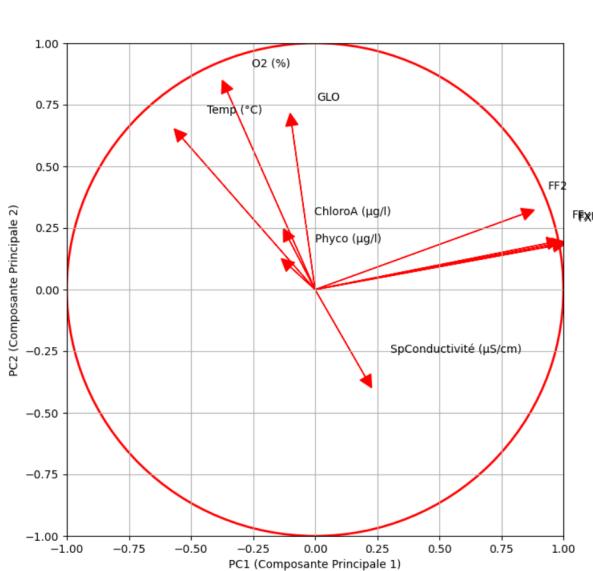


Figure 2.8 — Cercle de corrélation avec l'ajout du rayonnement global et des vitesses du vent

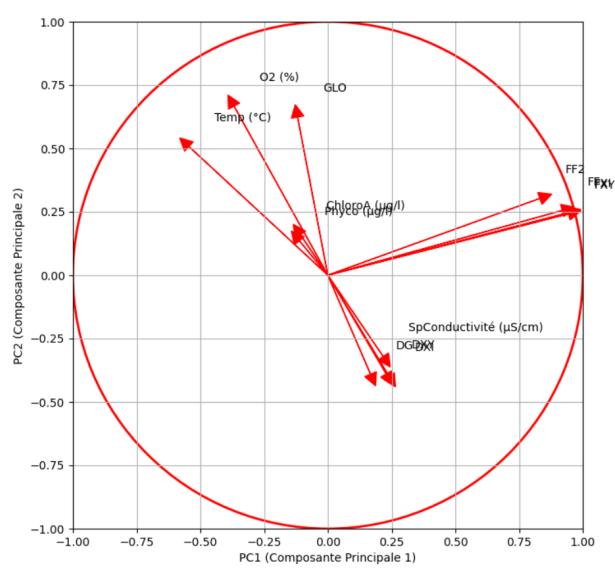


Figure 2.9 — Cercle de corrélation avec l'ajout de la durée du gel et des directions du vent

Sur cette dernière figure, on remarque tout d'abord que toutes les données concernant la vitesse du vent (FXI : m/s , FXY : m/s , FF : m/s , FF2 : m/s), à ST-GENES et VERNINES sont corrélées. On peut faire la même remarque sur les variables concernant la direction du vent. L'angle droit entre les variables de la direction du vent et de la vitesse du vent traduit une indépendance entre les variables. Les quantités de ChloroA et de Phyco semblent être inversement corrélées avec la direction du vent et la durée du gel DG. De plus ChloroA et Phyco semblent désormais être corrélées, et dépendre de la température ainsi que de la quantité d'oxygène. Ces résultats sont cohérents avec le fonctionnement des organismes : une augmentation de la température accélère le métabolisme ce qui entraîne une croissance des populations. Aussi, l'oxygène étant un produit de la photosynthèse, il n'est pas étonnant de voir sa concentration corrélée avec la chlorophylle-a. Il est plus étonnant est que cette corrélation n'apparaît pas plutôt dans les PCA.

Au vu de la sensibilité des données, ces résultats sont à considérer avec précaution, l'ajout de paramètres supplémentaires pourrait mener à des conclusions totalement différentes.

3.2. Réduction de dimension et clustering des données météorologiques

Nous avons appliqué l'algorithme UMAP (Voir Annexe B) à nos données afin de mieux comprendre la structure intrinsèque qui n'est pas évidente lors de l'observation des corrélations classiques. UMAP nous permet de représenter les relations complexes entre les variables dans un espace de dimension réduite. L'algorithme effectue la projection d'une manière intelligente de manière à conserver la structure des données. En l'occurrence, ici nos données météorologiques sont en 11 dimensions et l'algorithme UMAP nous permet de les visualiser en 2 voire 3 dimensions. L'UMAP est réalisé pour différents intervalles de temps. En revanche UMAP ne fait pas le clustering (regroupement en classe des données). Encore une fois, beaucoup d'algorithmes existent mais le plus connu est l'algorithme k-means ou bien un algorithme plus polyvalent, l'algorithme HDBSCAN (Voir Annexe C). C'est une version hiérarchique de DBSCAN qui permet de se défaire du choix du rayon des voisins.

