Error Detection on National Weather Stations

Chenjie Luo, Texas A&M University, and Kun Yang, Texas A&M University

ABSTRACT

In today's life, data plays a more and more important rule for us in order to analyze the phenomena in our society. For example, Weather forecast has a great influence on people's daily life. Therefore, there is a significant demand for weather services to guarantee the precision of their collected data in weather analyzing, daily temperature forecasting and alert on extreme weathers. Given an intuition that daily temperature in one station is supposed to have certain correlation with nearby stations, in face of an ocean of data we base on the correlation of each station's data and longitude and latitude of each station to detect the whether a station's data is likely to be inaccurate. In this project, Multivariate Gaussian Distribution Model is used for detection of error on certain day and conditional expectation is used to determine which station's data is likely to be an outliner.

1. Introduction

People would like to and tried to predict the weather since the ancient time, and now we have a mature system for estimate the climate's behavior. In [1] Cess, Porter, etc. showed us the traditional 19 climate models for weather prediction. But a fact is that all these prediction methods need accurate weather station data, so what if the data from the station is corrupted?

To solve this problem, Aliaksei Sandryhaila and José M. F. Moura introduced graph Fourier transformation to predict a single corrupted data from weather stations in [2] and [3]. They achieved a decent 89% accuracy. And this work inspired us as well, what if we introduce the material we have learned in the class to make the same prediction? So we did some investigating on spatial model estimation and found Antonio S. Cofino, Rafael Cano, Carmen Sordo and Jose M. Guti ′ errez used the Bayesian network to predict the rainfall using the spatial model in [4]. These works are the intuition of our project.

In our work, we captured 111 days' average temperature from 80 weather stations within a whole year all over the Contiguous United States. We will try to corrupt the data and find whether we can detect the error or not. The challenge of our work is that we don't have a clear image of the corrupted data. Because of this, we cannot have samples with a clear label for us to train as the traditional classification problems. We come up with our method in this project; it seems simple but effective.

This report is divided into four parts, in the following section you will see the method of detecting whether a day's data is corrupted or not, and then we will try to figure out where the error occurred. After that will be the result of the experiment and the conclusion of our project.

2. Detection of error day

According to the content from our course Estimation and Detection, we attempt to use Multivariate Gaussian Distribution as the model to figure out the single error existing in our datasets since this is the most straightforward method came to our mind. Multivariate Gaussian distribution model has an apparent advantage over others which is it can depict the correlation between all of the stations. We calculate the joint probability of each day's temperature given eighty stations' daily average temperature. If the probability is lower than our threshold, it is least likely to happen and we will infer there may exist corrupted data within this day. The first step we picked out the whole year's temperature data in 2003 in stations around the United States. Since stations from Alaska and Hawaii are poorly correlated with stations in Contiguous United States, we remove the observed data from these stations and remove days which contain missing data points in certain stations. In the end, we obtain eighty stations complete data lasting 111 days.

When applying Gaussian Model, the two key points we care about most are the mean and the variance. And we can prove that sample mean and sample covariance are two sufficient statistics for Multivariate Gaussian Distribution. For the multivariate Gaussian Model, it turns to be mean and covariance. After referring to the related materials, we realize training data size m needs to much greater than station size n, in order to obtain the positive definite covariance matrix.

$$\boldsymbol{P(X;\mu,\Sigma)} = \frac{\mathbf{1}}{(\mathbf{2}\boldsymbol{\pi})^{\frac{n}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}}\, \boldsymbol{exp}\,(\frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{X}-\boldsymbol{\mu})^{\boldsymbol{T}}\boldsymbol{\Sigma}^{-\mathbf{1}}(\boldsymbol{X}-\boldsymbol{\mu}))$$

$\boldsymbol{\mu}$ is the true mean and Σ is the covariance matrix. Here we use sample mean $\overline{\mathrm{x}}$ and sample covariance $\boldsymbol{Q}$

$$\overline{\boldsymbol{x}} = \frac{\mathbf{1}}{\boldsymbol{N}} * \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}} \boldsymbol{X}^{(\boldsymbol{i})}$$

$$\boldsymbol{Q} = \frac{\mathbf{1}}{\boldsymbol{N}-\mathbf{1}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}} (\boldsymbol{X}^{(\boldsymbol{i})}-\overline{\boldsymbol{x}})(\boldsymbol{X}^{(\boldsymbol{i})}-\overline{\boldsymbol{x}})^{\boldsymbol{T}}$$

as Maximum Likelihood Estimator of true mean $\boldsymbol{\mu}$ and true covariance Σ. After training the model, we can simply plug in the data needed to be verified. We use the training set to determine a threshold, if the joint probability of test day's temperature is below the threshold, we classify it as a day containing error data.

