University of Nottingham
UK | CHINA | MALAYSIA

Machine Learning Coursework

Yi XUE

December. 24 2020

1 Task 1

1.1 Data Pre-processing

All of the data pre-proccessing steps are finished by automated methods regardless of data size.

1.1.1 Discard noisy data

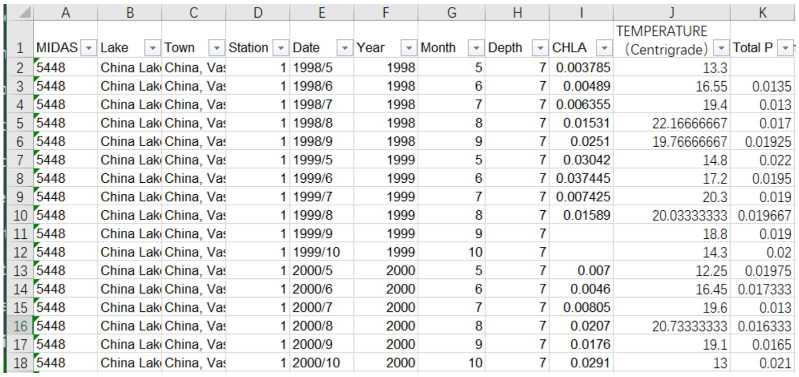
First, discard data with station 2 and depth not equal to 7. Due to that only sheet ’CHLA’ contains data with station 2, so they can not match the other two parameters (Temperature, total Phosphorus). In addition, considering that there is very few data with the depth not equal to 7, removing these irrelevant data. Besides, according to the requirement, only data from May to October need to be focused, so otherwise data are also discarded. This step is completed by the ’filter’ function in excel, which is the most simple way to winnow data.

1.1.2 Merge data

In this step, we need to calculate the average values of three variables for each months. In brief, I write a python program to calculate the average values for each variable for each month. Then, put these three sheets into one sheet and use excel built-in functions to merge them into same lines. The detailed steps are shown in appendix, including the codes in excel.

1.1.3 Discard missing data

After get the merging data , we need to discard missing data. Some data have no records for several months and it is difficult to complete them precisely, so I decide to remove data with little information. Finally, I preserve data from 1998 to 2013, except for 2004, from May to October.

