Determining Relative Errors of Satellite Precipitation Data over The Netherlands

QINGYU WANG

March, 2016

SUPERVISORS:
Dr. Y. Zeng (Yijian)
Dr. Ir. C.M.M.(Chris) Mannaerts

Determining Relative Errors of Satellite Precipitation Data over The Netherlands

QINGYU WANG

Enschede, The Netherlands,

March, 2016

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.

Specialization: Water Resources and Environmental Management

ABSTRACT

Satellite precipitation estimates data are widely used for a variety of studies, including the hydrologic and climate modeling, weather forecasting, and agriculture management or extreme events prediction. However, satellite precipitation estimation is inevitably followed with errors which are caused by different factors, therefore it is essential to evaluate the relative errors of satellite precipitation data. A realizable method which can be used to quantify the relative errors in large-scale datasets is triple collocation. This method can objectively obtains the relative errors for at least three or more independent products. But before estimation of relative errors, the bias of the products relative to each other should be reduced or removed. This study tests the cumulative distribution function (CDF) matching approach which aims to reduce the bias among three precipitation products over the Netherlands. Afterwards, the triple collocation technique is applied to determine the relative errors of these precipitation products. The three precipitation datasets are, the Climate Prediction Center morphing method (CMORPH), the Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks (PERSIANN) and the gridded rain gauge data interpolated from in situ rain gauge measurement data provided by the Royal Netherlands Meteorological Institute (KNMI).

The results suggest that CPMORPH product's behavior is better than PERSIANN's when they are correlated to the interpolation products. The cumulative distribution function (CDF) matching is a superior approach which can improve the correlation coefficient and reduce the root mean square error (RMSE) among precipitation products. For the relative errors among the three sets of precipitation data, it is found that the relative error of CMORPH is lower than the other two products', interpolation is at the medium while PERSIANN is the highest one.

Key words: precipitation products, bias correction, triple collocation, relative errors.

ACKNOWLEDGEMENTS

I would like to express my thanks to all of the people who support me during the study and MSc research in ITC.

First of all, I would like to express my honest gratitude to my first supervisor Dr. Yijian Zeng and my second supervisor Dr C.M.M.Mannaerts. Dr. Yijian Zeng has walked me through every stage from thesis proposal to accomplish with patience and encouragement, especially supported me with the CDF matching and triple collocation Matlab code. While Dr C.M.M.Mannaerts supported me with detailed ILWIS techniques and gave me clear comments of every steps. Their encouragement and trust helped me across the difficult period during the research. This thesis would not be finished without their help.

I would also extend my sincere gratitude to the PHD student Vahid Rahimpour, who patiently helped me solving the problems of the ILWIS syntax, ArcGIS model and R script code, without his support, the thesis couldn't be progressed so smoothly.

Special thanks to my friend Wang Yancheng who supported me with the Matlab code to help me improve the working efficiency.

Special thanks to the friends who lend me their laptops. The data collection of this thesis is a super huge workload, without their laptops the data couldn't be collected in such a limited time.

Thanks to all ITC staff, especially to all staff in WREM during the lectures. I learned a lot of useful knowledge and techniques which will help with my career in the future.

Finally, I am grateful to my beloved family for their permanent support including my boyfriend for his encourage and company.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1. Background

Precipitation is defined as any product of the condensation of atmospheric water vapour that falls under the gravity ("AMS Glossary,") As the most important component in water cycles and land-atmosphere interactions, precipitation plays an significant role in various hydrological models and studies (Tapiador et al., 2012). The study of surface precipitation is also important for society and people's livelihood, because inaccurate measurements and forecasts can mean risk to crops, livestock, property and even lives (Beek, 2013). Therefore, obtaining reliable and accurate precipitation data is crucial for local, regional and global agriculture management and hydrologic prediction, like urban flood early warning system.

In addition, precipitation has a more direct impact on human life than other atmospheric phenomena have, such as heavy rain events and flash floods (Vincenzo Levizzani, Amorati, & Meneguzzo, 2002). A representative heavy rainfall event is, on 26 August of 2010, the eastern part of The Netherlands with the bordering part of Germany were flushed by a series of heavy rainfall events, which lasting for more than a day. Observed in 24 h, there was over an area of 740 km^2 and more than 120 mm of rainfall. This extreme event caused local flooding at city centers, highways and agricultural fields, and also a considerable financial loss (Brauer et al., 2011). Figure1.1 shows the urban flooding in the Netherlands. In order to avoid such disaster, it is necessary not only to improve urban drainage systems, but also to estimate the precipitation in advanced, make an early warning system and extremely rapid response times. Therefore, the precipitation estimation plays an important role in the whole procedure.

Figure 1-1 The urban flooding in the Netherlands

There are several ways to estimate precipitation, traditionally precipitation is usually measured with rain gauges, but variant instruments have been developed until now. Most representative instruments are satellites, ground-based radar, distrometers and microwave links. As the most common measurement, the most important advantage for rain gauges is giving direct measurement of rain accumulation. However, there are several drawbacks of rain gauges, such as poor spatial coverage, suffered from wind effects and other resources of error (Beek, 2013). Besides, gauges are limited to land regions and islands, thus they are unable to verify oceanic rainfall estimations (Ebert & McBride, 2000) and (V. Levizzani, Bauer, & Turk, 2007).

Satellite precipitation estimates are widely used to measure global rainfall on near real-time and monthly timescales, which can be used for hydrological and climatological studies, tropical rainfall potential studies, numerical weather prediction (NWP) data assimilation, now-casting and flash flood warning, and water resources monitoring. In addition, satellites provide insight into the synoptic scale precipitation and are able to obtain an estimate of precipitation in areas where are too remote for ground-based instruments. However, satellite estimates are often affected by instrument noise, semitransparent clouds, and uncertainty in surface emission modelling (Miralles, Crow, & Cosh, 2010). In addition, the images from satellites are lack of the details and also usually have larger quantitative errors than ground-based instruments (Beek, 2013). Therefore, similar to any observation data, it is crucial to investigate their accuracy, internal variability and error structures. This investigation can be done by verifying the satellite estimates against independent data from rain gauges data (V. Levizzani et al., 2007).

1.2. Problem statement

When applying the precipitation data in hydrologic and climate studies, agriculture management or extreme atmospheric events prediction, it is essential to evaluate the internal variability and changes of precipitation. Thus, a long-term precipitation data from multiple source is needed for study the internal characteristic of precipitation. However, as the availability of precipitation estimates is increasing with a various of instruments, inevitably, the errors always follow the precipitation estimations. The errors are caused from several factors depending on the measurement instruments. For example, the errors of rain gauge measurements are from the gauges' spatial coverage, topography and environmental conditions such as wind and evaporation (Beek, 2013). For satellite retrievals (both radiometer and radar), the Larger quantitative errors are from the assumptions of the surface emissivity, neglecting evaporation below clouds, and empirical relationships (Alemohammad, McColl, Konings, Entekhabi, & Stoffelen, 2015). Therefore, validation of precipitation estimates from several products is always a problem which need to be solved urgently.

In addition, for most hydrological and climatological studies and models, it is necessary to understand the relative error structures among different precipitation datasets. However, different precipitation data in different spatial and temporal resolution are variant among each other. To solve the problems mentioned above, it is required to find the methods which can quantify the independent errors of each datasets and estimate the relative errors among them.

1.3. Research questions

Based on the problems proposed above, hereby present the research questions as following.

1. What are the statistic differences between the satellite precipitation products and the in-situ based products?
2. Can the bias-correction method (e.g. CDF matching method) improve the correlation between multiple sources of data?
3. How to quantify and estimate the relative errors among different precipitation products?
4. How do different temporal scales of datasets chosen will affect the results?

1.4. Objectives

1.4.1. Main objective

The main objective of this research is to characterize the statistic differences and estimate the relative errors of multiple precipitation products from CMORPH and PERSIANN satellites, and interpolation rain gauges data from the year of 2003 to 2013 over the Netherlands.

1.4.2. Sub objectives

1. To get the satellite observation and in-situ interpolate data over the Netherlands;
2. To estimate the statistic differences (e.g. correlation coefficient & root mean square error) among different precipitation products;
3. To test the cumulative distribution function (CDF) matching approach to reduce the bias among these three precipitation product;
4. To estimate the relative errors of each products using triple collocation technique;
5. Change the temporal resolutions of the precipitation products to see whether different datasets chosen will affect the error structures.

2. Literature review

2.1. Bias correction

Bias correction techniques have been applied on several studies especially on climate models, weather forecast models, radar and satellite precipitation products.

Early in 2007, When Leander, R. and Buishand were investigating the resampling of regional climate model output for the simulation of extreme river flows, they pay much attention to the bias correction of RACMO precipitation and found that a relatively simple nonlinear correction, not only adjusts both the biases in the mean and variability, but also leads to a better reproduction of observed extreme daily and multi-day precipitation amounts than the commonly used linear scaling correction (Leander & Buishand, 2007).

Based on Leander, R. and Buishand's study, W. Terink of Wageningen University applied bias correction on the forcing variables between downscaled ERA15 (ECMWF-reanalysis data) precipitation and temperature with observed precipitation and temperature over the Rhine basin. They corrected the precipitation by fitting the mean and coefficient of variation (CV) of the observations and the temperature by fitting the mean and standard deviation of the observations respectively. The results indicate that bias correction leads to a significant decrease of the precipitation and temperatures difference. (Terink, Hurkmans, Torfs, & Uijlenhoet, 2009)

Later on, a simple mean correction and a least-square bias correction techniques were applied on decadal forecasts which are produced from a state-of-the-art coupled forecasting model. The objective is to explore the possibility of reducing large systematic biases in the North Pacific sea surface temperature anomalies. The results show that the bias corrected forecasts reduced root mean square errors and also significantly improve the anomaly correlations with observations. In addition, the bias corrected forecasts better predict the extreme weather events than before and the prediction skill is also improved from less than a year to five years (Narapusetty, Stan, & Kumar, 2012).

Refer to the bias correction of precipitation products, different investigators proposed and developed several bias correction techniques to improve radar and satellite precipitation products. (Ahnert, Krajewski, & Johnson, 1986), (J. A. Smith & Krajewski, 1991) and (Anagnostou, Krajewski, Seo, & Johnson, 1998) implemented the mean field bias estimation and correction techniques to remove the biases of radar precipitation estimates based on rain gauges estimates. (Seo & Breidenbach, 2002) described a procedure for real-time correction of spatially nonuniform bias in radar precipitation data using rain gauge measurements, which was proved generally superior to mean field bias correction.

For satellite precipitation estimates, McCollum and his group evaluated he biases of satellite precipitation estimation algorithms over the Continental United States. A bias-adjusted radar rainfall product was created in this study and used for evaluation of two satellite precipitation estimation algorithms (McCollum, Krajewski, Ferraro, & Ba, 2002). (Smith et al., 2006) compared two methods:

a direct method and an indirect method to evaluate and reduce the bias of satellite precipitation estimates (T. M. Smith, Arkin, Bates, & Huffman, 2006). While, (Tesfagiorgis, Mahani, Krakauer, & Khanbilvardi, 2011) tested an ensemble based method which aimed to estimate spatially varying multiplicative biases in satellite precipitation estimates using a radar-gauge precipitation product and compare it with three previous bias correction methods.

Most of the above mentioned works assumed that appropriate bias correction method can effectively reduce or remove the bias of climate models, forecast models, radar and satellite precipitation products.

2.2 Cumulative Distribution Functions (CDF)

The cumulative distribution function (CDF) gives the cumulative probability of a distribution. Specifically, it gives the area under the probability density function, up to the value specified. The CDF can be used to determine the probability of a response being lower than a certain value, higher than a certain value, or between two values ("Using the cumulative distribution function (CDF)," n.d.). This method is widely used for the bias correction in climate models ("Cumulative Distribution Functions (CDFs) in detail," n.d; Olsson et al., 2015), precipitation models (Duan, Selker, & Grant, 1998; Asefa & Adams, 2013; Franses & Koning, 2003), temperature and precipitation forecast products (Luo, Zhu, & Springs, 2011; US Department of Commerce, n.d.).

2.3. Triple collocation

Nowadays, a widely used method -- triple collocation (TC), can be used to quantify the error structures in large-scale datasets (Roebeling, Wolters, Meirink, & Leijnse, 2012). It is a method which can objectively obtains the error estimates for at least three or more independent products.

Firstly introduced by (Stoffelen, 1998) and (Caires, 2003), this method was applied to estimate near-surface and ocean wind speed errors, but later applied more in many hydrological applications and models. For example, (K. Scipal, Holmes, de Jeu, Naeimi, & Wagner, 2008) used triple collocation to estimate the errors of passive microwave (TRMM radiometer) derived, active microwave (ERS-2scatterometer) derived, and model-based (ERA-Interim reanalysis) soil moisture products. The results suggest that triple collocation provides realistic error estimates. Follow on, Klaus Scipal applied this powerful tool on other global soil moisture products, the scatterometer data, radiometer data and model data. The results show that triple collocation is robust and it allows to derive objective error estimates. (Klaus Scipal, Dorigo, & De Jeu, 2010)

Miralles et al. (Miralles et al., 2010) estimated the errors of footprint-scale soil moisture products, which are acquired from passive microwave remote sensing, ground-based station, and land surface model-based soil moisture products. The results shown that triple collocation estimates point-to-footprint soil moisture sampling errors to within 0.0059 m^3 m^{-3} and improve the ability to validate satellite footprint-scale soil moisture products.

Cimini et al. (Cimini et al., 2012) investigated the atmospheric columnar integrated water vapor using triple collocation, aiming to account for errors inherently present in every integrated water vapor measurements. (Thao & Eymard, 2014) applied triple collocation to analysis the trend and variability of the atmospheric water vapor.

Ratheesh et al. (Ratheesh, Mankad, Basu, Kumar, & Sharma, 2013) researched the performance of sea surface salinity (SSS) via taking triple collocation to account the error characteristics of the Soil Moisture and Ocean Salinity (SMOS) satellite, Argo floats, and model data sets. The results prove that SMOS data appears to be of very good quality in the equatorial Indian Ocean and southern Indian Ocean.

VanDijk et al. (van Dijk, Renzullo, Wada, & Tregoning, 2014) estimated the error estimates for the sequential data assimilation scheme using triple collocation, when presenting a global water cycle reanalysis from several satellites.

Roebeling et al. (Roebeling et al., 2012) applied triple collocation technique to precipitation products by estimating the spatial and temporal error characteristics of three different precipitation products --Retrievals from SEVIRI instrument, weather radar observation data and gridded rain gauge observations in Europe. The results suggest that the triple collocation method provides realistic error estimates.

3. Dataset

3.1. Study area

The study area is selected as the Netherlands. The country covers an areas of 41543 square kilometers, the geographic coordinates is 5.45° E and 50.30° N. The geography of the Netherlands is specific because that about half of the surface of the Netherlands is less than 1 m above sea level, about 27% of the land lies below sea level and has been reclaimed and protected by the dikes. The topography of the Netherlands is relatively flat. Due to the proximity of the ocean and the effect of the north Atlantic Gulf Stream, it belongs to the temperature zone climate with small climatological variations. The mean annual rainfall changes from 725mm to 925mm. Because of the coastal effects, the amount of precipitation are smaller in the coastal zone in spring and larger in late autumn (Attema & Lenderink, 2011).

3.2. Precipitation data

There are four kinds of precipitation data which have been used in the Netherlands. As we need a long-term (from 2003 to 2013) precipitation data and also with high spatial and temporal resolution, therefore, the precipitation products from CMORPH and PERSIANN are the appropriate choices. Table 3-1 shows the detail information and characteristics of four different precipitation products.

Variable	Satellite-based product	Temporal resolution	Spatial resolution	Start	End	Comments
Precipitation	*CMORPH*	*Weekly*	*0.25 degrees*	*1998*	*2013*	*-*
		Daily				
		3-hourly				
		1-hurly	*8 km*			
		30 min				
	PERSIANN	*3-hourly*	*0.25 degrees*	*March 2000*	*10 months before present*	*Spatial coverage is: 60° to -60° lat 0° to 360° long*
		6-hourly	*0.25 degrees*	*March 2000*	*3 months before present*	*Spatial coverage is: 50° to -50° lat 0° to 360° long*
	MSG-CPP	*15 minute*	*3 km*	*2012*	*Ongoing*	*Spatial coverage is: 50° to -50° lat 80° to -80° long*
	H-SAF PR-OBS-05	*15 minute*	*Average over Europe: 8 km intended as sampling, ~ 30 km effective*	*2009 onwards*	*Ongoing*	*Spatial coverage is: 25-75°N lat, 25°W-45°E long*

Figure 3-1 Comparison of different precipitation products.

3.2.1. CMORPH precipitation data

The Climate Prediction Center morphing method (CMORPH) is a technique which can produce global precipitation analyses at a very high spatial and temporal resolution. It uses the relatively high quality precipitation estimates obtained from low orbiter satellite microwave observations, whose features are propagated via spatial information that is obtained entirely from half-hourly interval geostationary satellite infrared imagery (Joyce, Janowiak, Arkin, & Xie, 2004).