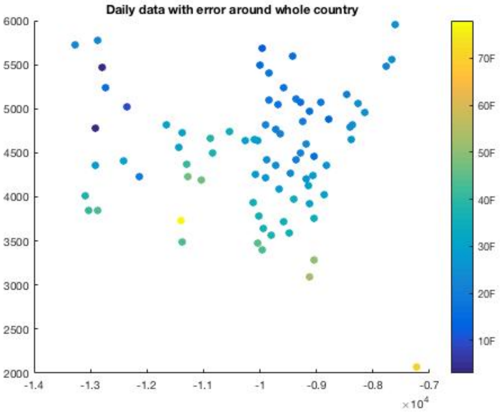

Fig.1. A detected day with error

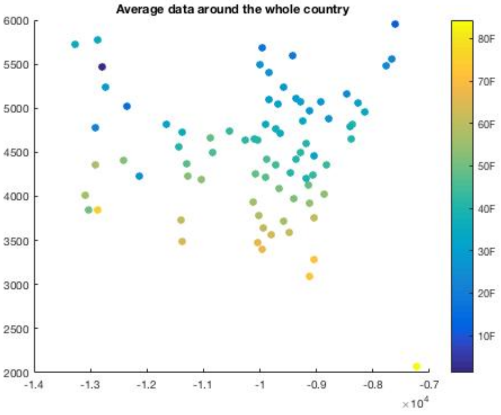

Fig.2. Average data around the country

Above is one day in the test set with certain corrupted data. As we compare this day with average data, we can intuitively detect this day is likely to be wrong as well. However, when we adjust the threshold, it is not surprising that the frequency of false alert increases while the correct detection increases. For example, when we take the threshold = –log(Joint Probability) = 300, the error detection could be up to 95% out of all error days while the false alert also accounts for 44%.

3. Station Detection

Up till now we can already detect whether the data is corrupted or not, in this part we will try to find the exact station which contain corrupted data.

The basic idea of this part is quite straightforward, as we already know that each station will in some sense have a strong correlation with its surrounding stations, we have the following assumption: if a specific station is an outliner, we can still have the right data by computing the conditional expectation of this station. After we acquired conditional expectations of all the stations, we make a comparison of the data with error and the conditional expectation, get the most significant absolute difference of the two sets, and this station is the station who holds the corrupted data.

$$\mathrm{E}_{\mathrm{con}_i} = \Sigma x_i p(x_i |\, x_1, x_2 \dots x_{i-1}\, , x_{i+1}, \dots x_n)$$

$$\theta = \mathrm{argmax}_\theta \left(abs\left(E_{con_i} - y_i\right)\right)$$

In the first equation, because of the Gaussian distribution, we say $\mathrm{x_i}$ is centered with the mean value of the station, and within a span of 3 times of the standard deviation of the station itself by approximation. The probability density is the density acquired in the second part, the LOO Gaussian density function. $\mathrm{y_i}$ is the real value we get for the stations.