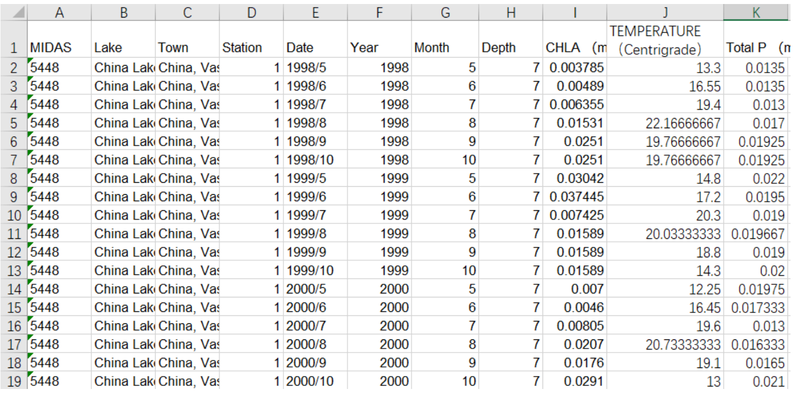


	A	B	C	D	E	F	G	H	I	J	K
1	MIDAS	Lake	Town	Station	Date	Year	Month	Depth	CHLA	TEMPERATURE (Centrigrade)	Total P
2	5448	China Lak	China, Va	1	1998/5	1998	5	7	0.003785	13.3	
3	5448	China Lak	China, Va	1	1998/6	1998	6	7	0.00489	16.55	0.0135
4	5448	China Lak	China, Va	1	1998/7	1998	7	7	0.006355	19.4	0.013
5	5448	China Lak	China, Va	1	1998/8	1998	8	7	0.01531	22.16666667	0.017
6	5448	China Lak	China, Va	1	1998/9	1998	9	7	0.0251	19.76666667	0.01925
7	5448	China Lak	China, Va	1	1999/5	1999	5	7	0.03042	14.8	0.022
8	5448	China Lak	China, Va	1	1999/6	1999	6	7	0.037445	17.2	0.0195
9	5448	China Lak	China, Va	1	1999/7	1999	7	7	0.007425	20.3	0.019
10	5448	China Lak	China, Va	1	1999/8	1999	8	7	0.01589	20.03333333	0.019667
11	5448	China Lak	China, Va	1	1999/9	1999	9	7		18.8	0.019
12	5448	China Lak	China, Va	1	1999/10	1999	10	7		14.3	0.02
13	5448	China Lak	China, Va	1	2000/5	2000	5	7	0.007	12.25	0.01975
14	5448	China Lak	China, Va	1	2000/6	2000	6	7	0.0046	16.45	0.017333
15	5448	China Lak	China, Va	1	2000/7	2000	7	7	0.00805	19.6	0.013
16	5448	China Lak	China, Va	1	2000/8	2000	8	7	0.0207	20.73333333	0.016333
17	5448	China Lak	China, Va	1	2000/9	2000	9	7	0.0176	19.1	0.0165
18	5448	China Lak	China, Va	1	2000/10	2000	10	7	0.0291	13	0.021

Figure 1: Data Example

1.2 Method 1 - Mean

Considering that there are only 90 lines of data, it is more convenient to complete data by calculating in excel file instead of python codes. Therefore, I complete the data by calculating the average value of one month before and after. For example, for missing data of August, I use built-in function ’AVERAGE’ in excel to calculate the mean value of July and September. For May and October, I use data of June and November respectively to fill in the missing part.



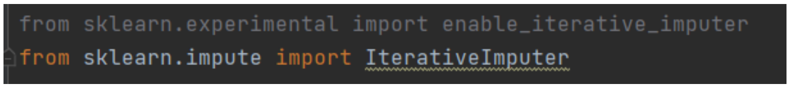
	A	B	C	D	E	F	G	H	I	J	K
1	MIDAS	Lake	Town	Station	Date	Year	Month	Depth	CHLA (m	TEMPERATURE (Centrigrade)	Total P (n
2	5448	China Lak	China, Va	1	1998/5	1998	5	7	0.003785	13.3	0.0135
3	5448	China Lak	China, Va	1	1998/6	1998	6	7	0.00489	16.55	0.0135
4	5448	China Lak	China, Va	1	1998/7	1998	7	7	0.006355	19.4	0.013
5	5448	China Lak	China, Va	1	1998/8	1998	8	7	0.01531	22.16666667	0.017
6	5448	China Lak	China, Va	1	1998/9	1998	9	7	0.0251	19.76666667	0.01925
7	5448	China Lak	China, Va	1	1998/10	1998	10	7	0.0251	19.76666667	0.01925
8	5448	China Lak	China, Va	1	1999/5	1999	5	7	0.03042	14.8	0.022
9	5448	China Lak	China, Va	1	1999/6	1999	6	7	0.037445	17.2	0.0195
10	5448	China Lak	China, Va	1	1999/7	1999	7	7	0.007425	20.3	0.019
11	5448	China Lak	China, Va	1	1999/8	1999	8	7	0.01589	20.03333333	0.019667
12	5448	China Lak	China, Va	1	1999/9	1999	9	7	0.01589	18.8	0.019
13	5448	China Lak	China, Va	1	1999/10	1999	10	7	0.01589	14.3	0.02
14	5448	China Lak	China, Va	1	2000/5	2000	5	7	0.007	12.25	0.01975
15	5448	China Lak	China, Va	1	2000/6	2000	6	7	0.0046	16.45	0.017333
16	5448	China Lak	China, Va	1	2000/7	2000	7	7	0.00805	19.6	0.013
17	5448	China Lak	China, Va	1	2000/8	2000	8	7	0.0207	20.73333333	0.016333
18	5448	China Lak	China, Va	1	2000/9	2000	9	7	0.0176	19.1	0.0165
19	5448	China Lak	China, Va	1	2000/10	2000	10	7	0.0291	13	0.021