En l'appliquant à nos données, nous obtenons la représentation 3D de la figure 2.10. On y observe la présence de quelques clusters mais la quasi-totalité des points sont considérés comme des valeurs aberrantes par HDBSCAN. Dû au fonctionnement du UMAP, il est difficile d'interpréter un tel graphe puisque les axes ne correspondent à rien de concret. Néanmoins, le fait que les clusters ne soient pas plus distincts des points aberrants laisse entendre que les données n'ont pas de structure claire.

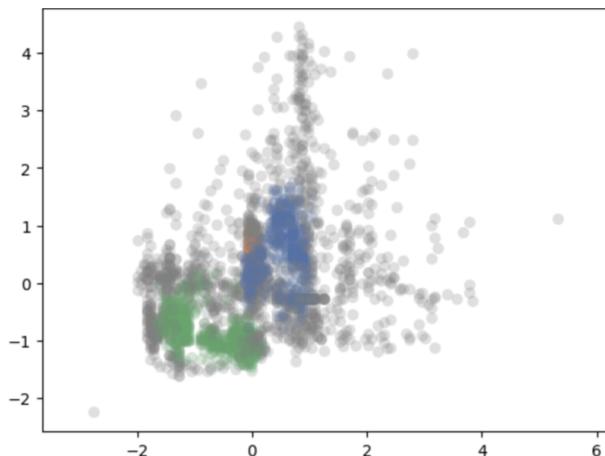


Figure 2.10 — Visualisation UMAP et clustering HDBSCAN des données

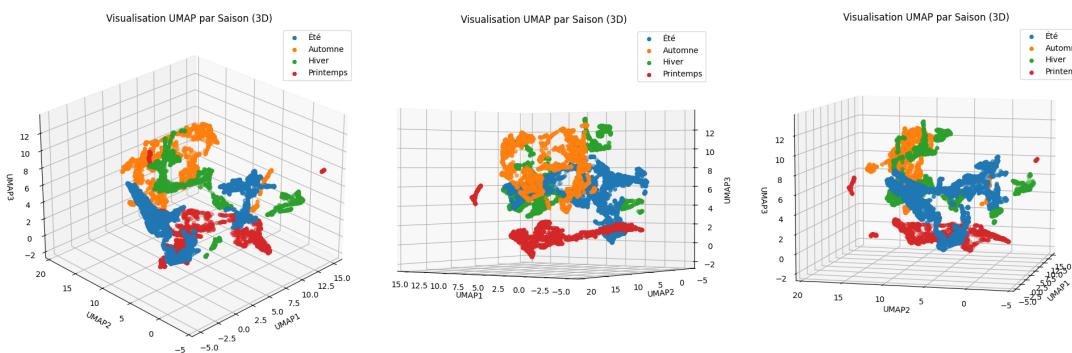


Figure 2.11 — Visualisation UMAP 3D et clustering HDBSCAN des données pour les paramètres `nb neighbor=15` et `min dist=0.1`

Les paramètres représentés ci-dessus sont la température, la concentration de chlorophylle-a, la quantité d'oxygène, la concentration de phycoérythrine, la conductivité, la vitesse du vent à Saint-Gènes, le rayonnement global horaire à Saint-Gènes, la durée du gel horaire à Vernines, la vitesse du vent horaire à Vernines, la direction du vent à Vernines ainsi que l'humidité relative à Vernines. Cet UMAP nous fournit ainsi un

espace réduit pour représenter au mieux nos 11 variables en dimension 3, de manière à y entrevoir des clusters correspondant aux saisons de l'année. Cette projection montre bien une cohérence entre les saisons au cours des ans, qui à défaut d'avoir été exploitée (dans l'optique d'énoncer une corrélation claire entre le taux de cyanobactéries dans le lac et les conditions météorologiques), existe.

3.3. Recouplement avec les données observées par satellite

Comparaison de la chlorophylle mesurée et observée

Afin d'évaluer l'efficacité des indices de concentration de chlorophylle-a calculés sur les images satellites, il est intéressant de les comparer avec les données fournies par la bouée. Dans un premier temps, il est donc nécessaire de localiser la bouée sur une image du lac (coordonnées en pixels). Ne possédant que les coordonnées géographiques, on sélectionne une fenêtre de l'image qui contient la bouée (taille 15×7 pixels). Sa taille n'est ni trop petite pour assurer la présence de la bouée et ni trop large pour éviter de fausser les résultats par rapport aux données fournies.