The COMRPH data used in this paper were downloaded from The National Oceanic and Atmosphere Administration (NOAA) Climate Prediction Center with 3-hourly temporal resolution and 25Km spatial resolution.

In order to automatically import and pre-process the CMORPH 0.25DEG 3HLY RAW data in ILWIS, a work flow was built, which contains two bath files. Batch file 1 was built to import the raw data, rename them and sequentially import the data in ILWIS and pre-process. Batch file 2 was built to read raw binary data in IWLIS maplist, mirror rotate the maplist and recompose the global map. Finally, after setting the georeference, the long set of recomposed files were completely accomplished. Thus, the CMORPH data of the Netherlands from 2003 to 2013, with 3-hourly temporal resolution and 25Km spatial resolution are available now. The complete batch files are added at the appendix of this paper.

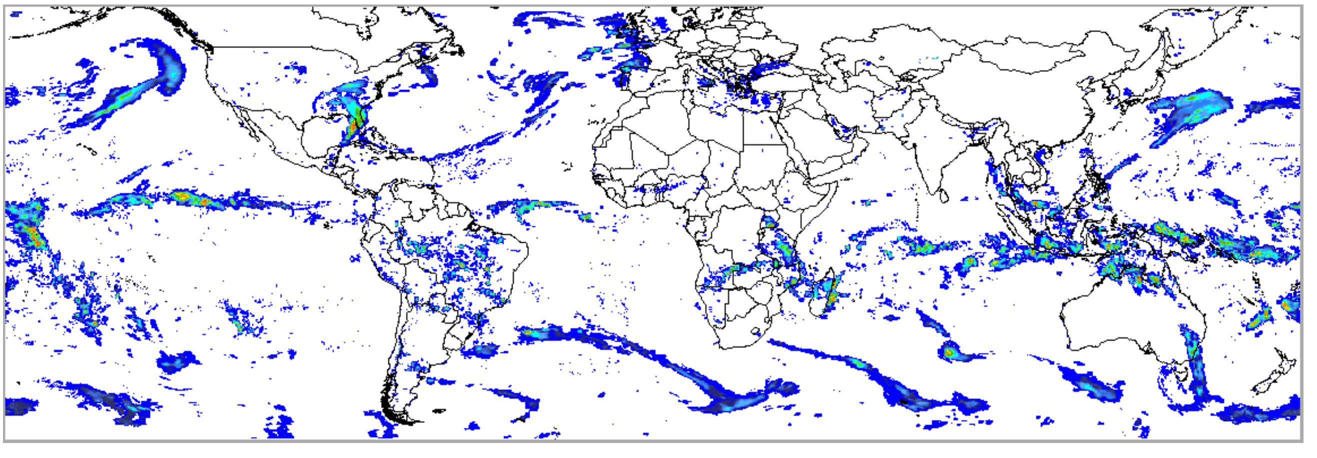

Figure 3-2 Global CMORPH 0.25 degree-3 hourly data of 20030101 at 0:00 UTC

3.2.2. PERSIANN precipitation data

The Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks (PERSIANN) system provides near real-time precipitation information, which started from March of 2000 with 0.25° spatial resolution in a consistent long-term record of remotely sensed precipitation observations. It uses neural network function classification/approximation procedures to estimate the precipitation rates at each 0.25° x 0.25° pixel of the long wave infrared brightness temperature image, which are provided by global geosynchronous satellites. The spatial coverage differs from the temporal resolution, for 3-hourly temporal resolution, the spatial coverage is 60° to -60° lat and 0° to 360° long. For 6- hourly temporal resolution, the spatial coverage is 50° to -50° lat and 0° to 360° long (“Satellite Precipitation - CHRS,” n.d.).

The PERSIANN precipitation data were download from Center for Hydrometeorology and Remote Sensing (CHRS) which is with 3-hourly temporal resolution and 25Km spatial resolution. The data was processed and recomposed using the same method with processing the COMRPH data. Thus, the PERSSIAN data of the Netherlands from 2003 to 2013, with 3-hourly temporal resolution and 25Km spatial resolution are available now.

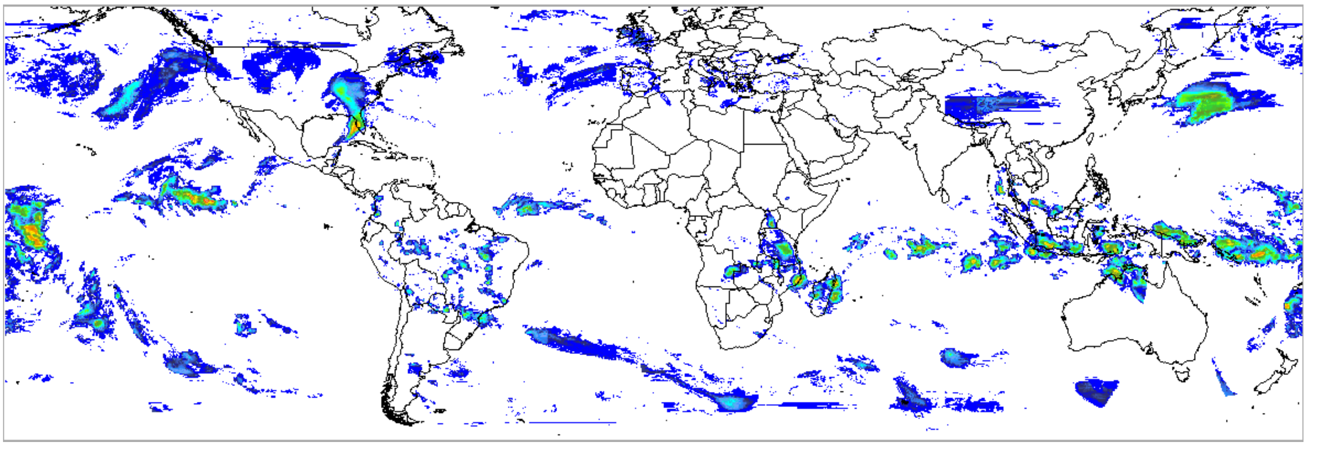

Figure 3-3 Global PERSSIAN 0.25 degree-3 hourly data of 20030101 at 0:00 UTC

3.2.3. In situ rain gauge data

The in situ rain gauge data is provided by The Royal Netherlands Meteorological Institute (KNMI) from the Klimaat Informatie Systeem (KIS) database. KNMI provides two independent rain gauge networks: the manual voluntary precipitation stations (approximately 325 gauges, ≈ 1 station per 100 km) and the automatic meteorological stations (approximately 37 gauges, ≈ 1 station per 1000 km^2). The manual voluntary precipitation stations measure the precipitation once a day at 08:00 UTC, while the meteorological automatic stations provide daily data and hourly data. As the datasets needed for this study is 3 hourly, therefore, the in situ rain gauge data from 37 automatic meteorological stations are used, except that one new station (station number is 215) began from 2011, one station (station number is 265) stopped at 2008, and 3 stations (stations’ number are 225, 242, 340) do not have hourly data. Eventually, there are 32 stations’ data available for the study. The locations of the 32 available automatic meteorological stations are shown in figure 3-4

Figure 3-4 Locations of 32 available automatic meteorological stations

4. Methodology

4.1. Flow chart

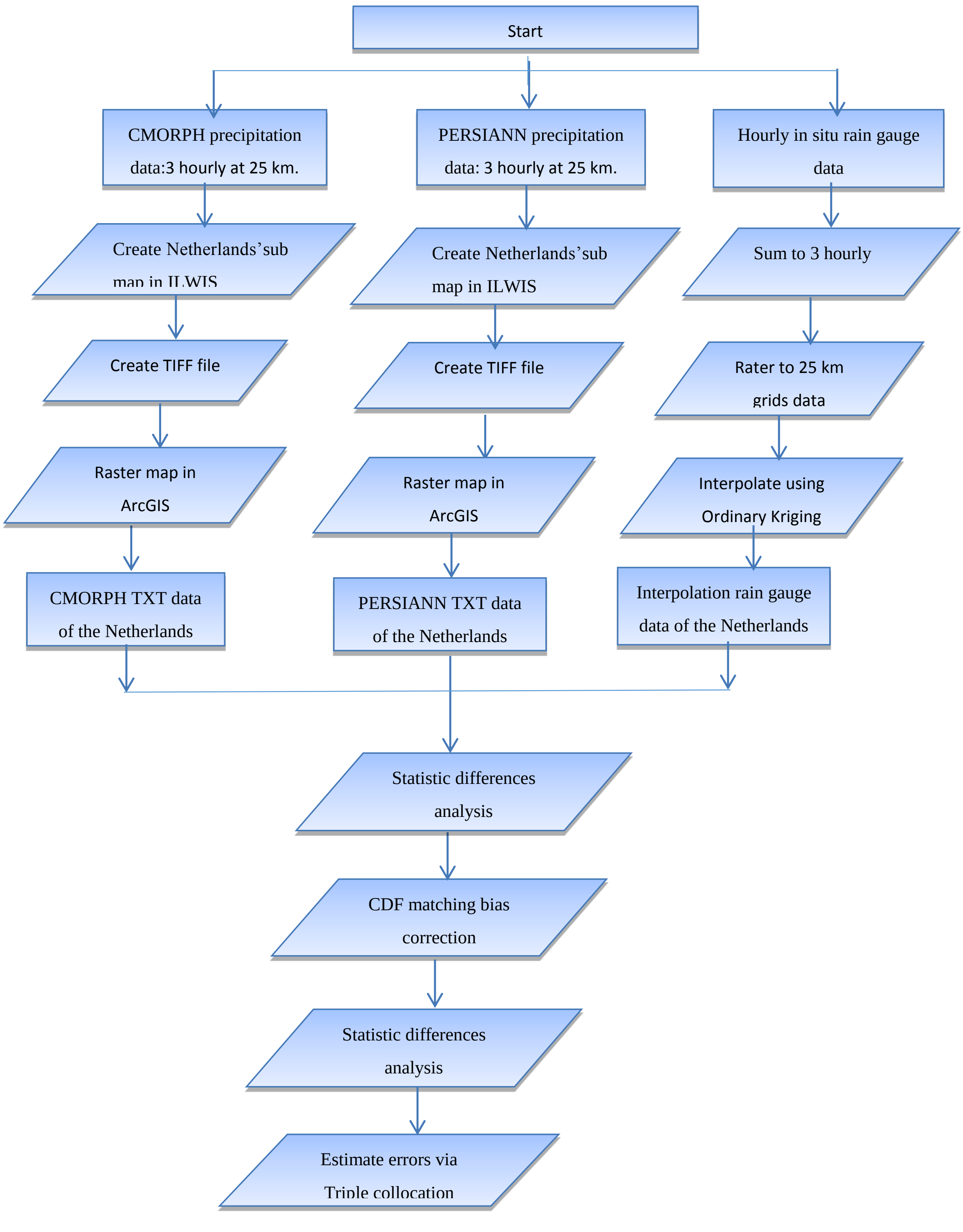

Figure 4-1 Flow chart of methodology

4.2. Create sub maps

Getting the Netherlands sub map from the global CMORPH precipitation map can be achieved by the ILWIS. Using the “File Menu”>“Create”>“Submap”>“Submap from Raster” to create a new sub map and give a new name to it. See figure 4-2 and figure 4-3 below.

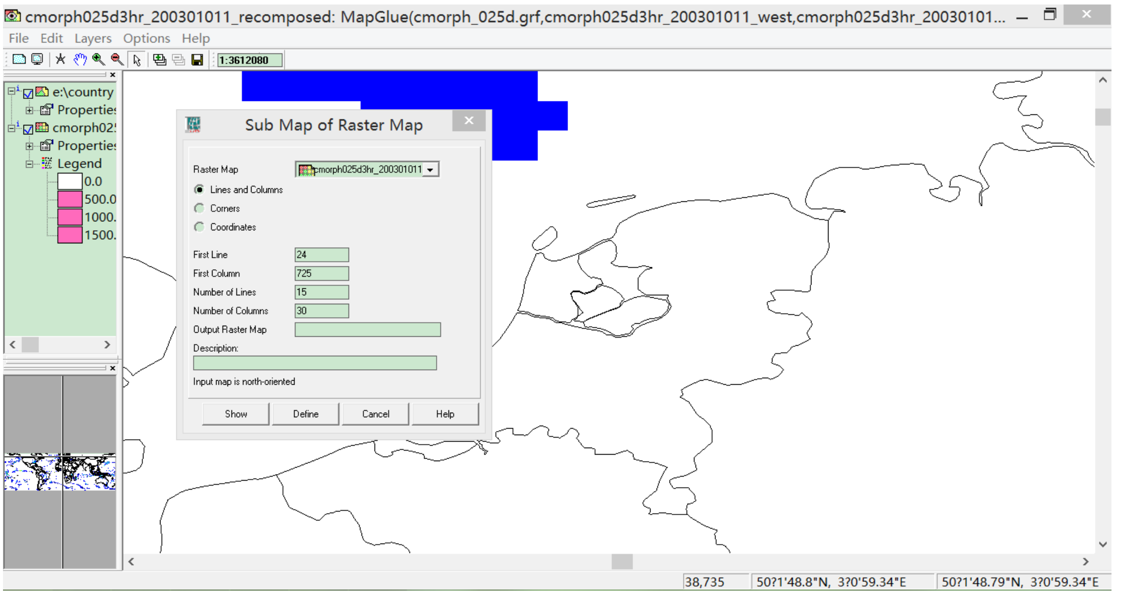

Figure 4-2 Create a sub map of the Netherlands

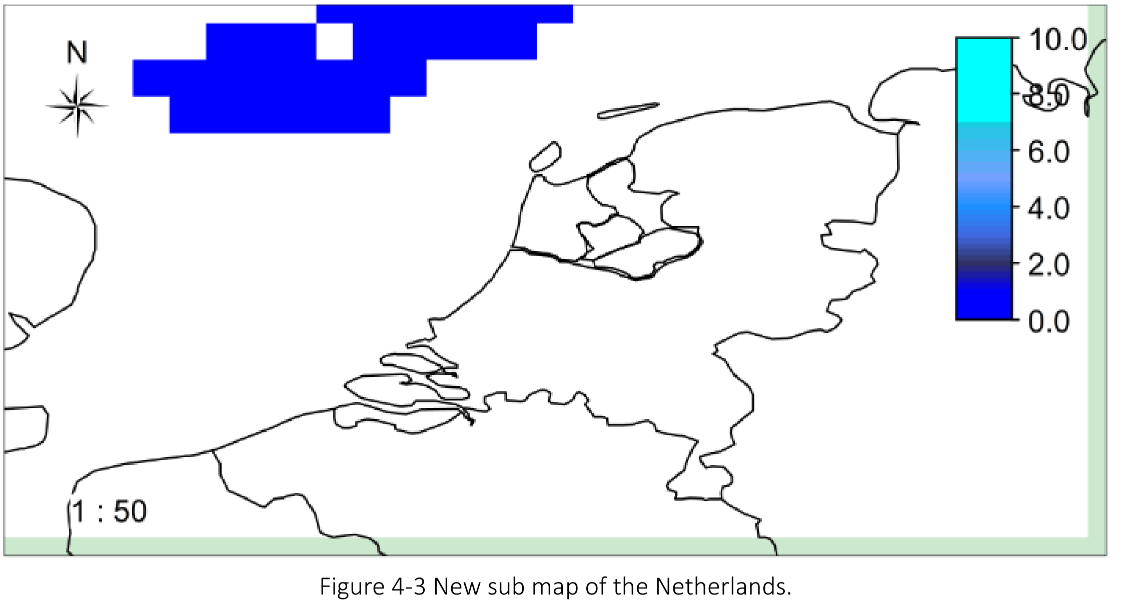

Figure 4-3 New sub map of the Netherlands.

As the CMORPH data needed for this study is 11 years from 2003 to 2013 and temporal resolution is 3 hourly. So there is a super large number of maps need to be edited using the same method. To improve the efficiency, a script was written to create the sub maps for a whole month. Thus, all the maps over the Netherlands can be created month by month. The syntax is:

```
Cmorph025d3hr_200301011.mpr = MapSubMap(cmorph025d3hr_200301011_recomposed,24,725,15,30)
```

Where the “Cmorph025d3hr_200301011.mpr” is the sub map name while “cmorph025d3hr_200301011_recomposed” is the original global map. The parameter 24, 725, 15, 30 are the first line, first column, number of lines, number of columns respectively.