Figure 2: Complete missing data with Mean Method

1.3 Method 2 - IterativeImputer

Iterative Imputer is a kind of Multivariate Imputation algorithm which considers not only the i-th feature non-missing values in that feature dimension itself, but also the entire set of available feature dimensions to predict the missing data. It is an experimental module in sklearn package (at least scikit-learn 0.21.) and is an ideal technique to estimate feature data that contains internal correlation.

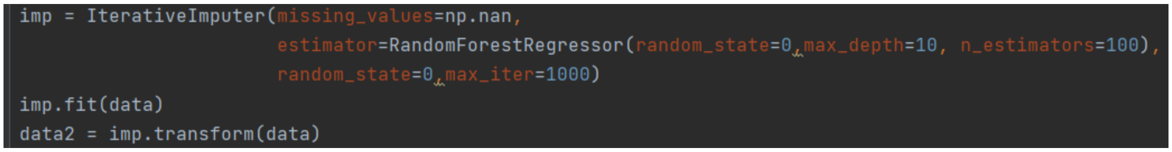
The main approach is an iterated round-robin fashion: at each step, a feature column is designated as output y and the other feature columns are treated as inputs X. A regressor is fit on (X, y) for known y. Then, the regressor is used to predict the missing values of y [4]. Here I use the year the the month as the X, and the three features CHLA, temperature and Total p as the target y. In this way, the correlation between these features stays the same.



```
from sklearn.experimental import enable_iterative_imputer
from sklearn.impute import IterativeImputer
```

Figure 3: Module sklearn.experimental

There are various estimator can be implemented with IterativeImputer such as BayesianRidge, KNeighborsRegressor, RandomForestRegressor, etc.[2] In this project, as all of the data are discrete, and it is not a classification problem, so regression is a better approach for imputing data. I use the RandomForestRegressor to complete data for this project.



```
imp = IterativeImputer(missing_values=np.nan,
                       estimator=RandomForestRegressor(random_state=0,max_depth=10, n_estimators=100),
                       random_state=0,max_iter=1000)
imp.fit(data)
data2 = imp.transform(data)
```

Figure 4: IterativeImputer Code

1.4 Analysis

Mean method is to use the average values of the proximity of missing elements. It is a kind of univariate imputation that only consider the same category feature of the element.

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees.Random decision forests correct for decision trees' habit of overfitting to their training set. In scikit learn, random forest is descibed as a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.[7]

Neither mean method or Random Forest method can predict the data accurately. The reasons are probably that: First, the sample dataset is too small to build a good training model. Second, the input data has bery little correlation with the output data in this project, so it is difficult to predict the correct data.

The results are shown below, and the codes can be seen in the code files.[2]