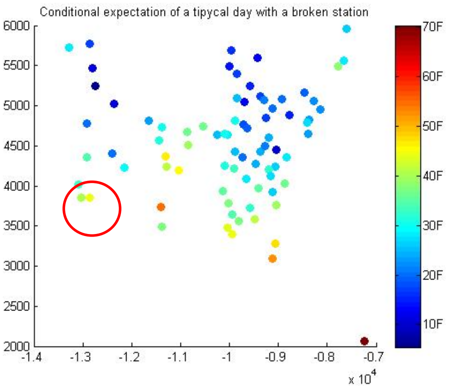

Fig.3. a typical conditional expectation

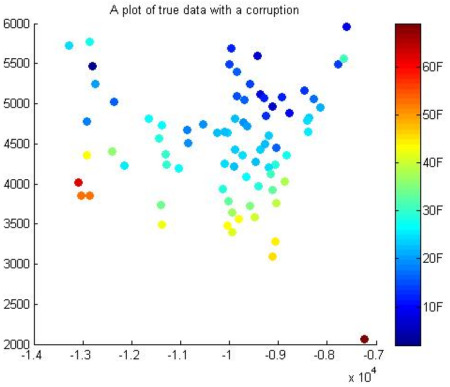

Fig.4. A ’true’ data

Not surprisingly we can find a massive difference between the ‘true’ data and the conditional expectation, this proves the feasibility of our method.

From the plots, we can also see that there is more than one point that varies from the conditional expectation. That is because, in this method, there is a high chance that we can get the right detection. But we also need to consider the following problem: if the error station is strongly correlated with other stations, a significant error may also result in a massive difference between the actual value and the conditional expectation, this is a part our work can’t solve right now, but we will talk about this in the conclusion part and come up with our thought to solve this problem.

4. Results

A. Error detection

When we used Leave-One-Out Estimate, in which we fed the Multivariate Gaussian Model with 110 points each time and left one day as the test set. We acquired the correct detection percentage at 95.7% while the false alert percentage was 44%.

B. Station detection

In station detection part, we used the conditional expectation and the absolute difference with the current test data, we acquired the accuracy at 75.15%.

Therefore, within the whole experiment, we can acquire up to 71.25% correct detection percentage.

5. Conclusion and future work

A. Conclusion of the work

In our project, we used the method taught in the class. Assuming that the weather's spatial model holds a correlation of the distance and thus can be represented using a multivariable non-independent Gaussian distribution. Then we use the maximum likelihood estimator to estimate the mean and covariance matrix of the system. After that, we compute the Log Likelihood to detect whether there is an error or not. To calculate the threshold, we make a mean of the LOO(Leave-One-Out) estimate of all right data likelihood and get the mean of these data. This will give us a false alert of 44% while correct detection of over 95%. As our primary objective is to make sure more errors are able to be detected, we consider this to be acceptable. The very last part of the system is to identify the position of the error. In this part, we get a total accuracy of over 75%, and we will talk about further possible improvement in the future work part.

B. Comparison with Existing work:

We have introduced [2] as our inspiration for this work, so how did our project perform compared to this great work? In [2], they reached an 89% of error detection rate with a 20-degree error. With the similar 20 degrees error, we can achieve a 95% which is more accurate than the work. But our project indeed needs much more raw data for training. In [2], they used data from a single month two detect a station number of 150. But in our work, as we said in part two, we need the sample size at least bigger than the feature dimension size, which means we need an enormous sample size to reach a better detection rate.

However, compared with graph Fourier transformation method in [2], we also have the advantage besides the higher accuracy. We can find the corrupted station at a rate of about 75%, that is to say, we have a total of 71% to correctly detect which station is broken. In this sense, we think our work did a pretty job here using a traditional method.

C. Future Work:

In the station detection part, we have talked about the relatively low rate of detection due to the strong correlation of some of the corrupted station and the usual stations. So how to reduce this effect. We think a simple method, judge that the error has occurred among the three stations with the biggest difference between the expectation and the 'true' data. This method can effectively increase the rate, but the problem is that this might significantly increase the cost for we trade off the accuracy of the exact station to get a higher accuracy of the right detection.

So a possible method is to find an abnormal point inside an area. For the conditional expectation of the error, a station that is close to its true temperature but the surrounding point are all far from a normal one, thus introduces an abnormal point or a ‘high frequency’ point in the system. We think this problem might be able to be solve on partial graph Fourier transformation. But this requires much more investigation which might be solved in the future.

REFERENCES

[1] A. Sandryhaila and J. M. F. Moura, "Discrete Signal Processing on Graphs," in IEEE Transactions on Signal Processing, vol. 61, no. 7, pp. 1644-1656, April1, 2013.

[2] A. Sandryhaila and J. M. F. Moura, "Discrete Signal Processing on Graphs: Frequency Analysis," in *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042-3054, June15, 2014.