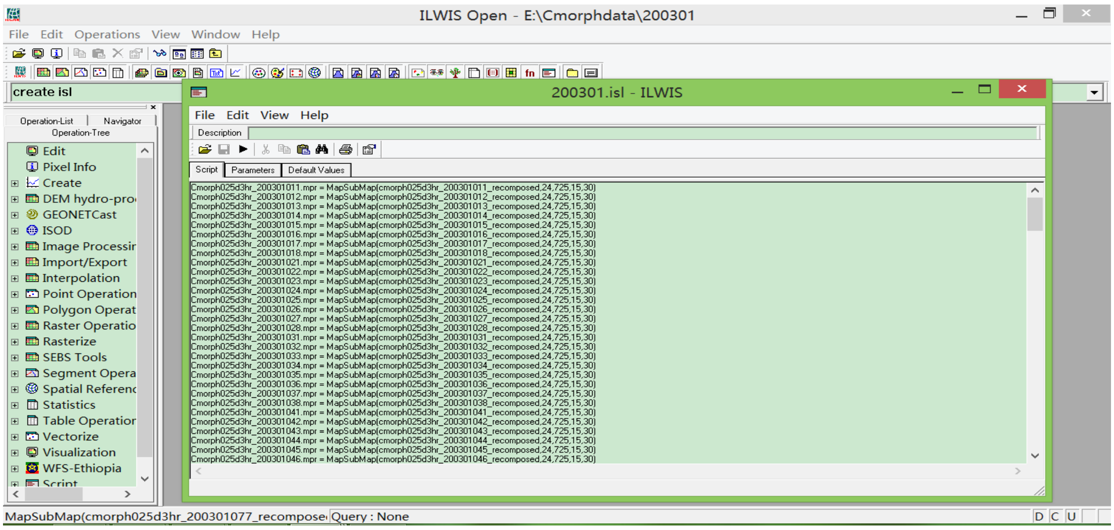

Figure 4-4 Create all sub maps using script batch file

4.3. Create TIFF file

For subsequently processing the raster map using Arc GIS, the map should be converted to raster type, which is the TIFF file. The syntax is:

export TIFF(cmorph025d3hr_200301011.mpr,G:\Graduation\TIFCMORPH\200301\cmorph025d3hr_200301011) .

Again, a script was created to process the batch file for each month then all the sub map will be exported to the TIFF file.

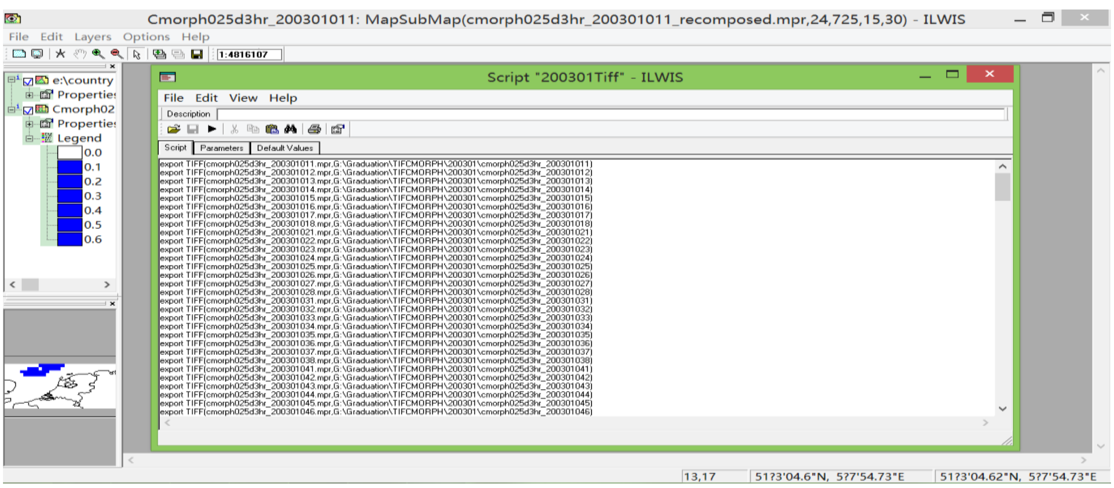

Figure 4-5 Create all TIFF files using script batch file

4.4. Raster map

Raster progress is processed to define the original coordinate system of the sub map, transform to the local coordinate system and extract the Netherlands out using the shape file. In this thesis, an

ArcGIS model is built to achieve this objective as the figure showing below.

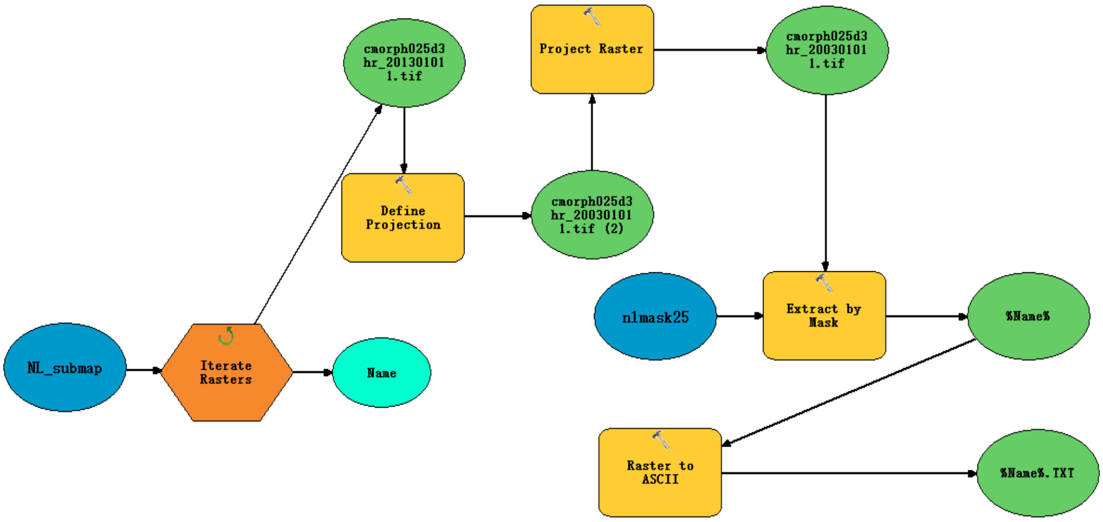

Figure 4-6 Flow chart of raster map

The first step is to import the sub map and define the coordinate system information stored with the dataset. Then using the projection raster tool to transform the original coordinate system WGS 84 (World Geodetic System 1984) to RD_New (Netherlands Projection). After that, extract the cells of a raster that corresponds to the areas defined by the Netherlands' mask. Finally, convert the raster dataset to the ASCII text file representing raster data. The result was shown in the figure below. Run the model for all of the CMORPH dataset and arrange them as the time list.

```
ncols         11
nrows         13
xllcorner     0
yllcorner     300000
cellsize      25000
NODATA_value  -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 0 -9999 -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 0 0 0 0
-9999 -9999 -9999 -9999 0 -9999 0 0 1 0 0
-9999 -9999 -9999 -9999 0 -9999 -9999 0 1 0 0
-9999 -9999 -9999 -9999 0 -9999 -9999 0 0 0 -9999
-9999 -9999 -9999 -9999 0 -9999 0 0 0 0 0
-9999 -9999 -9999 0 0 0 0 0 0 0 -9999
-9999 -9999 0 0 0 0 0 0 0 0 -9999
-9999 -9999 -9999 0 0 0 0 0 -9999 -9999 -9999
-9999 0 0 0 -9999 0 0 0 -9999 -9999 -9999
-9999 -9999 0 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 1 -9999 -9999 -9999

```

Figure 4-7 Raster datasets in TXT file

The structure of the ASCII file consists of header information containing a set of keywords, followed by cell values in row-major order. NCOLS and NROWS are the number of columns and rows in the raster defined by the ASCII file. XLLCORNER and YLLCORNER are the coordinates of the lower left corner of the lower left cell. CELLSIZE is the cell size of the raster. NODATA_VALUE is the value that is to represent No Data cells.

4.5. Interpolation of rain gauge data

4.5.1. Sum to 3 hourly data

The rain gauge data downloaded from KNMI website include data of temperature, sun hours, clouds and visibility, barometric pressure, wind speed and precipitation. Select the hourly precipitation amount data (RH) of 32 stations and sum to 3 hourly using Excel.

```
BRON:
KONINKLIJK NEDERLANDS METEOROLOGISCH INSTITUUT (KNMI)

SOURCE:
ROYAL NETHERLANDS METEOROLOGICAL INSTITUTE (KNMI)

YYYYMMDD  = datum (YYYY=jaar,MM=maand,DD=dag) / date (YYYY=year,MM=month,DD=day)
HH        = tijd (HH=uur, UT.12 UT=13 MET, 14 MEZT. Uurvak 05 loopt van 04.00 UT tot 5.00 UT / time (HH uur/hour, UT. 12 UT=13 MET, 14 MEZT. Ho
DD        = Windrichting (in graden) gemiddeld over de laatste 10 minuten van het afgelopen uur (360=noord, 90=oost, 180=zuid, 270=west, 0=wind
FH        = Uurgemiddelde windsnelheid (in 0.1 m/s) / Hourly mean wind speed (in 0.1 m/s)
FF        = Windsnelheid (in 0.1 m/s) gemiddeld over de laatste 10 minuten van het afgelopen uur / Mean wind speed (in 0.1 m/s) during the 10-m
FX        = Hoogste windstoot (in 0.1 m/s) over het afgelopen uurvak / Maximum wind gust (in 0.1 m/s) during the hourly division
T         = Temperatuur (in 0.1 graden Celsius) op 1.50 m hoogte tijdens de waarneming / Temperature (in 0.1 degrees Celsius) at 1.50 m at the
T10N      = Minimumtemperatuur (in 0.1 graden Celsius) op 10 cm hoogte in de afgelopen 6 uur / Minimum temperature (in 0.1 degrees Celsius) at
TD        = Dauwpuntstemperatuur (in 0.1 graden Celsius) op 1.50 m hoogte tijdens de waarneming / Dew point temperature (in 0.1 degrees Celsius
SQ        = Duur van de zonneschijn (in 0.1 uren) per uurvak, berekend uit globale straling  (-1 for <0.05 uur) / Sunshine duration (in 0.1 hou
Q         = Globale straling (in J/cm2) per uurvak / Global radiation (in J/cm2) during the hourly division
DR        = Duur van de neerslag (in 0.1 uur) per uurvak / Precipitation duration (in 0.1 hour) during the hourly division
RH        = Uursom van de neerslag (in 0.1 mm) (-1 voor <0.05 mm) / Hourly precipitation amount (in 0.1 mm) (-1 for <0.05 mm)
P         = Luchtdruk (in 0.1 hPa) herleid naar zeeniveau, tijdens de waarneming / Air pressure (in 0.1 hPa) reduced to mean sea level, at the
VV        = Horizontaal zicht tijdens de waarneming (0=minder dan 100m, 1=100-200m, 2=200-300m,..., 49=4900-5000m, 50=5-6km, 56=6-7km, 57=7-8km
N         = Bewolking (bedekkingsgraad van de bovenlucht in achtsten), tijdens de waarneming (9=bovenlucht onzichtbaar) / Cloud cover (in octan
U         = Relatieve vochtigheid (in procenten) op 1.50 m hoogte tijdens de waarneming / Relative atmospheric humidity (in percents) at 1.50 m
WW        = Weercode (00-99), visueel(WW) of automatisch(WaWa) waargenomen, voor het actuele weer of het weer in het afgelopen uur. Zie http://
IX        = Weercode indicator voor de wijze van waarnemen op een bemand of automatisch station (1=bemand gebruikmakend van code uit visuele wa
M         = Mist 0=niet voorgekomen, 1=wel voorgekomen in het voorgaande uur en/of tijdens de waarneming / Fog 0=no occurrence, 1=occurred duri
R         = Regen 0=niet voorgekomen, 1=wel voorgekomen in het voorgaande uur en/of tijdens de waarneming / Rainfall 0=no occurrence, 1=occurre
S         = Sneeuw 0=niet voorgekomen, 1=wel voorgekomen in het voorgaande uur en/of tijdens de waarneming / Snow 0=no occurrence, 1=occurred d
O         = Onweer 0=niet voorgekomen, 1=wel voorgekomen in het voorgaande uur en/of tijdens de waarneming / Thunder  0=no occurrence, 1=occurr
Y         = IJsvorming 0=niet voorgekomen, 1=wel voorgekomen in het voorgaande uur en/of tijdens de waarneming / Ice formation 0=no occurrence,
```

Figure 4-8 Select the hourly precipitation amount from rain gauge data

Create a MATLAB model to integrate the complex data to clear 3 hourly data for everyday over 11 years with 32 rain gauge stations' number on the leftmost row and the stations' name on the rightmost row. Here is an example of integrated 3 hourly precipitation showing below (Figure 4-9). The MATLAB syntax is attached to the appendix at the end of this thesis.

# DEZE GEGEVENS MOGEN VRIJ WORDEN GEBRUIKT MITS DE VOLGENDE BRONVERMELDING WORDT GEGEVEN:																																	
# KONINKLIJK NEDERLANDS METEOROLOGISCH INSTITUUT (KNMI)																																	
# THESE DATA CAN BE USED FREELY PROVIDED THAT THE FOLLOWING SOURCE IS ACKNOWLEDGED:																																	
# ROYAL NETHERLANDS METEOROLOGICAL INSTITUTE																																	
# Dagelijkse stationsneerslag																																	
#	NR	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Locatie
	210	0	0	0	0	0	1.9	0	0.1	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	1.6	0.4	0	Valkenbur
	235	0	0	0	0	0	0.4	0	0	0	0	0	0	1	0	0	0	0	0	0	0.1	0	0	0	0	0	0	0	0.3	0.3	1	0	DeKooy
	240	0	0.4	0	0	0	0.2	0	0	0	0	0	0	1.6	0	0	0	0	0	0	0.4	0.2	0.5	0.1	0	0	0	0	0	0.5	0.9	0	Schiphol
	249	0	0	0.1	0	0	0.3	0	0	0	0	0	0	2.2	0	0	0	0	0	0	0.3	0	0	0	0	0	0	0	0.1	0.2	0.2	0	Berkhout
	251	0	0	2	0	0	0	0	0	0	0.1	0	0	0.9	0	0	0	0	0	0	0.1	0	0	0	0	0	0	0	0.1	0.4	0	0	Hoorn(Ter
	257	0	0	1.2	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0.1	0.1	0	0.1	0	0	0	0	0	0.3	0.7	0	WijkaanZe
	260	0	0.2	0	0	0	0.9	0	0	0	0	0	0	0.2	0	0	0	0	0.2	0	0.1	1.6	1.6	0	0	0	0	0	1.2	2	0.9	0	De Bilt
	267	0	0	0.3	0	0	0	0	0.3	0	0.1	0	0	2.3	0	0	0	0	0	0	0.1	0.4	0	0	0	0	0	0	0.1	0	0	0	Stavoren
	269	0	0	1.4	0	0	0	0	0.2	0	0	0	0	1.2	0	0	0	0	0.4	0	0.6	1.7	0.8	0.1	0	0	0	0	0.2	0.1	0.1	0	Lelystad
	270	0	0	1.6	0	0	0	0	0.4	0	0	0	0	1.5	0	0	0	0	0.1	0	0.5	0.6	0	0	0	0	0	0	0.3	1.8	0	0	Leeuwarde
	273	0	0	0.5	0	0	0	0	1.1	0	0	0	0	3.3	0	0	0	0	0.5	0	0	1.8	0.1	0	0	0	0	0	0.8	0.2	0	0	Marknesse
	275	0	0.7	1.5	0	0	0	0	0.1	0	0	0	0	0	0	0	0	0	0.5	0	0.2	1.1	0.1	1.1	0	0	0	0	2.4	1.8	0	0	Deelen
	277	0	0	1.5	0	0	0	0	0	0	0	0	0.4	2.2	0	0	0	0	0.3	0	0.1	0.4	0	0	0	0	0	0	0.2	1.1	0.4	0	Lauwersoo
	278	0	0.1	1	0	0	0	0	0.3	0	0	0	0	0.7	0	0	0	0	0.2	0	0	0.4	0	0.2	0	0	0	0	2.2	1.1	0	0	Heino
	279	0	0	1.4	0	0	0	0	0	0	0	0	0	1.8	0	0	0	0	0.2	0	0	0.2	0	0	0	0	0	0.1	4	0.4	0.1	0	Hoogeveen
	280	0	0	1.9	0	0	0	0	0.6	0	0	0	0.3	3.7	0	0	0	0	0.3	0	0.1	0.8	0	0.2	0	0	0	0.1	0.1	1.6	0.1	0	Eelde
	283	0	0.5	0.5	0	0	0.8	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0.8	0.1	0.1	0	0	0	0.3	4	4.2	0.1	0	Hupsel
	286	0	0	3.7	0	0	0	0	0	0	0	0	0	3.4	0	0	0	0	0.1	0	0	0.4	0	0	0	0	0	0.4	0.5	0.2	0.1	0	NieuwBeer
	290	0	2.7	0.5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0.1	0	0.7	0.5	0	0.1	0	0	0	0.2	2.9	1.5	1.1	0	Twenthe
	310	0	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.2	0.1	0	0	0	0	1.9	1.9	2.2	0	Vlissinge
	319	0	0	2.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0.1	0.2	0	0	0	0	3.2	1.9	1.2	0	Westdorpe
	323	0	0.1	2.1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0.4	0.3	0	0.2	0.4	0	0	0	0	2.3	0.9	1.9	0	Wilhelmin
	330	0	0	0	0	0	0.3	0	0	0	0	0	0	1	0	0	0	0	0	0.1	0	0	0	0.5	0	0	0	0	2.5	0.2	0	0	HoekvanHo
	344	0	0	0	0	0	1.2	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0.2	0	0	0.4	0	0	0	0	1.8	0.5	0.4	0	Rotterdam
	348	0	0	0	0	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0.3	0.6	0.9	0.1	0	0	0	0	1.5	1.2	1.1	0.1	Cabauw
	350	0	1.4	0	0	0	0.5	0	0	0	0	0	0	0.3	0	0	0	0	0.1	0	0.5	0.4	1	0.3	0	0	0	0	5.6	1.7	0.8	0	GilzeRije
	356	0	0.2	0	0	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0.1	0.7	1.6	0.3	0	0	0	0	2.9	1.8	1.1	0	Herwijnen
	370	0	1.2	0	0	0	0.3	0	0	0	0	0	0	0.2	0	0	0	0	0.1	0	0.2	0	0.4	0	0	0	0	0.2	5	2.3	1.3	0	Eindhoven
	375	0	1.4	0	0	0	0.3	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0.4	0.2	0.3	0	0	0	0	0	3.7	3.3	0.6	0	Volkel
	377	0	0.9	0.5	0	0	0	0	0	0	0	0	0	0.1	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	3.7	1.8	0	0	Ell
	380	0	0.6	1.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	3.2	1.6	1.3	0	Maastrich
	391	0	1.1	0	0	0	0.3	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0.4	0.5	0	0	0	0	0	0	5	2	0.1	0	Arcen