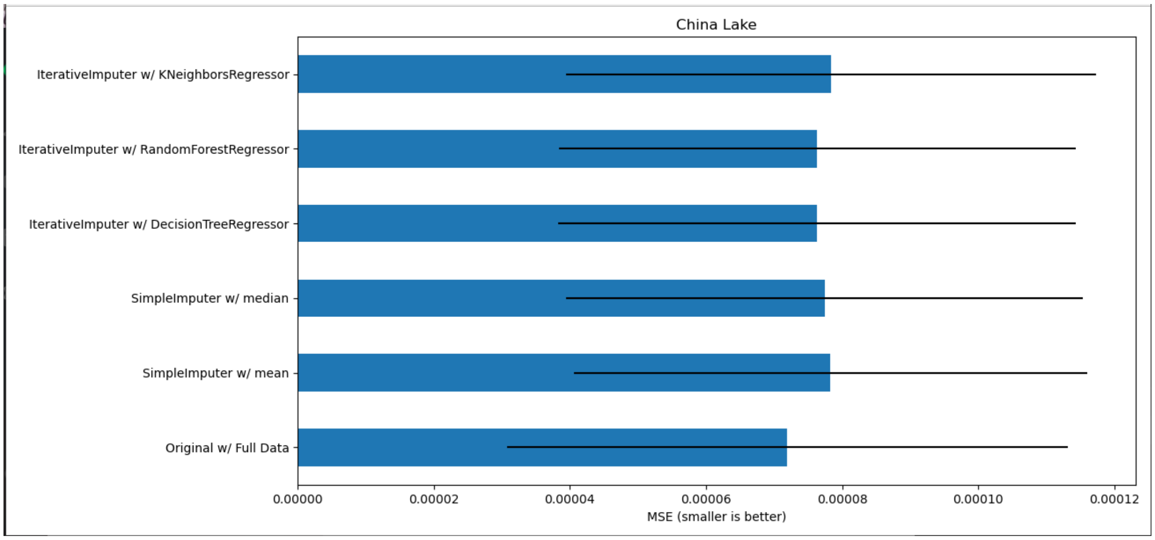
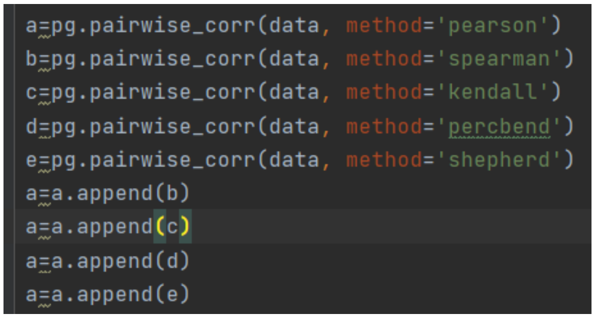


Figure 5: MSE

2 Task 2

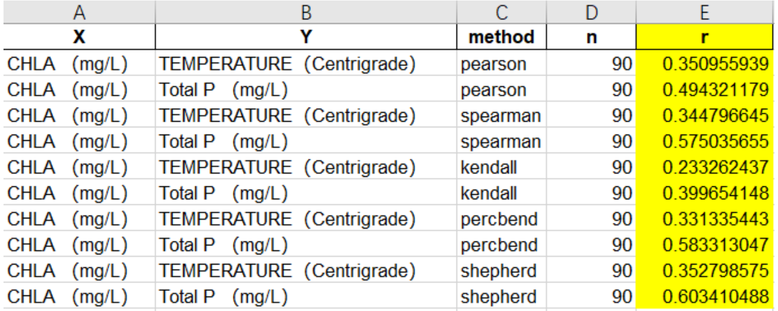
2.1 Methods

There are many kinds of methods to measure the correlation between variants. In this project, I use pearson, spearman, kendall, percentage bend and shepherd. First, I install the package ’pingouin’ which provides functions to implement these correlation measures directly. Then, I outupt them into the excel file together to show their correlation.[3] The results show that Temperature is a more significant factor than total Phosphorus in correlation with CHLA.



```
a=pg.pairwise_corr(data, method='pearson')
b=pg.pairwise_corr(data, method='spearman')
c=pg.pairwise_corr(data, method='kendall')
d=pg.pairwise_corr(data, method='percbend')
e=pg.pairwise_corr(data, method='shepherd')
a=a.append(b)
a=a.append(c)
a=a.append(d)
a=a.append(e)
```