On construit par la suite un fichier CSV qui contient la moyenne et les valeurs minimale et maximale de la concentration de chlorophylle-a pour chaque image que nous possédons.

Ces valeurs sont déterminées à l'aide de l'indice MPH. Puis nous recoupions les données calculées avec les données fournies. On affiche les données correspondant à l'intersection des dates présentes à la fois dans les données météorologiques et satellites :

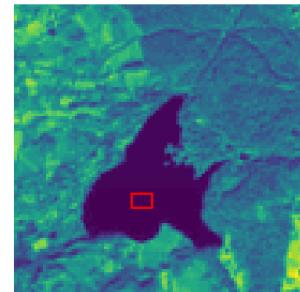


Figure 2.12 — Fenêtre de localisation de la bouée

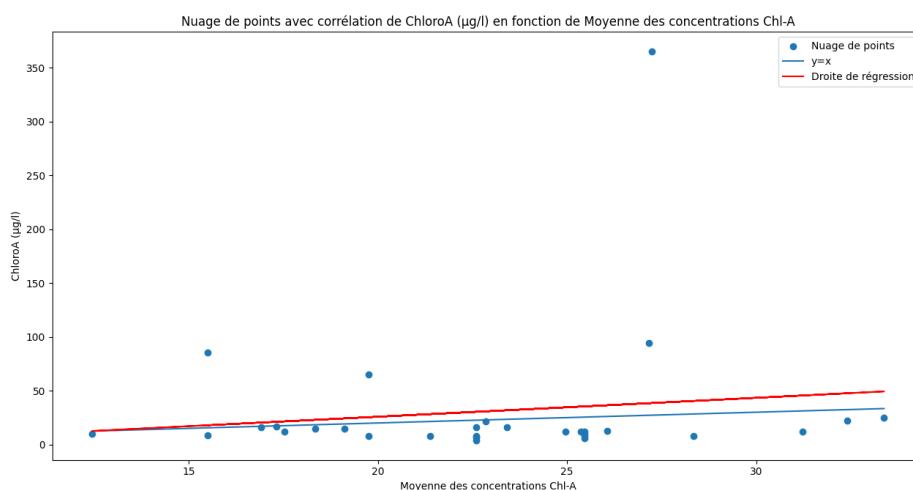


Figure 2.13 — Comparaison des concentrations mesurées et observées.

Ici, les mesures en abscisse sont celles calculées sur les images satellites et les mesures en ordonnée sont les données de la bouée. Chaque point représente une date. On remarque que la droite de régression s'approche du modèle $y = x$, ce qui traduit une cohérence entre les résultats calculés à l'aide des données satellites et les données in-situ.

Comparaison des luminosités mesurée et observée

Nous avons précédemment vu que l'indice OC2V4 permettait également d'estimer la quantité de lumière. Nous pouvons alors l'appliquer à la fenêtre correspondant à la position de la bouée et comparer les données (**Luminosité**) à celles fournies par la station météorologique de St-Gênes (**GLO**). Voir figure 2.14 ci-dessous. Notons que, les unités n'étant pas les mêmes (OC2V4 n'est qu'un indice, il ne respecte pas d'unité physique), représenter les données sous forme de plot permet de visualiser la corrélation à un facteur d'échelle près.

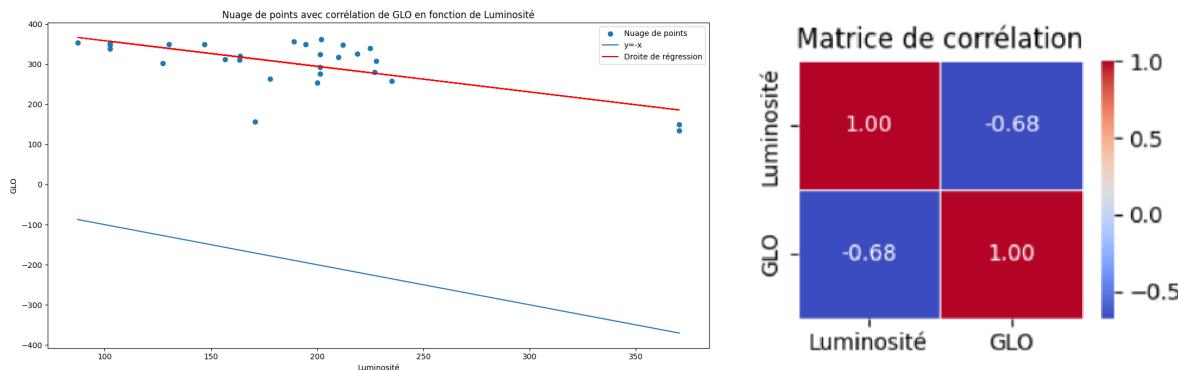


Figure 2.14 — Corrélation entre la lumière observée et mesurée

Les données sont inversement proportionnelles, ce qui se confirme grâce à la matrice de corrélations où nous obtenons un coefficient de corrélation proche de -1 . C'est un score qui se rapproche de la dépendance Temp ($^{\circ}\text{C}$) avec O2 (%).