Figure 4-9 Precipitation data for all 31 days of 200301(January) at 0:00 UTC

4.5.2. Interpolation

Interpolation is defined as, estimation of a variable at an unmeasured location from observed values at surrounding locations (Bohling, 2005). As the data collected from in situ rain gauges is discrete and random, therefore spatial interpolation is necessary for creating a continuous dataset. Until now, there has been a number of studies on analysis of the difference between various interpolation methods. Goovaerts (Goovaerts, 2000) compared different interpolation techniques including Thiessen polygon, inverse distance weighting (IDW) and ordinary kriging using annual and monthly rainfall observations. Kao & Hung (Kao & Hung, 2004) evaluated over twelve interpolation methods using the 5 meter DTM as test data. Hofstra et al. (Hofstra, Haylock, New, Jones, & Frei, 2008) compared six interpolation methods for the interpolation of daily precipitation, mean, temperature and sea level pressure from station data over the Europe. Based on the extensive review of literatures, kriging interpolation method is regarded as the optimum one. Ordinary kriging is one of kriging interpolation approaches, which measures unknown values by linear combination of random variables, using semivariogram analysis to estimate the weight of data and describe the spatial correlation. In this thesis, the ordinary kriging is selected to interpolate the in situ rain gauge management data through R Scripts.

R is a free software environment for statistical computing and graphics, which uses packages for geospatial analysis (R-Project). The interpolation is processed with R studio. Also the 25 km grid map is applied on R syntax to raster 1km grids to 25m. Mask map of the Netherlands is applied to extract the cells of a raster that correspond to the areas defined by the mask map. The R syntax is attached to the appendix at the end of this thesis.

The result is showed on the figure below. It is the interpolation precipitation on 0:00 UTC of 1st January of 2003.

```
NCOLS 11
NROWS 13
XLLCORNER 0
YLLCORNER 3e+05
CELLSIZE 25000
NODATA_VALUE -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 -9999 0 -9999 -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 0 0 0 0
-9999 -9999 -9999 -9999 0 -9999 0 0 0 0 0
-9999 -9999 -9999 -9999 0 -9999 -9999 0 0 0 0
-9999 -9999 -9999 -9999 0 -9999 -9999 0 0 0 -9999
-9999 -9999 -9999 -9999 0 -9999 0 0 0 0 0
-9999 -9999 -9999 0 0 0 0 0 0 0 -9999
-9999 -9999 0 0 0 0 0 0 0 0 -9999
-9999 -9999 -9999 0 0 0 0 0 -9999 -9999 -9999
-9999 0 0 0 -9999 0 0 0 -9999 -9999 -9999
-9999 -9999 0 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999
-9999 -9999 -9999 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999

```

Figure 4-10 The interpolation precipitation data of 1st January of 2003 at 0:00 UTC

The structure of the ASCII (txt) file consists of header information containing a set of keywords, followed by cell values in row-major order. NCOLS and NROWS are the number of columns and rows in the raster defined by the ASCII file. XLLCORNER and YLLCORNER are the coordinates of the lower left corner of the lower left cell. CELLSIZE is the cell size of the raster. NODATA_VALUE is the value that is to represent No Data cells.

4.5.3 CDF Matching

As precipitation data set derived from satellite is characterized by its specific value and dynamical range. Therefore, satellite data always require scaled before their actual use within hydrological or meteorological models (K Scipal, Drusch, and Wagner, 2008). Generally, the most widely used scaling technique is the cumulative distribution function (CDF) matching approach (Drusch, 2005).

The cumulative distribution function (CDF) evaluated at "x", which is the probability that a real-valued random variable X will take a value less than or equal to x. In another word, it means CDF(x) = Pr(*X*<=x), where Pr represents probability. In this paper, the CDF matching technique is used to adjust two satellite observations against the interpolation precipitation products and applied for each 25km pixel individually.

4.6. Error estimation using triple collocation

Triple collocation can be used to estimate the random error variance in three collocated datasets of the same geophysical variable (Stoffelen, 1998). Triple collocation assumes the following error model for each time series:

$$R = \alpha + \beta R_t + \varepsilon$$

Assume Rt is the true value of precipitation, α and β are additive and multiplicative biases of the data and ε is the relative errors which we want to estimate. In order to Estimate the relative error ε, it is necessary to scale or calibrate the datasets to the reference dataset (removing α and β) and calculating the relative error based on these datasets.

Datasets from three precipitation products related to the truth are assumed as Rx, Ry, and Rz, based on the defined error model:

$$R_x = \alpha_1 + \beta_x R_t + \varepsilon_x$$

$$R_y = \alpha_y + \beta_y R_t + \varepsilon_y$$

$$R_z = \alpha_z + \beta_z R_t + \varepsilon_z$$

Using mean-standard deviation scaling to bring the data to the same mean and standard deviation as the reference dataset. When the three datasets have the same mean and standard deviation, the unknown true value of precipitation can be removed:

$$R_x^*-R_y^* = \varepsilon_x^*- \varepsilon_y^*$$

$$R_x^*-R_z^* = \varepsilon_x^*- \varepsilon_z^*$$

$$R_y^*-R_z^* = \varepsilon_y^*- \varepsilon_z^*$$

From these three scaled datasets, cross multiplying above equations, the mean variance of relative errors can be fully determined by three independent and calibrated precipitation estimates using the following equations:

$$\langle(\varepsilon_x^*)^2 \rangle= \langle(R_x^*-R_y^*)(R_x^*-R_y^*) \rangle$$

$$\langle(\varepsilon_y^*)^2 \rangle= \langle(R_y^*-R_x^*)(R_y^*-R_z^*) \rangle$$

$$\langle(\varepsilon_z^*)^2 \rangle= \langle(R_z^*-R_x^*)(R_z^*-R_y^*) \rangle$$

Among above equations, the ⟨⟩brackets mean the temporal mean, ε_x^*, ε_y^*, ε_z^* are the relative errors.

5. Result and discussion

5.1 Statistic difference among different precipitation products

The statistic difference will be investigated by calculating the correlation coefficient and the root mean square error (RMSE). The correlation coefficient is a coefficient that elucidates a quantitative measure of some type of correlation and dependence, it measures the statistical strength of association between two random variables or observed data values. The most common correlation coefficient, called the Pearson product-moment correlation coefficient, which measures the strength of the linear relationship between variables. ("Correlation coefficient - Wikipedia, the free encyclopedia,"). In this paper, the correlation coefficient between CMORPH precipitation products vs PERSIAAN precipitation products, CMORPH precipitation products vs Interpolation rain gauge products, PERSIANN precipitation products vs Interpolation rain gauge products are estimated respectively. The results are shown at the figure below.

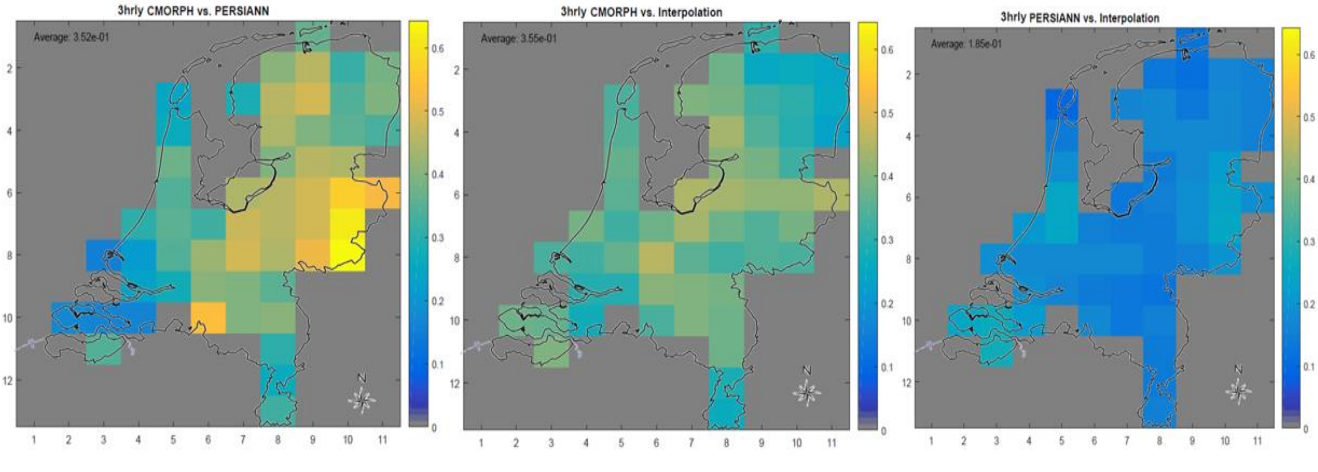

Figure 5-1 The correlation coefficient of each two precipitation products in 3 hourly temporal resolution

Generally, the value of a correlation coefficient can range between -1 and 1 and the weakest linear relationship is indicated by a correlation coefficient equal to 0. The greater the absolute value of a correlation coefficient, the stronger the linear relationship between two variables. From the figure, it can be seen that the average correlation coefficient of CMORPH vs. PERSIANN is 0.352, the value of CMORPH vs. interpolation is 0.355, and of PERSIANN vs. interpolation is 0.185. The result proves that the correlation between CMORPH and interpolation datasets is higher than the other two pairs, while the correlation between PERSIANN and interpolation datasets is the weakest one.

5.2 Root Mean Square Error

The Root Mean Square Error is common used for the measurement of the difference between values predicted by a model or an estimation and the values actually observed. The RMSE represents the sample standard deviation of the differences between predicted values and observed values. It makes an excellent general purpose error metric for numerical predictions. In this thesis, RMSE will be calculated by taking the root of the sum of all squares of the differences between each individual pixels, divided by the total number of pixels:

$$\mathrm{RMSE} = \sqrt{\frac{\sum(A_i - B_i)^2}{n}}$$

Where i denotes for each individual pixel and n represents the total number of pixels. If the spatial distribution of the datasets is same, the values will be zero. The lower root-mean-squared error (RMSE) indicates the product is likely to give more reliable estimation values reference to the interpolation product. The results are shown at figure 5-2, the average root-mean-squared error of CMORPH vs. interpolation is 3.35, while for PERSIANN vs. interpolation the value is 4.13. Therefore, the CMORPH's estimation is likely more reliable than PERSIANN's.

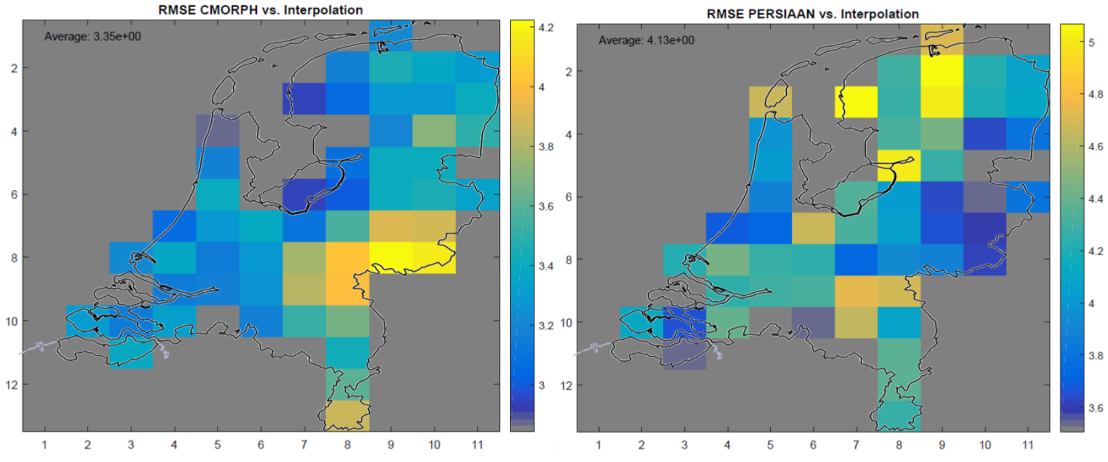

Figure 5-2 Root Mean Square Error of 3 hourly CMORPH and PERSIANN reference to interpolation.

5.3 Data histogram

In order to further observe the three precipitation products, a pixel which from row 5 and column 8 from each products is selected out to do the analysis. Histogram is a kind of statistical graphical representation of the frequency distribution of data grouped into classes, which comes from a continuous probability distribution. It consists of a series of high longitudinal stripes or lines representing data distribution. As it is shown in histogram figure 5-3, for CMORPH data, the frequency of 0 to 5 mm precipitation close to 1700, for PERSIANN data the frequency is almost the same with CMORPH, but for interpolation data, the frequency is close to 14000. There appears a great difference between two satellite estimation and interpolation estimation because of the satellite's unsuccessful retrievals of precipitation for the relatively low precipitation amount. In addition, the histogram also provides the information that in the Netherlands, the precipitation amount mostly concentrates on 0-5mm for 3hourly products.

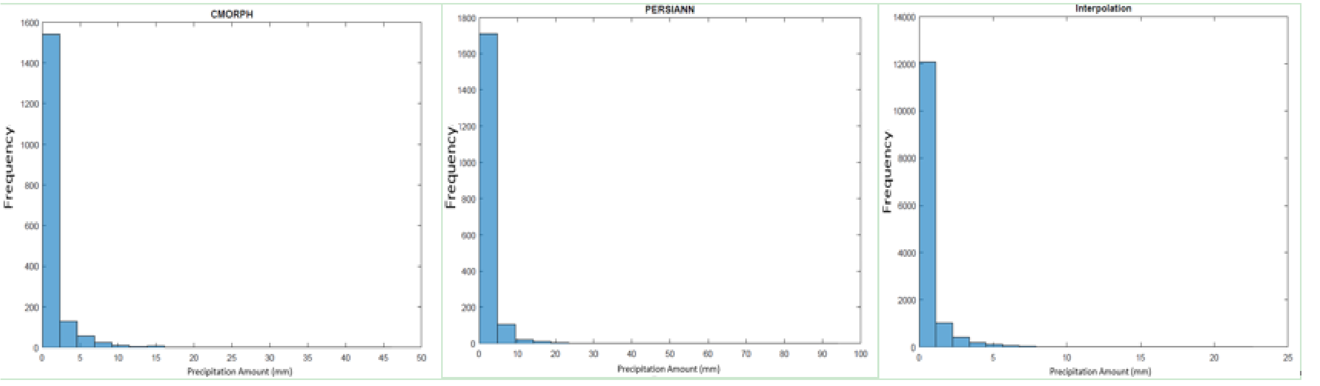

Figure 5-3 Histogram of one pixel from 3 hourly CMORPH, PERSIANN and interpolation precipitation products

5.4 Cumulative Distribution Function (CDF) plot

An empirical CDF plot is a graph that can be used to evaluate the fit of a distribution of the observation data, estimate percentiles, and compare different sample distributions. The CDF plot below clearly shows the fitness of CMMORPH, PERSIANN and interpolation products, with the precipitation value on the X-coordinate and the percentage of values on the Y-coordinate. Obviously, the two satellite products' fitness is better than they each compares to interpolation products.