Figure 6: Correlation Methods



A	B	C	D	E
X	Y	method	n	r
CHLA (mg/L)	TEMPERATURE (Centrigrade)	pearson	90	0.350955939
CHLA (mg/L)	Total P (mg/L)	pearson	90	0.494321179
CHLA (mg/L)	TEMPERATURE (Centrigrade)	spearman	90	0.344796645
CHLA (mg/L)	Total P (mg/L)	spearman	90	0.575035655
CHLA (mg/L)	TEMPERATURE (Centrigrade)	kendall	90	0.233262437
CHLA (mg/L)	Total P (mg/L)	kendall	90	0.399654148
CHLA (mg/L)	TEMPERATURE (Centrigrade)	percbend	90	0.331335443
CHLA (mg/L)	Total P (mg/L)	percbend	90	0.583313047
CHLA (mg/L)	TEMPERATURE (Centrigrade)	shepherd	90	0.352798575
CHLA (mg/L)	Total P (mg/L)	shepherd	90	0.603410488

Figure 7: Correlation Results

2.2 Analysis

In statistics, the Pearson correlation coefficient (PCC), also referred to as Pearson’s r, the Pearson product-moment correlation coefficient (PPMCC), or the bivariate correlation, is a statistic that measures linear correlation between two variables X and Y. It has a value between +1 and -1. A value of +1 is total positive linear correlation, 0 is no linear correlation, and -1 is total negative linear correlation. Pearson’s correlation coefficient is the covariance of the two variables divided by the product of their standard deviations. The form of the definition involves a ”product moment”, that is, the mean (the first moment about the origin) of the product of the mean-adjusted random variables; hence the modifier product-moment in the name.[6]

Spearman’s rank correlation coefficient is a nonparametric measure of rank correlation (statistical dependence between the rankings of two variables). It assesses how well the relationship between two variables can be described using a monotonic function. The Spearman correlation between two variables is equal to the Pearson correlation between the rank values of those two variables; while Pearson's correlation assesses linear relationships, Spearman's correlation assesses monotonic relationships (whether linear or not). If there are no repeated data values, a perfect Spearman correlation of +1 or -1 occurs when each of the variables is a perfect monotone function of the other.[8]

The Kendall rank correlation coefficient is a statistic used to measure the ordinal association between two measured quantities. It is a measure of rank correlation: the similarity of the orderings of the data when ranked by each of the quantities. Intuitively, the Kendall correlation between two variables will be high when observations have a similar (or identical for a correlation of 1) rank (i.e. relative position label of the observations within the variable: 1st, 2nd, 3rd, etc.) between the two variables, and low when observations have a dissimilar (or fully different for a correlation of -1) rank between the two variables.[5]

Percentage bend correlation is introduced by Wilcox (1994), it is based on a down-weight of a specified percentage of marginal observations deviating from the median (by default, 20 percent).Shepherd's Pi correlation is equivalent to a Spearman's rank correlation after outliers removal (by means of bootstrapped Mahalanobis distance).[1]

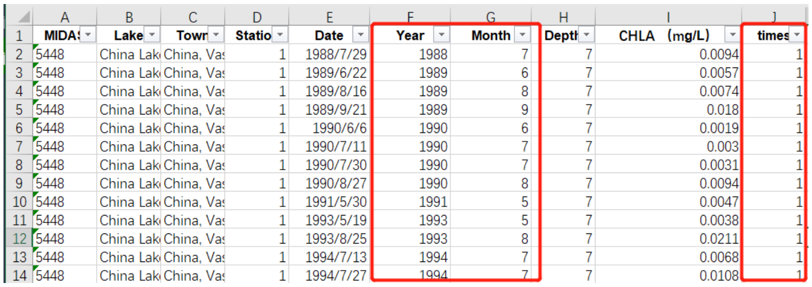
Pearson is not a good coefficient for this project, because generally, pearson correlation is a statistic that measures linear correlation between two variables. It is also a better method for normal distribution data and continuous data. In this project, the data are discrete.There are also inherent flaws in Pearson correlation method: it does not consider the influence of repeated data; it is not sensitive for absolute value. By contrast, Spearman's coefficient is appropriate for both continuous and discrete ordinal variables.Both Kendall and Spearman can be formulated as special cases of a more general correlation coefficient.