Comparaison de la concentration en chlorophylle et de la bathymétrie mesurées et observées

Comme expliqué plus haut dans la partie concernant la bathymétrie, nous avons extrait les concentrations moyennes de chlorophylle-a pour chacune des trois zones correspondant aux profondeurs (bord : 0-7m, inter : 7-12m, centre : 12-16m). De cette manière, nous pouvons les visualiser :

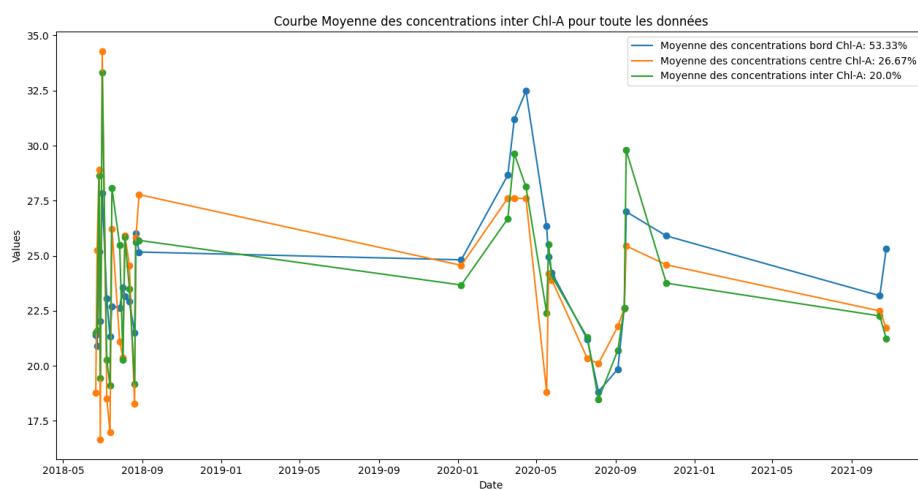


Figure 2.15 — Concentration de la chlorophylle a selon les zones du lac

On se demande si une certaine zone est plus concentrée en chlorophylle-a que les autres. Il est ici difficile de distinguer une quelconque distinction d'une zone où les valeurs sont maximales par rapport aux

autres. Cependant, en calculant la fréquence (en pourcentage) à laquelle les différentes zones présentent une concentration supérieure aux autres, nous observons que la zone caractérisant le bord est plus souvent concentrée en chlorophylle-a (53% du temps). Nous pouvons relier ce résultat au fait que la température est corrélée à la concentration de chlorophylle-a (évoqué plus haut 2.8, 2.9). En effet, les bords du lac étant moins profonds, la température est donc plus importante qu'au centre du lac. De plus, nous pouvons également supposer que le courant est moins puissant sur les bords qu'au centre. C'est pour cela que la chlorophylle-a se situe en grande quantité sur les rives du lac d'Aydat.

ANNEXES

A Cross-Corrélation

Le principe de la cross-corrélation est essentiellement le même que celui de la convolution : on considère deux images en niveaux de gris que l'on représente par les matrices $A = (a_{ij})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} \in \mathcal{M}_{n,m}(\mathbb{R})$ et $B = (b_{ij})_{\substack{1 \leq i \leq p \\ 1 \leq j \leq q}} \in \mathcal{M}_{p,q}(\mathbb{R})$. Pour éviter les problèmes de définition, on étendant ces deux matrices par des zéros et la cross-corrélation entre ces deux matrices est l'application :

$$A \star B : (u, v) \mapsto \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} a_{ij} \times b_{i+u, j+v}$$

Dans notre cas, A est l'image du lac et B le masque. L'idée derrière cette formule et que $A \star B$ ne prendra de grandes valeurs que si un grand nombre de produit $a_{ij} \times b_{i+u, j+v}$ sont non nuls, ce qui arrive lorsque le masque se superpose le mieux au lac (*sur Wikipédia se trouve une animation qui explique très bien le principe*). Il suffit alors de récupérer le vecteur (u_{\max}, v_{\max}) qui maximise $A \star B$ pour obtenir la translation qui superpose au mieux la masque sur le lac.