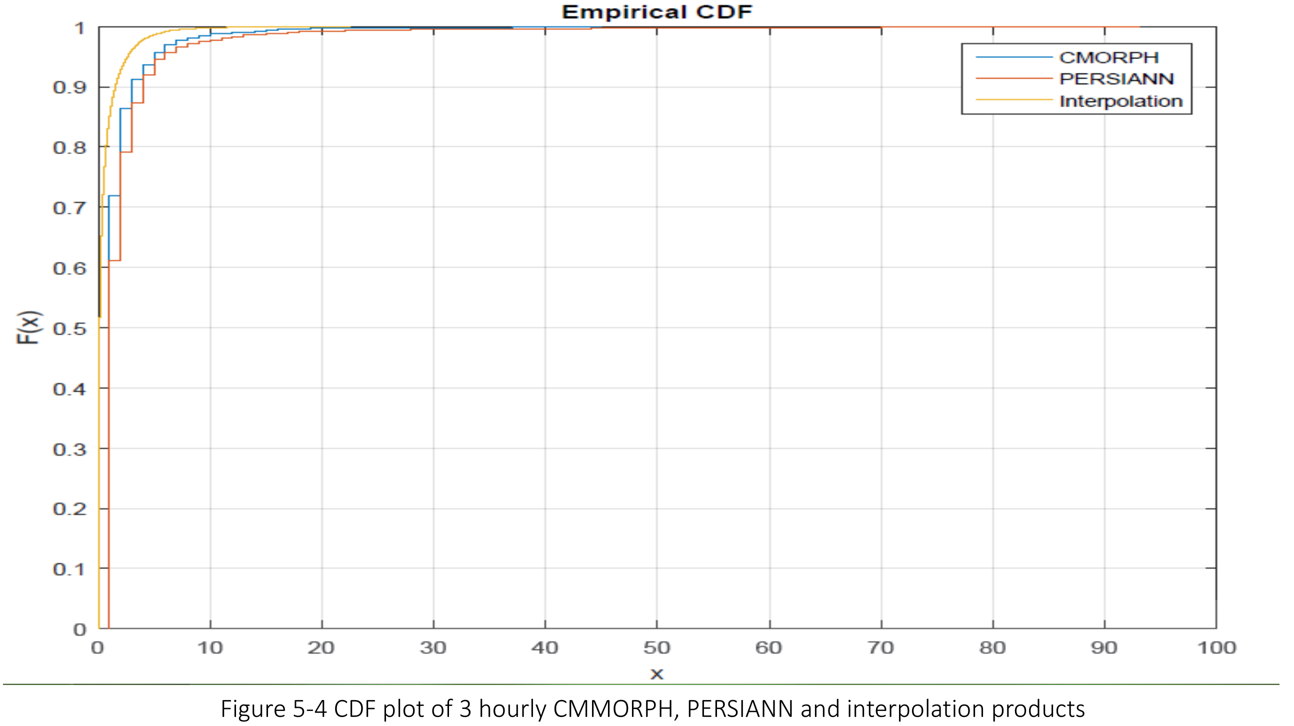

Figure 5-4 CDF plot of 3 hourly CMMORPH, PERSIANN and interpolation products

5.5 Bias correction (CDF matching)

The figure 5-5 shows results of the correlation coefficient of CMORPH and PERSIANN products versus the interpolation product, after the cumulative distribution function (CDF) matching. From the figure we can see, the average correlation coefficient of CMORPH vs. interpolation is 0.386 and the value of PERSIANN vs. interpolation is 0.221. Compared with the values before CDF matching, the average correlation coefficient of CMORPH vs. interpolation improved from 0.352 to 0.386 and the value of

PERSIANN vs. interpolation is improved from 0.185 to 0.221. The improvement of average correlation coefficient is because the CDF matching approach reduced the systematic differences between the satellite datasets and interpolation datasets.

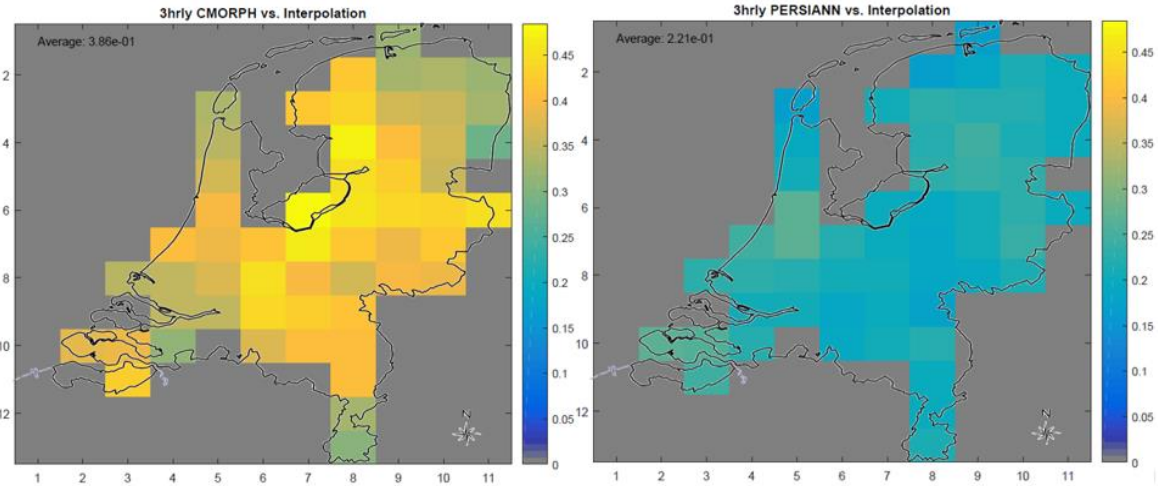

Figure 5-5 The correlation coefficient of CDF matched 3 hourly CMORPH and PERSIANN products

The figure 5-6 shows the lower root-mean-squared error (RMSE) of CDF matched CMORPH and PERSIANN versus interpolation. The average root-mean-squared error of CMORPH vs. interpolation is 3.14, while for PERSIANN vs. interpolation the value is 2.80. Both of them are lower than before CDF matching' values, which proves that the CDF matching reduced the root-mean-squared error of these two precipitation products.

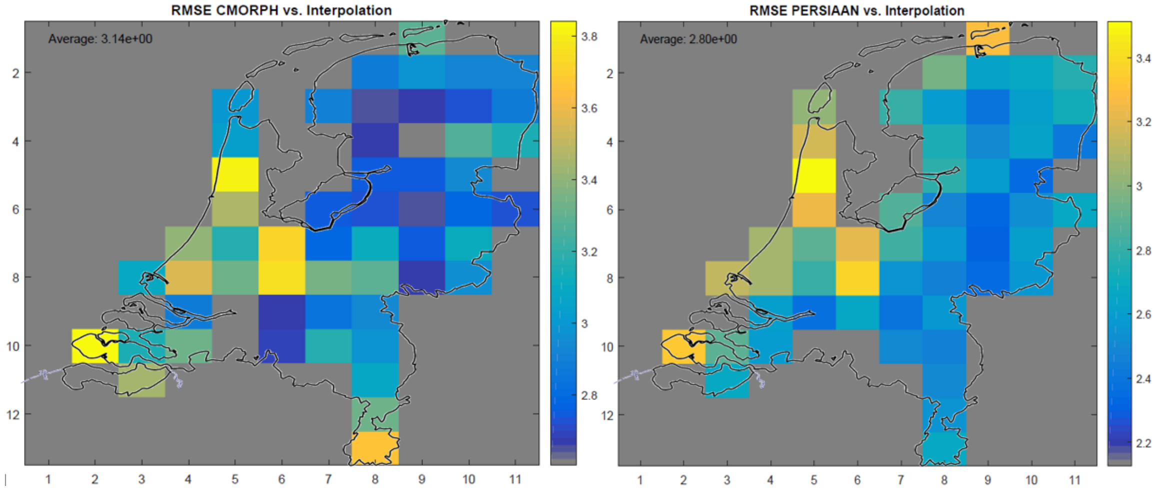

Figure 5-6 Root Mean Square Error of CDF matched 3 hourly CMORPH and PERSIANN reference to interpolation

The pixel from row 5 and column 8 was chosen from the two bias corrected products to draw the histogram, and shown in figure 5-7. According to the statistics, the frequency of precipitation amount range from 0-5mm, for CMORPH it is about 880, for PERSIANN is 910, while the frequency of precipitation amount range from 0-10mm, for CMORPH it is about 950, for PERSIANN is 930. The statistical result illustrates that the CDF Matching bias correction provided a much better correlation among these two satellite precipitation products.

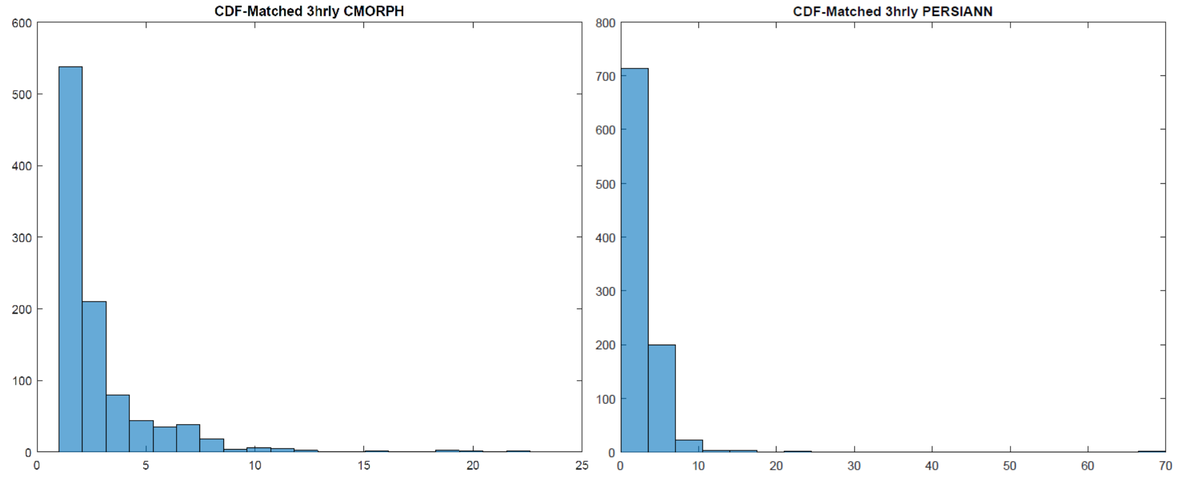

Figure 5-7 Histogram of one pixel from CDF matched 3 hourly CMORPH, PERSIANN precipitation products

The CDF plot of the CDF matched 3hourly CMMORPH, PERSIANN and interpolation products also proved the conclusion above, the two satellite products do match each other much better than before. It is to note that the CDF curve of the CDF matched satellite products were not approaching to that of the interpolation data, although themselves are much closer to each other when compared to before CDF matching. This is probably due to the fact that only the collocated precipitation data were implemented with CDF matching, while the CDF plot below was drawn considering the whole set of data. Nevertheless, we do see the improved correlation coefficient and the reduced RMSE.

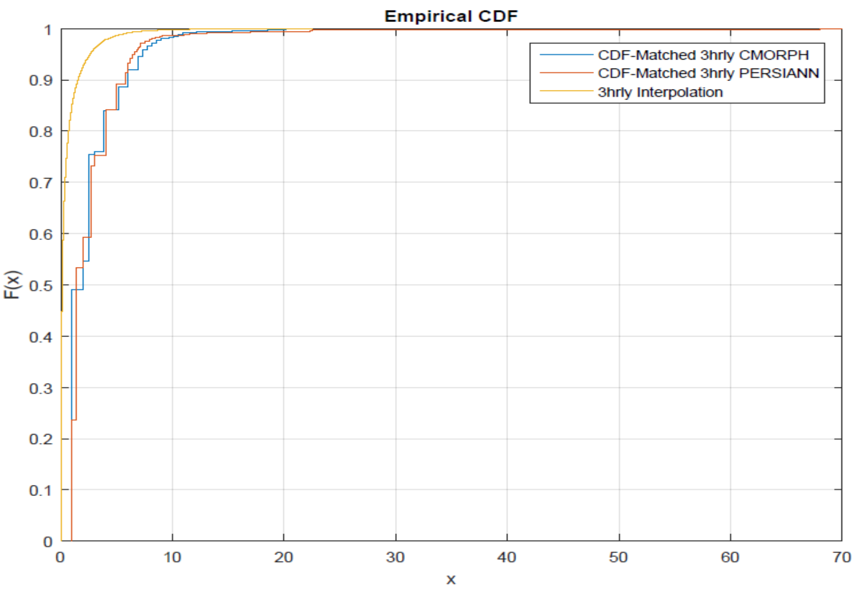

Figure 5-8 CDF plot of CDF matched 3hourly CMMORPH, PERSIANN and interpolation products

5.6 Rescale to daily data

Rescale the temporal scale of these three precipitation products to daily for further investigation. The correlation coefficient figure of each pairs is shown at the figure 5-9. It can be seen from the figure that, the average correlation coefficient of CMORPH vs. PERSIANN is 0.303, the value of CMORPH vs. interpolation is 0.476, and of PERSIANN vs. interpolation is 0.229. The results illustrate that the correlation of CMORPH versus interpolation is stronger, CMORPH versus PERSIANN is the middle one, while PERSIANN versus interpolation is weaker.

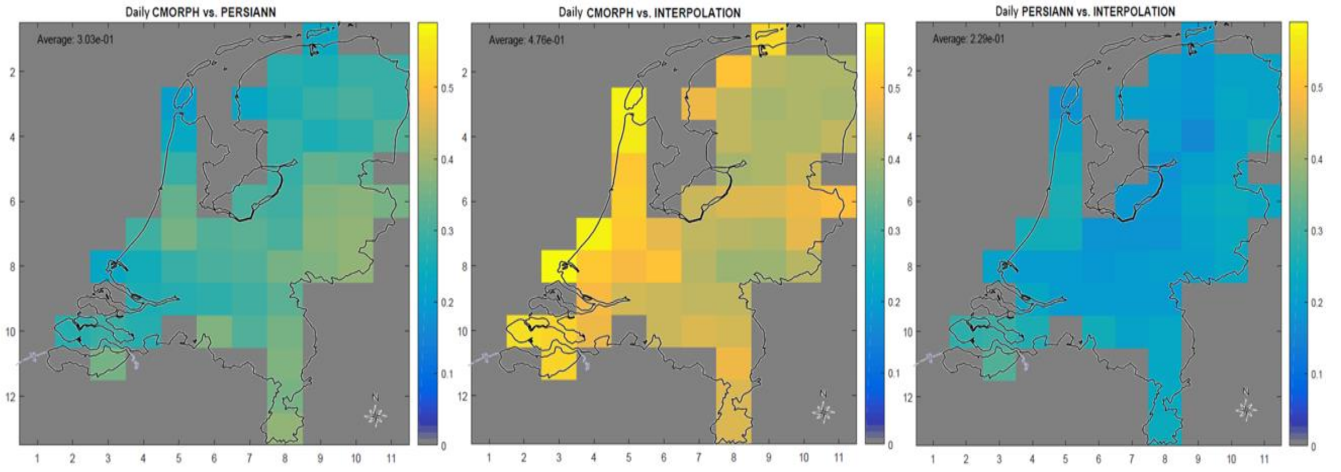

Figure 5-9 The correlation coefficient of each two precipitation products in daily temporal resolution

The Root Mean Square Error figure of daily scale precipitation products are shown on figure 5-10 below. The average root-mean-squared error of CMORPH vs. interpolation is 3.50, while for PERSIANN vs. interpolation the value is 5.99. Based on the average RMSE value, it seems the CMORPH's estimation is more reliable than PERSIANN's.

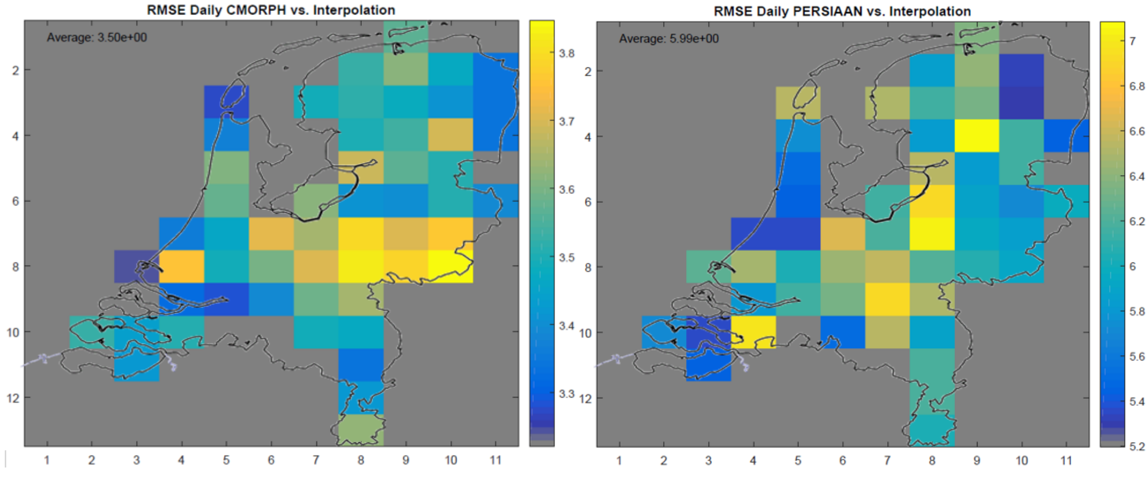

Figure 5-10 Root Mean Square Error of daily CMORPH and PERSIANN reference to interpolation

The histogram of each daily precipitation products from pixel row 5 and column 8 is drawn in the figure 5-11. For the CMORPH precipitation product, the frequency of 0 to 5 mm precipitation is close to 1000, for the PERSIANN is around 780, while interpolation is 2400. As for the frequency of 5 to 10 mm precipitation, CMORPH is 100, PERSIANN is around 130 and interpolation is 330 approximately. The result illustrates that for the precipitation less than 5mm, the two satellite precipitation product are inferior to the interpolation precipitation product because of the satellite's weak estimation to a small amount of precipitation.