3 Appendix

3.1 Data-Preprocessing - Merge

3.1.1 Merge the data of same month

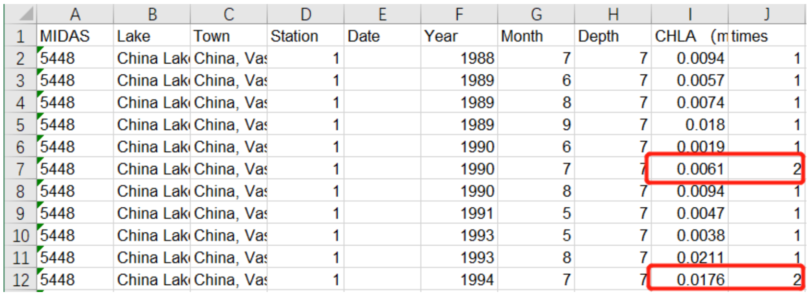
Firstly, I split the ’year’ and ’month’ from ’date’, and then add a new column with a parameter ’1’ for counting the ’times’ of addition for following steps.



	A	B	C	D	E	F	G	H	I	J
1	MIDAS	Lake	Town	Statio	Date	Year	Month	Deptl	CHLA (mg/L)	times
2	5448	China Lak	China, Vas	1	1988/7/29	1988	7	7	0.0094	1
3	5448	China Lak	China, Vas	1	1989/6/22	1989	6	7	0.0057	1
4	5448	China Lak	China, Vas	1	1989/8/16	1989	8	7	0.0074	1
5	5448	China Lak	China, Vas	1	1989/9/21	1989	9	7	0.018	1
6	5448	China Lak	China, Vas	1	1990/6/6	1990	6	7	0.0019	1
7	5448	China Lak	China, Vas	1	1990/7/11	1990	7	7	0.003	1
8	5448	China Lak	China, Vas	1	1990/7/30	1990	7	7	0.0031	1
9	5448	China Lak	China, Vas	1	1990/8/27	1990	8	7	0.0094	1
10	5448	China Lak	China, Vas	1	1991/5/30	1991	5	7	0.0047	1
11	5448	China Lak	China, Vas	1	1993/5/19	1993	5	7	0.0038	1
12	5448	China Lak	China, Vas	1	1993/8/25	1993	8	7	0.0211	1
13	5448	China Lak	China, Vas	1	1994/7/13	1994	7	7	0.0068	1
14	5448	China Lak	China, Vas	1	1994/7/27	1994	7	7	0.0108	1

Figure 8: Split & add new column

Then I need to calculate the average CHLA for each month. I use python ’xlrd’ to import data, and then I merge those lines of same ’year’ and ’month’ by adding up ’CHLA’ and ’times’ as long as the data are of the same month.



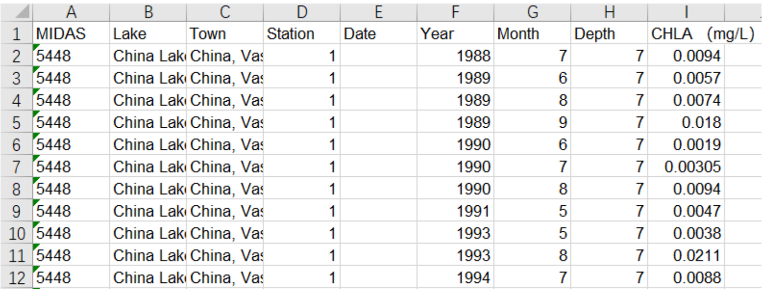
	A	B	C	D	E	F	G	H	I	J
1	MIDAS	Lake	Town	Station	Date	Year	Month	Depth	CHLA (m	times
2	5448	China Lak	China, Vas	1		1988	7	7	0.0094	1
3	5448	China Lak	China, Vas	1		1989	6	7	0.0057	1
4	5448	China Lak	China, Vas	1		1989	8	7	0.0074	1
5	5448	China Lak	China, Vas	1		1989	9	7	0.018	1
6	5448	China Lak	China, Vas	1		1990	6	7	0.0019	1
7	5448	China Lak	China, Vas	1		1990	7	7	0.0061	2
8	5448	China Lak	China, Vas	1		1990	8	7	0.0094	1
9	5448	China Lak	China, Vas	1		1991	5	7	0.0047	1
10	5448	China Lak	China, Vas	1		1993	5	7	0.0038	1
11	5448	China Lak	China, Vas	1		1993	8	7	0.0211	1
12	5448	China Lak	China, Vas	1		1994	7	7	0.0176	2