De façon plus générale, la cross-corrélation de deux applications est donnée par la formule :²

$$\forall f, g \in L^2(\mathbb{R}^n, \mathbb{C}), f \star g(t) = \int_{\mathbb{R}^n} \overline{f(\tau)} g(t + \tau) d\tau$$

Bien qu'elle ne soit pas exactement une convolution, elle hérite de propriétés similaires à cette dernière par rapport à la transformée de Fourier. Cela va nous permettre d'accélérer le calcul de la translation (u_{\max}, v_{\max}) : en notant \mathcal{F} la transformée de Fourier et \mathcal{F}^{-1} la transformée inverse, on a la formule :

$$\forall (u, v) \in \mathbb{Z}^2, \quad \mathcal{F}(f \star g) = \overline{\mathcal{F}(f)} \mathcal{F}(g)$$

Montrons le en dimension 1, *i.e.* avec $f, g \in L^2(\mathbb{R}, \mathbb{C})$. En s'autorisant à intervertir les intégrales et avec le changement de variable $t = u - \tau$, nous obtenons :

$$\begin{aligned} \forall \xi \in \mathbb{R}, \mathcal{F}(f \star g)(t) &= \int_{\mathbb{R}} \int_{\mathbb{R}} \overline{f(\tau)} g(t + \tau) d\tau e^{-2i\pi\xi t} dt \\ &= \int_{\mathbb{R}} \int_{\mathbb{R}} \overline{f(\tau)} g(u) e^{-2i\pi\xi(u-\tau)} du dt \\ &= \int_{\mathbb{R}} \int_{\mathbb{R}} \overline{f(\tau)} g(u) e^{-2i\pi\xi(u-\tau)} du dt \\ &= \int_{\mathbb{R}} \overline{f(\tau)} e^{2i\pi\xi\tau} d\tau \int_{\mathbb{R}} g(u) e^{-2i\pi\xi u} du \\ &= \overline{\int_{\mathbb{R}} f(\tau) e^{-2i\pi\xi\tau} d\tau} \int_{\mathbb{R}} g(u) e^{-2i\pi\xi u} du = \overline{\mathcal{F}(f)} \mathcal{F}(g) \end{aligned}$$

La même formule s'applique dans le cas discret de sorte que, en notant \odot le produit d'Hadamard, l'on ait :

$$\begin{aligned} \mathcal{F}(A \star B) &= \overline{\mathcal{F}(A)} \odot \mathcal{F}(B) \implies A \star B = \mathcal{F}^{-1}\left(\overline{\mathcal{F}(A)} \odot \mathcal{F}(B)\right) \\ &= \mathcal{F}^{-1}\left(\mathcal{F}(A) \odot \mathcal{F}(B)\right) \end{aligned}$$

² $L^2(\mathbb{R}^n, \mathbb{C})$ est l'ensemble des applications à valeur de \mathbb{R}^n dans \mathbb{C} dont le carré intégrable.

Comme cette formule demande moins de temps de calcul, c'est celle-ci que l'on applique dans notre algorithme.

B UMAP

B.1. Uniformisation des données

Comme expliqué dans la section II., UMAP (Uniforme Manifold Approximation and Projection) est un algorithme de réduction de dimension de données qui “conserve au mieux la topologie des données d'origine”. Nous reviendrons sur ce que cela signifie, mais moralement, c'est dire qu'il conserve au mieux la structure du nuage de points que sont les données. Pour que la théorie mathématique sous-jacente soit applicable, il faut que les points soient uniformément répartis dans l'espace où ils se trouvent. Cela n'étant pas possible en pratique, les auteurs proposent de modifier la métrique.

En effet, plutôt que de considérer les données dans un espace \mathbb{R}^n munie d'une distance, ils proposent de passer par le biais d'une métrique. Une métrique étant une notion de distance dont la nature dépend de la position entre les points où elle est appliquée.

De cette façon, dans les zones où il y a beaucoup de points, les distances seront “plus courtes”, là où dans les zones moins denses, les distances seront “plus grandes”. Au sens de cette métrique, le nuage de points représentant les données est uniformément distribué.

En pratique, ce formalisme n'étant pas applicable, on passe par la théorie des graphes. De ce point de vue, la métrique au point x_i est représentée par des arêtes partant de x_i dont les poids associés sont donnés par la distance entre x_i et les autres points.