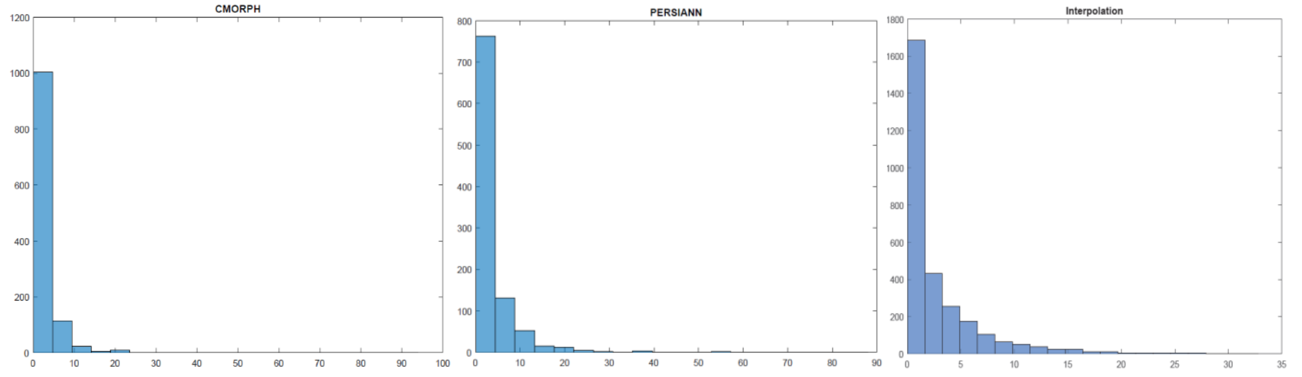

Figure 5-11 Histogram of one pixel from daily CMORPH, PERSIANN and interpolation precipitation products

The CDF plot showing on figure 5-12 shows the distribution of each three precipitation products. From the figure 5-12 we can see that the stepped line of two satellite precipitation products are very close to the interpolation products curve, which illustrates that the two satellite precipitation products distribution fit well with interpolation products.

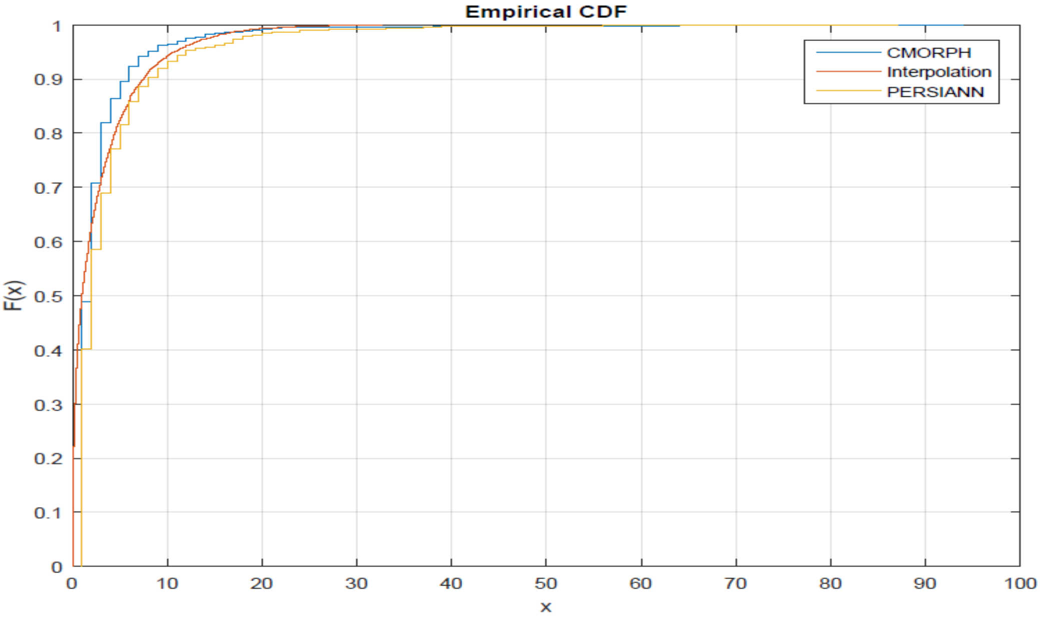

Figure 5-12 CDF plot of daily CMMORPH, PERSIANN and interpolation products

The CDF matching approach has been applied on the two pairs of precipitation products, using the interpolation product as the reference. From the figure 5-13 we can see, the average correlation coefficient of CMORPH vs. interpolation is 0.569 and the value of PERSIANN vs. interpolation is 0.419. Compared with the values before CDF matching, the average correlation coefficient of CMORPH vs. interpolation improved from 0.476 to 0.569 and the value of PERSIANN vs. interpolation is improved from 0.229 to 0.419. Obviously, the correlation coefficient between two datasets improved markedly after CDF matching reducing the bias of them.

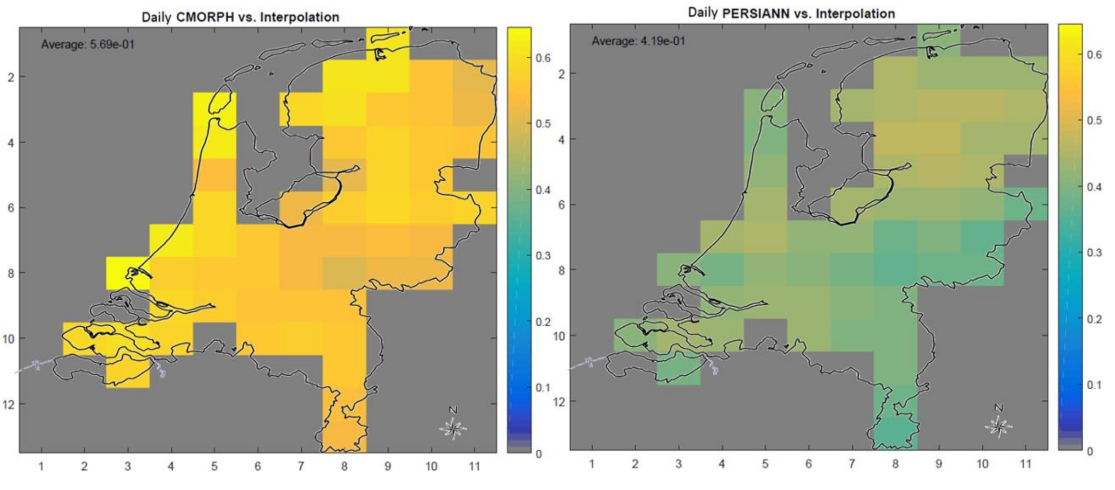

Figure 5-13 The correlation coefficient of CDF matched daily CMORPH and PERSIANN products

After CDF matching, calculate the Root Mean Square Error again to see the difference. The average root-mean-squared error of CMORPH vs. interpolation is 3.45, while for PERSIANN vs. interpolation the value is 3.65. Compared with the value before CDF matching, it is found that for CMORPH, the value shows a slight reduction only 0.05 less, but for PERSIANN, the value drops from 5.99 to 3.65. Anyhow, the CDF matching do reduce the root-mean-squared error of these precipitation products.

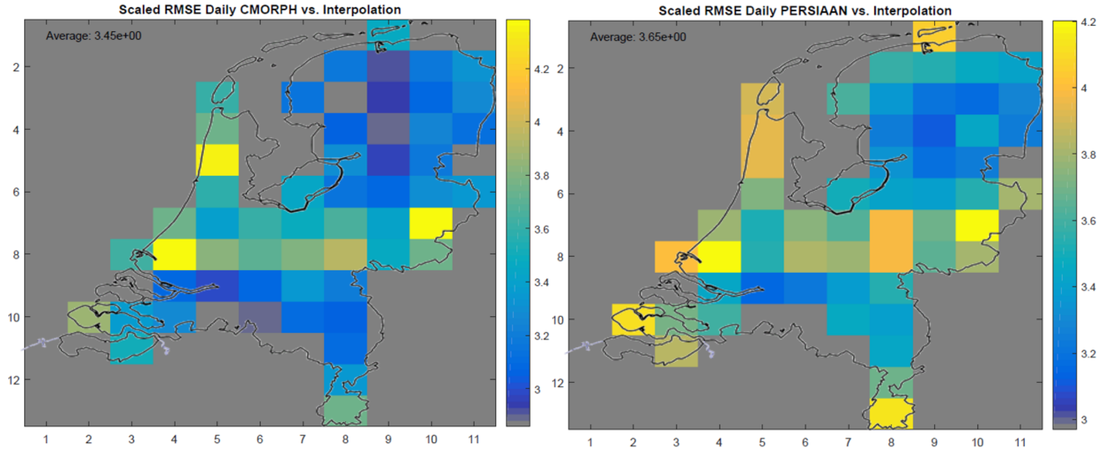

Figure 5-14 Root Mean Square Error of daily CMORPH and PERSIANN reference to interpolation

The figure 5-15 showing below indicates distribution of each CDF matched daily precipitation products from pixel row 5 and column 8. From the histogram we can get the information that for the CDF matched CMORPH precipitation product, the frequency of 5 to 10 mm precipitation is close to 530, for the PERSIANN is around 400, while the frequency of 10 to 15 mm precipitation, CMORPH is 60, PERSIANN is around 30.

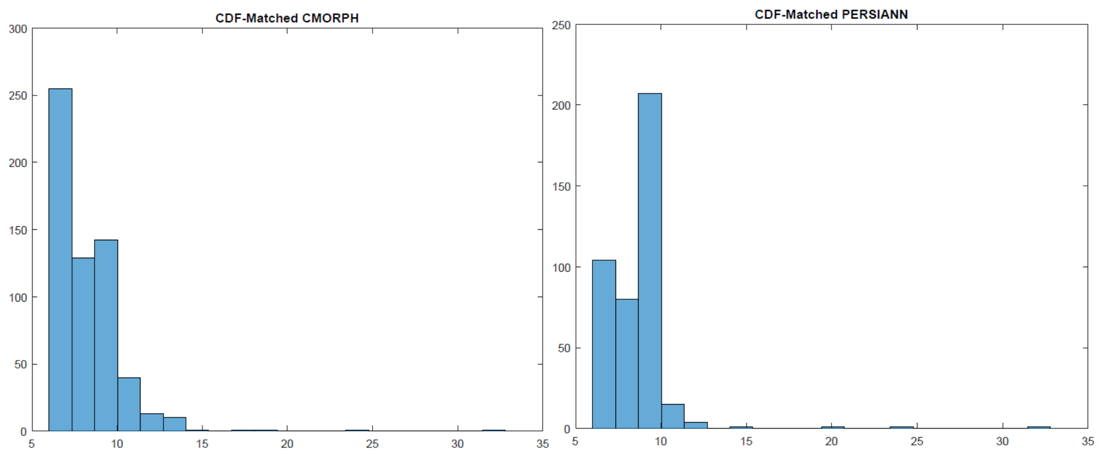

Figure 5-15 Histogram of one pixel from CDF matched daily CMORPH, PERSIANN and interpolation precipitation products

Finally, the empirical CDF plot which shows the fitness of the distribution of each three CDF matched daily precipitation products is given below (figure 5-16). As the figure shows, the two satellite products matched each other much better than before CDF matching. However, they depart from the interpolation CDF curve for the precipitation data between 0-10mm, while much closer to interpolation CDF curve for precipitation > 10mm. This is again due to the selection of only collocated data for implementing CDF matching, which also leads to the shift of satellite's estimation of low precipitation amount to slightly higher precipitation amount after the bias-correction.

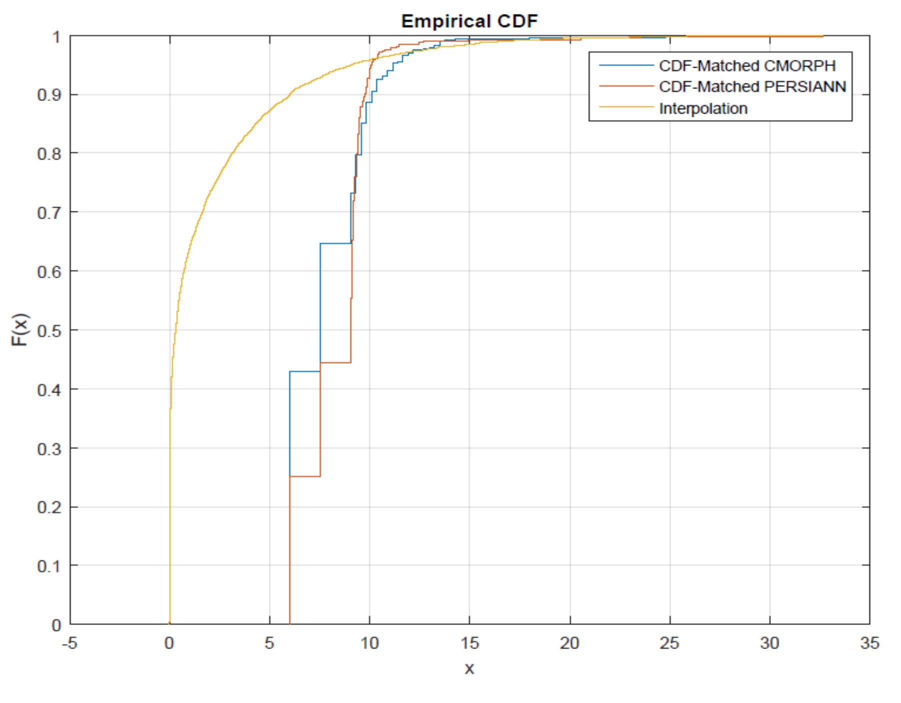

Figure 5-16 CDF plot of CDF matched daily CMMORPH, PERSIANN and interpolation products

In conclusion, CDF Matching bias correction provided a much better correlation among these three precipitation products. Although one can see the improved correlation coefficients for the daily products is obvious for CMORPH, it is interesting to see that from the RMSE results the PERSIAAN shows better improvement.

5.7 Triple collocation

In this section, we present the results of triple collocation analysis. The data used for this analysis are the CMORPH, PERSIANN and interpolation precipitation products, the time scale is 3 hourly and daily respectively.

5.7.1 Triplet number

Firstly, the 3 hourly and daily scale's triplet number were respectively shown in the figure 5-17 below. The triplet number in this paper is defined as the number of estimations who are collocated to each other among these three precipitation datasets. From the figure we can see, the daily datasets have much greater triplet numbers than 3hourly. This indicates that at high temporal resolution (e.g. 3hrly in this study) satellite data cannot accurately predict the occurrences of precipitation. It seems that the daily datasets are preferable to be processed using triple collocation technique.

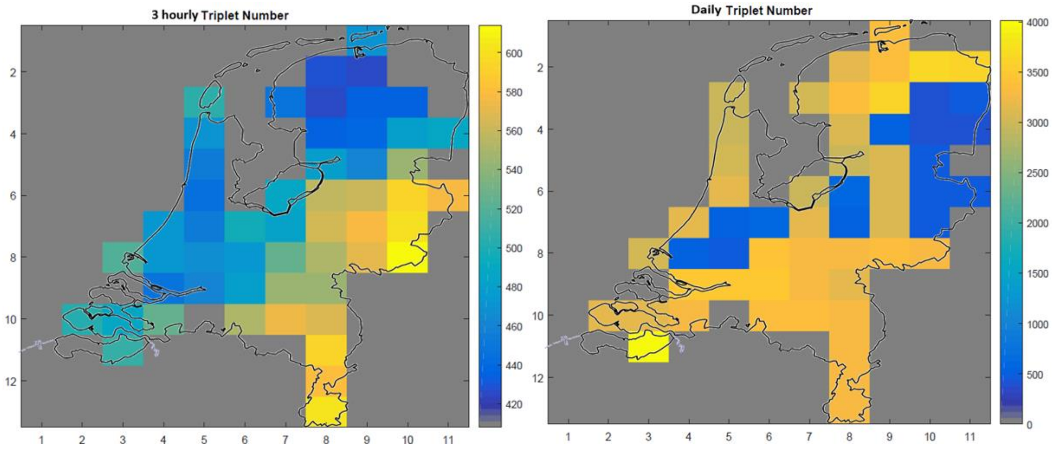

Figure 5-17 Triplet number of 3 hourly and daily scale respectively

5.7.2 P-Value

In statistics, the p-value is a function of the observed sample results (expressed as a test statistic) that is used for testing a statistical hypothesis. That means when we perform a hypothesis test in statistics, a p-value can help to weigh the strength of the evidence and determine the significance of the results. The p-value indicates how extreme is the value found for the test statistic in the distribution under the null hypothesis. The smaller the p-value, the more extreme the outcome. ("p-value - Wikipedia,") In this section, we apply the p-value to determine the significance of the correlation of these three products in 3 hourly and daily respectively. The results are showing at the figure 5-18, in order to demonstrate the result more clearly, we show the 1 minus p-value. From the figure we can get that, overall, the p value of the daily scale result is smaller than that of 3 hourly result.