Figure 9: Merge Example

Then let CHLA divided by ’times, and we get the average CHLA of every month. After that, apply these steps for Temperature and total Phosphorus.

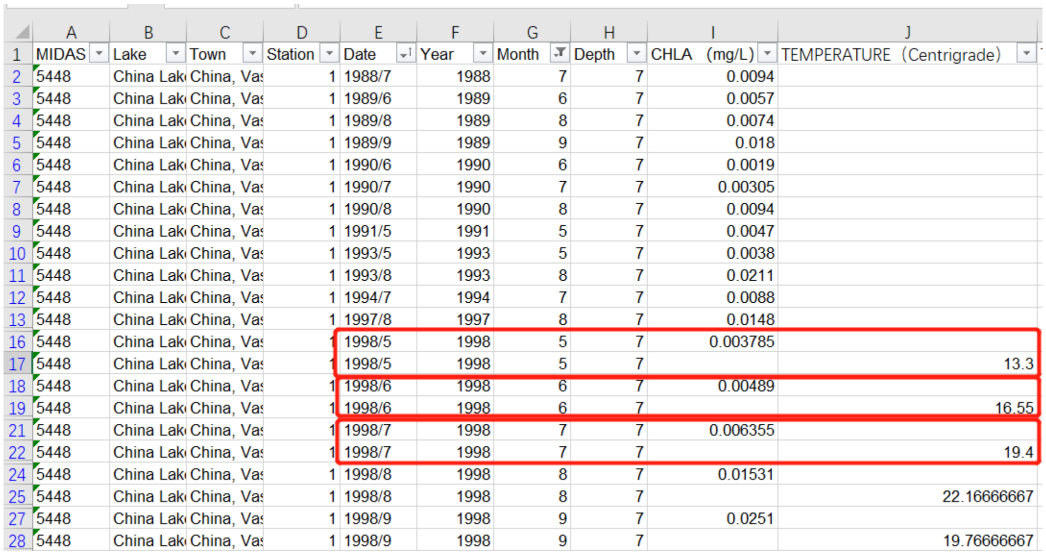
3.1.2 Merge three parameters

In order to merge three separate sheets of CHLA, Temperature, and total Phosphorus, I first merge CHLA and Temperature. Firstly, combine them into a same sheet and sort the data according to the date by using ’filter’ in excel. Then we can see some data are of same month and we need to merge them into the same line.
I use the built-in excel programming language to do this step:
After merge three sheets, we get the final data.



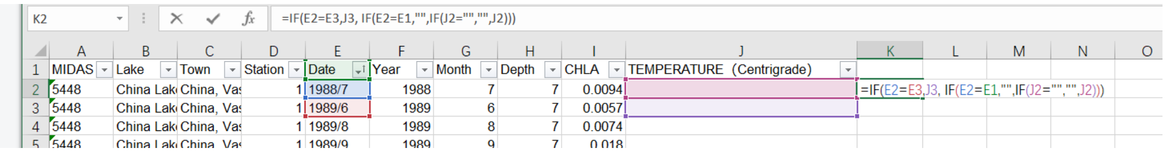
	A	B	C	D	E	F	G	H	I
1	MIDAS	Lake	Town	Station	Date	Year	Month	Depth	CHLA (mg/L)
2	5448	China Lak	China, Va	1		1988	7	7	0.0094
3	5448	China Lak	China, Va	1		1989	6	7	0.0057
4	5448	China Lak	China, Va	1		1989	8	7	0.0074
5	5448	China Lak	China, Va	1		1989	9	7	0.018
6	5448	China Lak	China, Va	1		1990	6	7	0.0019
7	5448	China Lak	China, Va	1		1990	7	7	0.00305
8	5448	China Lak	China, Va	1		1990	8	7	0.0094
9	5448	China Lak	China, Va	1		1991	5	7	0.0047
10	5448	China Lak	China, Va	1		1993	5	7	0.0038
11	5448	China Lak	China, Va	1		1993	8	7	0.0211
12	5448	China Lak	China, Va	1		1994	7	7	0.0088