Pour rendre uniforme la distribution des points, on considère en tout point x_i les k plus proches $\{x_{i_j}\}_{1 \leq j \leq k}$ voisins (où k est un paramètre à préciser) ; et on note ρ_i la distance associée au point le plus proche :

$$\rho_i = \min_{x_i \neq x_{i_j}} \{d(x_i, x_{i_j}) \mid 1 \leq j \leq k\}$$

Puis on associe à chaque arête (x_i, x_{i_j}) de ce voisinage le poids :

$$w(x_i, x_{i_j}) = \exp \left(\frac{-\max \{0, d(x_i, x_{i_j}) - \rho_i\}}{\sigma_i} \right)$$

où σ_i est un paramètre de régularisation calculé par rapport à ρ_i et k . Ici, on prend le contre-pied avec l'idée de la métrique. Plutôt que d'associer un poids grand aux longues arêtes émergeant de x_i , on fait l'inverse. On peut le voir comme une valeur de proximité de x_{i_j} par rapport à x_i : plus un point est proche de x_i plus son lien avec x_i sera fort.

La présence du $-\rho_i$ dans le min nous garantie que, pour chaque point x_i , au moins une arête (x_i, x_{i_j}) a un poids de 1 ($\exp(0) = 1$). C'est en ce sens que la distribution des points est homogène pour la métrique.

Cela étant dit, le graphe obtenue n'est plus symétrique ce qui pose problème pour la suite. En effet, le poids associée à l'arête (x_i, x_j) n'est pas le même que celui associé à (x_j, x_i) puisque le premier dépend du voisinage de x_i et le second du voisinage de x_j . Pour retrouver cette symétrie, on a associé à l'arête $\{x_i, x_j\}$ (non orientée) le point :

$$w^*(\{x_i, x_j\}) = w(x_i, x_j) + w(x_j, x_i) - w(x_i, x_j)w(x_j, x_i)$$

En interprétant le poids $w(x_i, x_j)$ comme la probabilité que l'arête allant de x_i à x_j existe, alors $w^*(x_i, x_j)$ correspond à la probabilité que l'une des deux arêtes (x_i, x_j) et (x_j, x_i) existe.

Le graphe obtenu représente la “topologie des données” citée plus haut et c'est cette structure que l'on cherche à conserver au mieux lors de la projection.

B.2. Projection en dimension d

Pour faire cette projection, on se fixe une dimension $d \ll n$ et on projette le graphe obtenu en dimension d . Pour se faire, les auteurs proposent d'utiliser le *Laplacien symétrisé normalisé*. On ne rentrera pas dans les détails, la seule chose à retenir est que cette méthode est rapide à calculer et permet une convergence plus rapide de ce qui suit.

Cette première projection étant grossière, l'on procède à une descente de gradient pour ajuster les position des points et poids des arrêtes du graphe projeté. Pour cela, on considère comme fonction coût la *cross-entropy* :

$$C((A, \mu), (A, \nu)) = \sum_{a \in A} \mu(a) \ln \left(\frac{\mu(a)}{\nu(a)} \right) + (1 - \mu(a)) \ln \left(\frac{1 - \mu(a)}{1 - \nu(a)} \right)$$

où A est la matrice d'adjacence du graphe, *e.i.* pour tout (i, j) , $A_{ij} = w^*(x_i, x_j)$. Sans rentrer dans les détails de ce que sont μ et ν , il faut voir que le premier membre contrôle la concentration des points proches, ce qui s'apparente à des clusters. Le second, quand à lui, contrôle l'écart dans entre les clusters. Ainsi, après régression on obtient un nuage de points en dimension d qui représente au mieux les zones denses tout en conservant les espaces entre clusters.

L'affichage est ensuite fait en fonctions des points $w^*(x_i, x_j)$ du graphe projeté. Il est important de noter qu'avec une telle représentation les axes ne correspondent à rien par rapport aux variables d'origines. Les seuls interprétations que l'on puisse faire d'une telle représentation est la présence ou non de clusters, traduisant plus ou moins de structure dans les données d'origine.

Cela étant dit, le paramètre k introduit plus haut joue un rôle important dans la représentation des données : là où un petit k va préserver les structures local du nuage, un k plus grand conversera mieux la structure global au prix des détails. En effet, plus k est petit moins un point x_i a de voisins proches dans le sens où il sera lié à moins de point. Cela à pour conséquence de ne pas tenir compte de liens plus larges, plus globaux, avec d'autres points.

C HDBSCAN

C.1. Algorithme DBSCAN

Pour effectuer le clustering (regroupement des données), on utilise l'algorithme HDBSCAN sur les données où l'on a appliqué l'algorithme UMAP (Voir Annexe B).