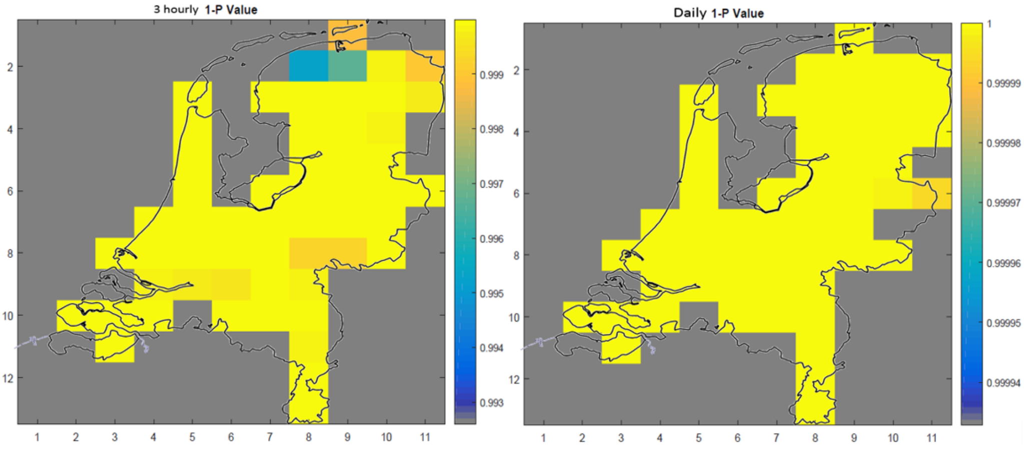

Figure 5-18 The p-value of three products in 3 hourly and daily scale

5.7.3 Min and Max correlation coefficient

Figure 5-19a and Figure 5-19b show the minimum and maxima correlation coefficient of 3 hourly and daily scale respectively. For every pixel, selecting the minimum value and maxima value from the three pairs products' correlation coefficient, and draw the result as the figure below.

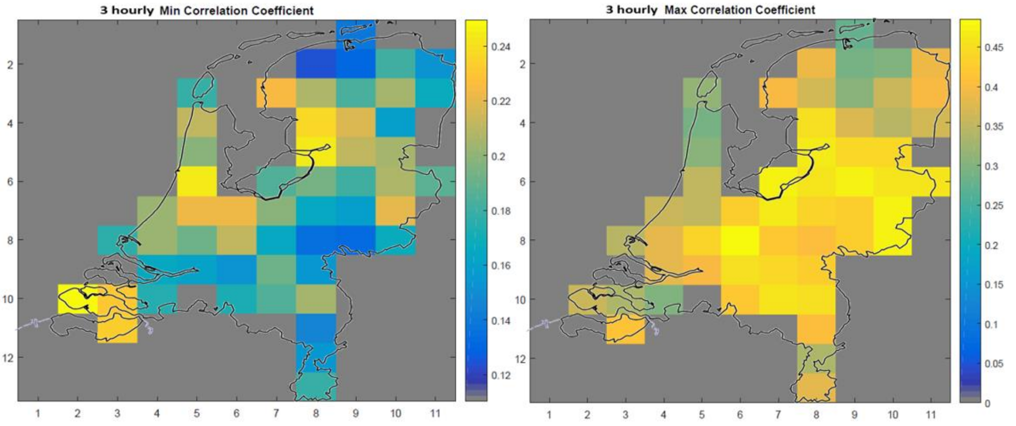

Figure 5-19a Min and Max correlation coefficient of 3 hourly scale

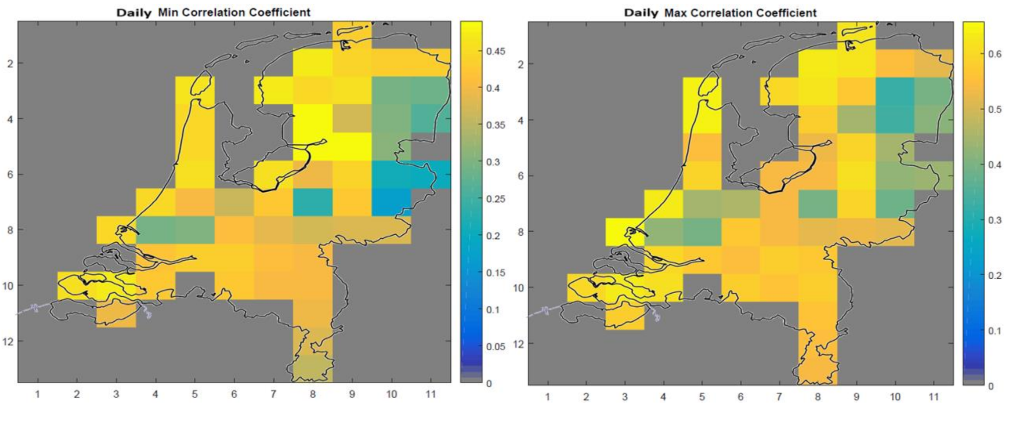

Figure 5-19b Min and Max correlation coefficient of daily scale

5.7.4 Relative errors

The results of triple collocation process is showed below as figure 5-20a and 5-20b for the 3 hourly and daily scale respectively. Compare these figures carefully, it is not difficult to find that, for 3 hourly scale, the average relative error of CMORPH is 0.58, PERSIANN is 3.64 while interpolation is 2.68. For daily scale, the average relative error of CMORPH is 1.93, PERSIANN is 5.47 while interpolation 4.31. Therefore, the conclusion can be summarized that the relative error of CMORPH is the lowest among these three products and interpolation is at the medium while PERSIANN is the highest one.

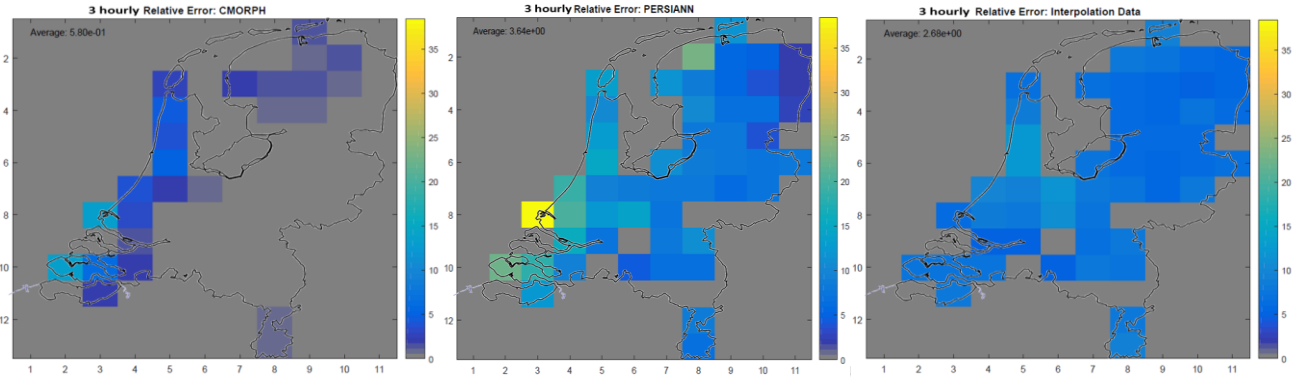

Figure 5-20a Relative errors of CMORPH, PERSIANN and interpolation products in 3hourly scale

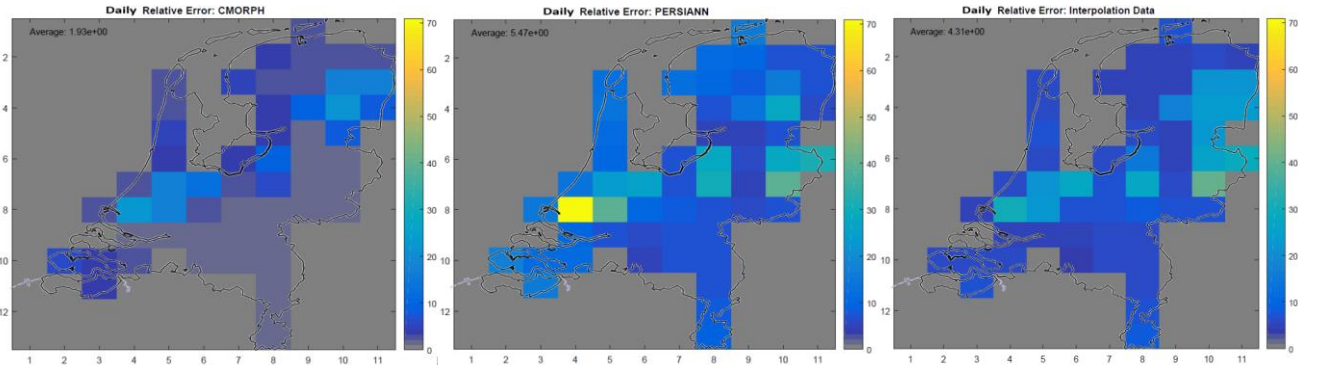

Figure 5-20b Relative errors of CMORPH, PERSIANN and interpolation products in daily scale

6. Conclusion and prospect

Based on the research above, the following conclusions can be drawn:

1. The correlation between CMORPH and interpolation rain gauge data is the strongest, two satellite precipitation products (CMORPH and PERSIANN) is medium while PERSIANN and interpolation rain gauge data is the weakest one.
2. CPMORPH product’s behavior is better than PERSIANN’s when they are correlated to the interpolation products.
3. For the low precipitation amount like 0-5mm, the two satellites provide a relatively weak retrieval.
4. Cumulative distribution functions (CDF) matching is a superior approach which can reduce the bias among several datasets, improve the correlation coefficient and reduce the RMSE among them.
5. The relative error of CMORPH is lower than the other two products’, interpolation is the medium while PERSIANN is the highest one.

   The research can be referenced to the bias correction and triple collocation of the precipitation products over the Netherlands. The results of this paper can be useful for further determination of the relative weights of these precipitation products and obtain a merged precipitation product.

Reference

Ahnert, P., Krajewski, W., & Johnson, E. (1986). Kalman Filter Estimation of Radar-Rainfall Field Bias. *Preprints, 23rd Conf. On Radar Meteor., …*. Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Kalman+Filter+Estimation+of+Radar-Rainfall+Field+Bias#0

Alemohammad, S. H., McColl, K. a., Konings, a. G., Entekhabi, D., & Stoffelen, a. (2015). Characterization of precipitation product errors across the US using multiplicative Triple Collocation. *Hydrology and Earth System Sciences Discussions*, *12*(2), 2527–2559. http://doi.org/10.5194/hessd-12-2527-2015

AMS Glossary. (n.d.). Retrieved August 27, 2015, from http://glossary.ametsoc.org/wiki/Main_Page

Anagnostou, E. N., Krajewski, W. F., Seo, D.-J., & Johnson, E. R. (1998). Mean-Field Rainfall Bias Studies for WSR-88D. *Journal of Hydrologic Engineering*, *3*(3), 149–159. http://doi.org/10.1061/(ASCE)1084-0699(1998)3:3(149)

Asefa, T., & Adams, A. (2013). Reducing bias-corrected precipitation projection uncertainties: A Bayesian-based indicator-weighting approach. *Regional Environmental Change*, *13*(SUPPL.1), 111–120. http://doi.org/10.1007/s10113-013-0431-9

Attema, J., & Lenderink, G. (2011). Mean precipitation changes and coastal effects in the ENSEMBLES regional climate model simulations. *KNMI Scientific Report 2011-03*, 42 pp.

Beek, R. van de. (2013, June 10). *Rainfall variability in the Netherlands from radars, rain gauges, and disdrometers*. s.n.]. Retrieved from http://edepot.wur.nl/259986

Bohling, G. (2005). Kriging. *Kansas Geological Survey*, (October), 1–20. Retrieved from http://people.ku.edu/~gbohling/cpe940/Kriging.pdf

Brauer, C. C., Teuling, a. J., Overeem, a., Van Der Velde, Y., Hazenberg, P., Warmerdam, P. M. M., & Uijlenhoet, R. (2011). Anatomy of extraordinary rainfall and flash flood in a Dutch lowland catchment. *Hydrology and Earth System Sciences*, *15*(6), 1991–2055. http://doi.org/10.5194/hess-15-1991-2011

Caires, S. (2003). Validation of ocean wind and wave data using triple collocation. *Journal of Geophysical Research*, *108*(C3), 3098. http://doi.org/10.1029/2002JC001491

Cimini, D., Pierdicca, N., Pichelli, E., Ferretti, R., Mattioli, V., Bonafoni, S., … Perissin, D. (2012). On the accuracy of integrated water vapor observations and the potential for mitigating electromagnetic path delay error in InSAR. *Atmospheric Measurement Techniques*, *5*(5), 1015–1030. http://doi.org/10.5194/amt-5-1015-2012

Correlation coefficient - Wikipedia, the free encyclopedia. (n.d.). Retrieved February 12, 2016, from https://en.wikipedia.org/wiki/Correlation_coefficient

Cumulative Distribution Functions (CDFs) in detail. (n.d.). Retrieved from http://ukclimateprojections.metoffice.gov.uk/22618

Duan, J. F., Selker, J., & Grant, G. E. (1998). Evaluation of probability density functions in precipitation models for the Pacific Northwest. *Journal of the American Water Resources*

Association, *34*(3), 617–627. http://doi.org/10.1111/j.1752-1688.1998.tb00959.x

Ebert, E. ., & McBride, J. . (2000). Verification of precipitation in weather systems: determination of systematic errors. *Journal of Hydrology*, *239*(1-4), 179–202. http://doi.org/10.1016/S0022-1694(00)00343-7

Franses, P. H., & Koning, A. J. (2003). Did the incidence of high precipitation levels increase? Statistical evidence for the Netherlands.

Goovaerts, P. (2000). Geostatistical approaches for incorporating elevation into the spatial interpolation of rainfall. *Journal of Hydrology*, *228*(1-2), 113–129. http://doi.org/10.1016/S0022-1694(00)00144-X

Hofstra, N., Haylock, M., New, M., Jones, P., & Frei, C. (2008). Comparison of six methods for the interpolation of daily, European climate data. *Journal of Geophysical Research*, *113*(D21), D21110. http://doi.org/10.1029/2008JD010100

Joyce, R. J., Janowiak, J. E., Arkin, P. a., & Xie, P. (2004). CMORPH: A Method that Produces Global Precipitation Estimates from Passive Microwave and Infrared Data at High Spatial and Temporal Resolution. *Journal of Hydrometeorology*, *5*(3), 487–503. http://doi.org/10.1175/1525-7541(2004)005<0487:CAMTPG>2.0.CO;2

Kao, C. Y. S., & Hung, F. L. P. (2004). Twelve Different Interpolation Methods: a Case Study. *Proceedings of the XXth ISPRS Congress*, *35*, 778–785. Retrieved from http://www.isprs.org/proceedings/XXXV/congress/comm2/papers/231.pdf

Leander, R., & Buishand, T. A. (2007). Resampling of regional climate model output for the simulation of extreme river flows. *Journal of Hydrology*, *332*(3-4), 487–496. http://doi.org/10.1016/j.jhydrol.2006.08.006

Levizzani, V., Amorati, R., & Meneguzzo, F. (2002). A Review of Satellite-based Rainfall Estimation Methods. Retrieved from http://www.researchgate.net/publication/252272255_A_Review_of_Satellite-based_Rainfall_Estimation_Methods

Levizzani, V., Bauer, P., & Turk, F. J. (2007). *Measuring Precipitation from Space: EURAINSAT and the Future*. Springer Science & Business Media. Retrieved from https://books.google.com/books?id=uks14jNh8bwC&pgis=1

Luo, Y., Zhu, Y., & Springs, C. (2011). • Develop and enhance bias-correction and.

McCollum, J. R., Krajewski, W. F., Ferraro, R. R., & Ba, M. B. (2002). Evaluation of Biases of Satellite Rainfall Estimation Algorithms over the Continental United States. *Journal of Applied Meteorology*, *41*, 1065–1080. http://doi.org/10.1175/1520-0450(2002)041<1065:EOBOSR>2.0.CO;2

Miralles, D. G., Crow, W. T., & Cosh, M. H. (2010). Estimating Spatial Sampling Errors in Coarse-Scale Soil Moisture Estimates Derived from Point-Scale Observations. *Journal of Hydrometeorology*, *11*(6), 1423–1429. http://doi.org/10.1175/2010JHM1285.1

Narapusetty, B., Stan, C., & Kumar, A. (2012). Applicability of simple bias correction methods in improving decadal forecast skill, (March 2012).