Figure 10: Results of Merge



	A	B	C	D	E	F	G	H	I	J
1	MIDAS	Lake	Town	Station	Date	Year	Month	Depth	CHLA (mg/L)	TEMPERATURE (Centrigrade)
2	5448	China Lak	China, Va	1	1988/7	1988	7	7	0.0094	
3	5448	China Lak	China, Va	1	1989/6	1989	6	7	0.0057	
4	5448	China Lak	China, Va	1	1989/8	1989	8	7	0.0074	
5	5448	China Lak	China, Va	1	1989/9	1989	9	7	0.018	
6	5448	China Lak	China, Va	1	1990/6	1990	6	7	0.0019	
7	5448	China Lak	China, Va	1	1990/7	1990	7	7	0.00305	
8	5448	China Lak	China, Va	1	1990/8	1990	8	7	0.0094	
9	5448	China Lak	China, Va	1	1991/5	1991	5	7	0.0047	
10	5448	China Lak	China, Va	1	1993/5	1993	5	7	0.0038	
11	5448	China Lak	China, Va	1	1993/8	1993	8	7	0.0211	
12	5448	China Lak	China, Va	1	1994/7	1994	7	7	0.0088	
13	5448	China Lak	China, Va	1	1997/8	1997	8	7	0.0148	
16	5448	China Lak	China, Va	1	1998/5	1998	5	7	0.003785	
17	5448	China Lak	China, Va	1	1998/5	1998	5	7		13.3
18	5448	China Lak	China, Va	1	1998/6	1998	6	7	0.00489	
19	5448	China Lak	China, Va	1	1998/6	1998	6	7		16.55
21	5448	China Lak	China, Va	1	1998/7	1998	7	7	0.006355	
22	5448	China Lak	China, Va	1	1998/7	1998	7	7		19.4
24	5448	China Lak	China, Va	1	1998/8	1998	8	7	0.01531	
25	5448	China Lak	China, Va	1	1998/8	1998	8	7		22.16666667
27	5448	China Lak	China, Va	1	1998/9	1998	9	7	0.0251	
28	5448	China Lak	China, Va	1	1998/9	1998	9	7		19.76666667

Figure 11: Results of Combine



```
K2    =IF(E2=E3,J3, IF(E2=E1,"",IF(J2="","",J2)))
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	MIDAS	Lake	Town	Station	Date	Year	Month	Depth	CHLA	TEMPERATURE (Centrigrade)					
2	5448	China Lak	China, Va	1	1988/7	1988	7	7	0.0094		=IF(E2=E3,J3, IF(E2=E1,"",IF(J2="","",J2)))				
3	5448	China Lak	China, Va	1	1989/6	1989	6	7	0.0057						
4	5448	China Lak	China, Va	1	1989/8	1989	8	7	0.0074						
5	5448	China Lak	China, Va	1	1989/9	1989	9	7	0.018						

Figure 12: Merge

References

[1] Correlation types.

[2] Imputing missing values with variants of iterativeimputer.

[3] pingouin.corr.

[4] sklearn.impute.iterativeimputer.

[5] Kendall rank correlation coefficient, Dec 2020.

[6] Pearson correlation coefficient, Nov 2020.

[7] Random forest, Dec 2020.

[8] Spearman's rank correlation coefficient, Dec 2020.