L'algorithme HDBSCAN est une version hiérarchique de DBSCAN (Density-Based Spatial Clustering of Application with Noise). C'est un algorithme de clustering qui permet, grâce à l'utilisation de la densité du jeu de données, de faire un clustering. En effet, si nos données sont composées de plusieurs groupes denses et distincts, alors l'algorithme DBSCAN fonctionne parfaitement pour faire le clustering. En ce sens, il est plus précis et polyvalent que l'algorithme k-means. Il utilise deux paramètres, ε et $\text{min}_{\text{points}}$, l'un correspond au rayon de l' ε -voisinage d'un point et l'autre définit le seuil à partir duquel l' ε -voisinage d'un point x du jeu de données est considéré comme dense. DBSCAN sépare les données en 3 catégories :

- Les points centraux (ou core-points) sont les points qui sont au cœur des clusters, *i.e.* leur voisinage doit compter plus de (au moins ?) $\text{min}_{\text{points}}$ de points.
- Les points frontières (ou border-points) qui sont voisins d'un point-coeur sans pour autant en être : ils correspondent à la frontière d'un cluster.
- Enfin les points aberrants, qui ne correspondent à aucun des deux cas. DBSCAN les traite très bien puisqu'il ne leur associe pas de clusters.

Cet algorithme fonctionne de la manière suivante : on prend un point x du jeu de données X qui n'a pas encore été visité et on le considère comme visité. On calcule l' ε -voisinage de ce point (*i.e.* la boule de centre x et de rayon ε) et on vérifie si le voisinage est dense. Si c'est le cas on assigne à tous les points dans

l' ε -voisinage un cluster et on regarde l' ε -voisinage des voisins de x .

Sinon, si le voisinage de x n'est pas dense on le considère comme point aberrant et on passe au point suivant.

Le bon réglage des paramètres est très important car si ε est trop petit, beaucoup de points seront considérés comme du bruit; au contraire s'il est trop grand, les clusters seront trop grands. De même, si $\text{min}_{\text{points}}$ est trop faible, trop de voisinages seront considérés comme dense et s'il est trop élevé, alors pas assez de voisinage seront considéré comme dense.

Le réglage de $\text{min}_{\text{points}}$ est assez facile. En général on prend $\text{min}_{\text{points}} = 2 \times k$ (avec k la dimensionnalité des données). Le réglage de ε est plus compliqué, mais on peut s'en sortir heuristiquement : il suffit de faire le graphe des k -distances, c'est-a-dire les plus petites distances tels qu'on a $k + 1$ points depuis le point utilisé. ε est alors donné par l'ordonnée de la valeur du point de changement de pente.

Nous souhaitons identifier la meilleure valeur pour le paramètre ε afin d'optimiser l'exécution de l'algorithme DBSCAN. De plus, si la densité des données est variable, il est difficile de trouver le bon compromis pour ce paramètre.

C.2. Algorithme HDBSCAN

Pour réaliser cet algorithme il faut créer un graphe de *mutual reachability*, c'est-a-dire un graphe qui relie chaque point x à tous les autres et dont les arêtes ont pour poids la *mutual reachability distance*. La mutual reachability distance entre deux points $x \in X$ et $x' \in X$ est le maximum entre leur k -distance respective et de la distance entre x et x' .

Ensuite, on réduit ce graphe en considérant l'arbre de poids minimum et on supprime itérativement les arêtes de poids les plus élevés.

Chaque arête a un poids ε qui décrit une solution de l'algorithme DBSCAN pour le paramètre ε . La suppression itérative de ces arêtes permet de définir un dendrogramme qui décrit les apparitions et disparitions de cluster en fonction de la suppression des arêtes.

Le problème c'est que le dendrogramme complet est illisible, il est donc simplifié en ne prenant que les clusters pertinents. Cette simplification se fait par la résolution d'un problème d'optimisation binaire sur la stabilité générale des clusters.

TABLE DES FIGURES

1.1	Résolution disponible sur le télescope Sentinel-2 pour chaque longueur d'onde	2
1.2	Comparaison entre une image fournie et deux images téléchargées de résolutions respectives 20 et 10 mètres. Toutes ont été prises à la date du 24/06/2023	3
1.3	Copie d'écran d'une image du lac depuis Copernicus	3
1.4	Copie d'écran d'une image du lac pour la longueur d'onde B8A depuis Copernicus	3
1.5	Masque obtenu après modification de l'image B8A à la date du 08/02/2015	4
1.6	Masque tronqué	4
1.7	Exemple de détection du lac sur différents niveaux de couverture nuageuse	6
1.8	Représentation du lac plus ou moins couvert associé à son histogramme tronqué pour supprimer les potentiels nuages Dates : lac (15/10/2017), nuages (28/08/2023), lac avec les nuages (25/08/2023)	6
1.9	Comparaison de l'intensité des pixels entre une image avec un taux de chlorophylle très important et une image nuageuse	8
1.10	Masquage d'une image après seuillage pour supprimer les nuages	8
1.11	Comparaison des différents indices de réflectance et de leur concentration en Chlorophylle-a associée	10
1.12	Lignes de niveaux de la bathymétrie du lac	10
1.13	Color Map de la bathymétrie calculée	10
1.14	Masque avec les trois niveaux de bathymétrie	11
1.15	Carte bathymétrique superposée à la Color map de concentration Chl-a	11
1.16	Carte bathymétrique superposée à la color map de concentration Chl-a en fonction des niveaux de profondeurs	12
2.1	Données représentées avec et sans observation des outliers pour les paramètres Temp, ChloraA, O2, Phyco, SpConductivité avec Z=3	15
2.2	Étude des corrélations croisant les variables redondantes	15
2.3	Matrice de corrélation croisant les variables redondantes	16
2.4	Cercle de corrélation généré à partir de l'analyse en composantes principales (PCA)	16
2.5	Matrice de corrélation croisant les variables	17
2.6	Croisement des concentrations de ChloraA et Phyco	18
2.7	Cercle de corrélation avec les variables SpConductivité, Phyco, ChloraA, Temp, O2	19
2.8	Cercle de corrélation avec l'ajout du rayonnement global et des vitesses du vent	19
2.9	Cercle de corrélation avec l'ajout de la durée du gel et des directions du vent	19
2.10	Visualisation UMAP et clustering HDBSCAN des données	20
2.11	Visualisation UMAP 3D et clustering HDBSCAN des données pour les paramètres nb_neighbor=15 et min_dist=0.1	20
2.12	Fenêtre de localisation de la bouée	21
2.13	Comparaison des concentrations mesurées et observées	21
2.14	Corrélation entre la lumière observée et mesurée	22
2.15	Concentration de la chlorophylle a selon les zones du lac	22

TABLE DES CODES

1	Fonction de generation des masques du fichier Python/CalculMask.py	4
2	Fonction qui recupere le seuil et le masque sans nuages	7
3	Seuillage binaire de la bathymétrie	12

RÉFÉRENCES

- [1] Deguene Diene (2023), *Analyse par images satellites de la dynamique d'eutrophisation des lacs : application au lac d'Aydat*
- [2] Stumpf, R. P., Holderied, K., & Sinclair, M. (2003). Determination of water depth with high-resolution satellite imagery over variable bottom types. *Limnology and Oceanography*, 48(1part2), 547-556. https://doi.org/10.4319/lo.2003.48.1_part_2.0547
- [3] Leland McInnes, John Healy, James Melville (2020), *UMAP : Uniform Manifold Approximation and Projection for Dimension Reduction*, Doi : 10.48550/arXiv.1802.03426
- [4] Vincent Courjault-Rade (2018), *Ballstering: un algorithme de clustering dédié à de grands échantillons*.
Regarder la partie HDBSCAN.
- [5] Grendaitė, D., Stonevičius, E., Karosienė, J., Savadova, K., & Kasperovičienė, J. (2018). Chlorophyll-a concentration retrieval in eutrophic lakes in Lithuania from Sentinel-2 data. *Geologija*, 4(1). <https://doi.org/10.6001/geol-geogr.v4i1.3720>
- [6] Bramich, J. M., Bolch, C. J. S., & Fischer, A. M. (2021). Improved red-edge chlorophyll-a detection for Sentinel 2. *Ecological Indicators*, 120, 106876. <https://doi.org/10.1016/j.ecolind.2020.106876>
- [7] Vazquez, M. V., Acuña-Alonso, C., Somoza, J. L. R., & Álvarez, X. (2021). Remote detection of cyanobacterial blooms and chlorophyll-A analysis in a eutrophic reservoir using Sentinel-2. *Sustainability*, 13(15), 8570. <https://doi.org/10.3390/su13158570>
- [8] O'Reilly, J. E., Maritorena, S., Mitchell, B. G., Siegel, D. A., Carder, K. L., Garver, S. A., Kahru, M., & McClain, C. R. (1998). Ocean Color chlorophyll algorithms for SEAWIFS. *Journal of Geophysical Research*, 103(C11), 24937-24953. <https://doi.org/10.1029/98jc02160>