Olsson, T., Jakkila, J., Veijalainen, N., Backman, L., Kaurola, J., & Vehvil??inen, B. (2015).

Impacts of climate change on temperature, precipitation and hydrology in Finland - Studies using bias corrected Regional Climate Model data. *Hydrology and Earth System Sciences*, *19*(7), 3217–3238. http://doi.org/10.5194/hess-19-3217-2015

p-value - Wikipedia. (n.d.). Retrieved February 15, 2016, from https://nl.wikipedia.org/wiki/P-waarde

Ratheesh, S., Mankad, B., Basu, S., Kumar, R., & Sharma, R. (2013). Assessment of satellite-derived sea surface salinity in the Indian Ocean. *IEEE Geoscience and Remote Sensing Letters*, *10*(3), 428–431. http://doi.org/10.1109/LGRS.2012.2207943

Roebeling, R. a., Wolters, E. L. a., Meirink, J. F., & Leijnse, H. (2012). Triple Collocation of Summer Precipitation Retrievals from SEVIRI over Europe with Gridded Rain Gauge and Weather Radar Data. *Journal of Hydrometeorology*, *13*(5), 1552–1566. http://doi.org/10.1175/JHM-D-11-089.1

Satellite Precipitation - CHRS. (n.d.). Retrieved August 28, 2015, from http://chrs.web.uci.edu/research/satellite_precipitation/activities00.html

Scipal, K., Dorigo, W., & De Jeu, R. (2010). Triple collocation - A new tool to determine the error structure of global soil moisture products. *International Geoscience and Remote Sensing Symposium (IGARSS)*, *3*(2), 4426–4429. http://doi.org/10.1109/IGARSS.2010.5652128

Scipal, K., Holmes, T., de Jeu, R., Naeimi, V., & Wagner, W. (2008). A possible solution for the problem of estimating the error structure of global soil moisture data sets. *Geophysical Research Letters*, *35*(24), L24403. http://doi.org/10.1029/2008GL035599

Seo, D.-J., & Breidenbach, J. P. (2002). Real-Time Correction of Spatially Nonuniform Bias in Radar Rainfall Data Using Rain Gauge Measurements. *Journal of Hydrometeorology*, *3*(2), 93–111. http://doi.org/10.1175/1525-7541(2002)003<0093:RTCOSN>2.0.CO;2

Seyyedi, H. (2010). Comparing satellite derived rainfall with ground based radar for North-western Europe, (August), 96.

Smith, J. A., & Krajewski, W. F. (1991). Estimation of the Mean Field Bias of Radar Rainfall Estimates. *Journal of Applied Meteorology*, *30*(4), 397–412. http://doi.org/10.1175/1520-0450(1991)030<0397:EOTMFB>2.0.CO;2

Smith, T. M., Arkin, P. A., Bates, J. J., & Huffman, G. J. (2006). Estimating Bias of Satellite-Based Precipitation Estimates. *Journal of Hydrometeorology*, *7*(5), 841–856. http://doi.org/10.1175/JHM524.1

Stoffelen, A. (1998). Toward the true near-surface wind speed: Error modeling and calibration using triple collocation. *Journal of Geophysical Research*, *103*(C4), 7755. http://doi.org/10.1029/97JC03180

Tapiador, F. J., Turk, F. J., Petersen, W., Hou, A. Y., García-Ortega, E., Machado, L. A. T., … de Castro, M. (2012). Global precipitation measurement: Methods, datasets and applications. *Atmospheric Research*, *104-105*, 70–97. http://doi.org/10.1016/j.atmosres.2011.10.021

Terink, W., Hurkmans, R. T. W. L., Torfs, P. J. J. F., & Uijlenhoet, R. (2009). Bias correction of temperature and precipitation data for regional climate model application to the Rhine basin. *Hydrol. Earth Syst. Sci. Discuss.*, *6*(4), 5377–5413. http://doi.org/10.5194/hessd-6-5377-2009

Tesfagiorgis, K., Mahani, S. E., Krakauer, N. Y., & Khanbilvardi, R. (2011). Bias correction of satellite rainfall estimates using a radar-gauge product – a case study in Oklahoma (USA). *Hydrology and Earth System Sciences*. http://doi.org/10.5194/hess-15-2631-2011

Thao, S., & Eymard, L. (2014). Trend and variability of the atmospheric water vapor: a mean sea level issue. *Journal of Atmospheric …*. Retrieved from http://journals.ametsoc.org/doi/abs/10.1175/JTECH-D-13-00157.1

US Department of Commerce, N. E. S. R. L. (n.d.). ESRL : PSD : Precipitation Forecast Products Based on NCEP GEFS Reforecasts, Version2 . Retrieved from http://www.esrl.noaa.gov/psd/forecasts/reforecast2/analogs/index2.html

Using the cumulative distribution function (CDF). (n.d.). Retrieved from http://support.minitab.com/en-us/minitab/17/topic-library/basic-statistics-and-graphs/probability-distributions-and-random-data/basics/using-the-cumulative-distribution-function-cdf/

van Dijk, A. I. J. M., Renzullo, L. J., Wada, Y., & Tregoning, P. (2014). A global water cycle reanalysis (2003–2012) merging satellite gravimetry and altimetry observations with a hydrological multi-model ensemble. *Hydrology and Earth System Sciences*, *18*(8), 2955–2973. http://doi.org/10.5194/hess-18-2955-2014

APPENDIX

1. Syntax of batch file 1 to import the raw CMORPH precipitation data of 200301.

J:\batchfiles\multiread_RAW_CMORPH_025D3HR_start_v1.bat 2016年2月6日 17:14

```
rem import and pre-process CMORPH 025deg-3hly rainfall data in Ilwis
rem CMORPH raw binary data filename is tricky with timestamp in extention
rem please also read CMORPH data info on file formats, contents (see data website)
rem datafile used here is flat binary (float32), no header, containing 8*3hly global data at
0.25deg
rem open ilwis first in work directory and close again (to register internally)
rem this batch file should be in your work directory
rem
rem copy original file(s) from any storage (here D:\..) to work directory
copy  D:\test\cmorph\RAW\CMORPH_V1.0_RAW_0.25deg-3HLY_200301??
rem rename files and copy timestamp from file.extension to filename
ren *.0_RAW_0.25deg-3HLY_20030101 RAW_CMORPH_20030101
ren *.0_RAW_0.25deg-3HLY_20030102 RAW_CMORPH_20030102
ren *.0_RAW_0.25deg-3HLY_20030103 RAW_CMORPH_20030103
ren *.0_RAW_0.25deg-3HLY_20030104 RAW_CMORPH_20030104
ren *.0_RAW_0.25deg-3HLY_20030105 RAW_CMORPH_20030105
ren *.0_RAW_0.25deg-3HLY_20030106 RAW_CMORPH_20030106
ren *.0_RAW_0.25deg-3HLY_20030107 RAW_CMORPH_20030107
ren *.0_RAW_0.25deg-3HLY_20030108 RAW_CMORPH_20030108
ren *.0_RAW_0.25deg-3HLY_20030109 RAW_CMORPH_20030109
ren *.0_RAW_0.25deg-3HLY_20030110 RAW_CMORPH_20030110
ren *.0_RAW_0.25deg-3HLY_20030111 RAW_CMORPH_20030111
ren *.0_RAW_0.25deg-3HLY_20030112 RAW_CMORPH_20030112
ren *.0_RAW_0.25deg-3HLY_20030113 RAW_CMORPH_20030113
ren *.0_RAW_0.25deg-3HLY_20030114 RAW_CMORPH_20030114
ren *.0_RAW_0.25deg-3HLY_20030115 RAW_CMORPH_20030115
ren *.0_RAW_0.25deg-3HLY_20030116 RAW_CMORPH_20030116
ren *.0_RAW_0.25deg-3HLY_20030117 RAW_CMORPH_20030117
ren *.0_RAW_0.25deg-3HLY_20030118 RAW_CMORPH_20030118
ren *.0_RAW_0.25deg-3HLY_20030119 RAW_CMORPH_20030119
ren *.0_RAW_0.25deg-3HLY_20030120 RAW_CMORPH_20030120
ren *.0_RAW_0.25deg-3HLY_20030121 RAW_CMORPH_20030121
ren *.0_RAW_0.25deg-3HLY_20030122 RAW_CMORPH_20030122
ren *.0_RAW_0.25deg-3HLY_20030123 RAW_CMORPH_20030123
ren *.0_RAW_0.25deg-3HLY_20030124 RAW_CMORPH_20030124
ren *.0_RAW_0.25deg-3HLY_20030125 RAW_CMORPH_20030125
ren *.0_RAW_0.25deg-3HLY_20030126 RAW_CMORPH_20030126
ren *.0_RAW_0.25deg-3HLY_20030127 RAW_CMORPH_20030127
ren *.0_RAW_0.25deg-3HLY_20030128 RAW_CMORPH_20030128
ren *.0_RAW_0.25deg-3HLY_20030129 RAW_CMORPH_20030129
ren *.0_RAW_0.25deg-3HLY_20030130 RAW_CMORPH_20030130
ren *.0_RAW_0.25deg-3HLY_20030131 RAW_CMORPH_20030131
rem
rem unzip not needed here so I add rem to these 2 unzip command lines
rem set PATH=%D:\Programs\7z%\
rem LFNFOR On
rem for %%j in (*.0_RAW_0.25deg-3HLY_200301??) do ren %%j RAW_CMORPH_200301??.x
rem execute the next preprocessor batch file sequentally for all file timestamps
rem passes also parameter (longfilename)
for %%j in (RAW_CMORPH_*) do multiread_RAW_CMORPH_025D3HR_preproc_v1.bat %%j
```

-1-

2. Syntax of batch file 2 to read, mirror rotate and recompose the maplist.

E:\batchfiles\multiread_RAW_CMORPH_025D3HR_preproc_v1.bat 2016年2月6日 16:12

```
@echo off
echo rem: CMORPH 025D 3HR time series dataset import to ilwis
rem source data from CMORPH archive
rem note different filenaming from
echo rem: sample file name =
rem pick timestamp from long filename (passed parameter start batchfile )
set longfilename=%1
set shortfilename1=%longfilename:~11,8%
rem watch your program paths this batch code needs
set PATH=%G:\Graduation\DATA\Ilwis372%
rem
rem import as maplist

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%o.mpl:=maplist('RAW_CMORPH_%shortfilename1%',
genras,Convert,1440,8,0,BSQ,Real,4,NoSwap,CreateMpr)

rem rotate input map list

call Ilwis.exe -C
cmorph025d3hr_%shortfilename1%.mpl:=MapListApplic('cmorph025d3hr_%shortfilename1%o',MapMirrorR
otate(##,MirrHor))

rem create subset maps

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%1_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_1',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%1_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_1',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%2_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_2',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%2_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_2',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%3_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_3',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%3_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_3',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%4_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_4',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%4_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_4',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%5_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_5',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%5_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_5',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%6_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_6',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%6_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_6',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%7_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_7',1,1,480
,720)
call ilwis.exe -C
```

-1-

E:\batchfiles\multiread_RAW_CMORPH_025D3HR_preproc_v1.bat 2016年2月6日 16:12

```
cmorph025d3hr_%shortfilename1%7_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_7',1,721,4
80,720)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%8_east.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_8',1,1,480
,720)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%8_west.mpr:=MapSubMap('cmorph025d3hr_%shortfilename1%_8',1,721,4
80,720)

rem add submap georef
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%1_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%1_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%2_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%2_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%3_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%3_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%4_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%4_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%5_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%5_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%6_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%6_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%7_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%7_west.mpr cmorph025d_west

call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%8_east.mpr cmorph025d_east
call ilwis.exe -C setgrf cmorph025d3hr_%shortfilename1%8_west.mpr cmorph025d_west

rem glue maps using georef global

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%1_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%1_west,cmorph025d3hr_%shortfilename1%1_east,replace)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%2_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%2_west,cmorph025d3hr_%shortfilename1%2_east,replace)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%3_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%3_west,cmorph025d3hr_%shortfilename1%3_east,replace)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%4_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%4_west,cmorph025d3hr_%shortfilename1%4_east,replace)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%5_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%5_west,cmorph025d3hr_%shortfilename1%5_east,replace)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%6_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%6_west,cmorph025d3hr_%shortfilename1%6_east,replace)

call ilwis.exe -C
cmorph025d3hr_%shortfilename1%7_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%7_west,cmorph025d3hr_%shortfilename1%7_east,replace)
call ilwis.exe -C
cmorph025d3hr_%shortfilename1%8_recomposed.mpr:=MapGlue(cmorph_025d.grf,cmorph025d3hr_%shortfi
lename1%8_west,cmorph025d3hr_%shortfilename1%8_east,replace)

rem further processing using Ilwis Script (see separate script)

rem delete intermediate files and obsolete objects
del CMORPH_V?.?_RAW_0.25deg-3HLY_*.*
del cmorph025d3hr_%shortfilename1%o?.*
del cmorph025d3hr_%shortfilename1%_?.*
```

-2-

E:\batchfiles\multiread_RAW_CMORPH_025D3HR_preproc_v1.bat 2016年2月6日 16:12

```
del cmorph025d3hr_%shortfilename1%.mpl
del cmorph025d3hr_%shortfilename1%?_east.*
del cmorph025d3hr_%shortfilename1%?_west.*

:END
call ilwis.exe -C closeall
```

-3-

3. R script code of ordinary kriging interpolation for the month of 200301.

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
######################################################################################
setwd("D:/Interpolation/kriging")
library(rgdal)
library(maptools)
library(gstat)
library(MASS)
library(Momocs)
library(fields)
library(ggplot2)
library(car)
library(sp)
######################################################################################
# locaton of stations:
NL32station_XY <- read.csv("./input/NL32station_XY.csv")
str(NL32station_XY)
# fix the coordinates:
NL32station_XY$X <- NL32station_XY$X..km. * 1000
NL32station_XY$Y <- NL32station_XY$Y..km. * 1000
# fix the station names:
NL32station_XY$Locatie <- as.factor(gsub(pattern='[[:space:]]+$', replacement='',
NL32station_XY$Locatie))
######################################################################################
# import the precipitation values:
tmp <- readLines(paste(getwd(), "/", "./input/Insitu_200301.txt", sep="/"))
tmp <- gsub(pattern=' . ', replacement=' 0.0 ', tmp, fixed = TRUE)
tmp <- gsub(pattern='[[:space:]]{2,7}', replacement=',', tmp)
tmp[8]
write.table(t(matrix(unlist(strsplit(tmp, ",")[-c(1:8)]), nrow=34)), "tmp.txt",
col.names=FALSE, row.names=FALSE)
Insitu_200301 <- read.table("tmp.txt", col.names=c("empty1", "NR", paste("D", 1:31, sep=""),
"Locatie"))
Insitu_200301$empty1 <- NULL
Insitu_200301$Locatie <- as.factor(gsub(pattern='[[:space:]]+$', replacement='',
Insitu_200301$Locatie))
str(Insitu_200301)
######################################################################################
# merge the precipitation and coordinates:
Insitu_200301.XY <- merge(x=NL32station_XY[,c("X", "Y", "Locatie")], y=Insitu_200301,
by=c("Locatie"), all.x=TRUE)
str(Insitu_200301.XY)
coordinates(Insitu_200301.XY) <- ~X+Y
######################################################################################
# import gridded map:
grids25km <- readGDAL("./input/prec25km.asc")
names(grids25km) <- "prec25km"
grids25km$nlmask25km <- as.factor(readGDAL("./input/nlmask25km.asc")$band1)
######################################################################################
######################################################################################
# percipitation values are commonly skewed:
D1.sel <- !is.na(Insitu_200301.XY$D1)
D1.bubble=bubble(Insitu_200301.XY[D1.sel, "D1"])
D2.sel <- !is.na(Insitu_200301.XY$D2)
D2.bubble=bubble(Insitu_200301.XY[D2.sel, "D2"])
D3.sel <- !is.na(Insitu_200301.XY$D3)
D3.bubble=bubble(Insitu_200301.XY[D3.sel, "D3"])
D4.sel <- !is.na(Insitu_200301.XY$D4)
D4.bubble=bubble(Insitu_200301.XY[D4.sel, "D4"])
D5.sel <- !is.na(Insitu_200301.XY$D5)
D5.bubble=bubble(Insitu_200301.XY[D5.sel, "D5"])
D6.sel <- !is.na(Insitu_200301.XY$D6)
D6.bubble=bubble(Insitu_200301.XY[D6.sel, "D6"])
D7.sel <- !is.na(Insitu_200301.XY$D7)
D7.bubble=bubble(Insitu_200301.XY[D7.sel, "D7"])
D8.sel <- !is.na(Insitu_200301.XY$D8)
D8.bubble=bubble(Insitu_200301.XY[D8.sel, "D8"])
D9.sel <- !is.na(Insitu_200301.XY$D9)
D9.bubble=bubble(Insitu_200301.XY[D9.sel, "D9"])
D10.sel <- !is.na(Insitu_200301.XY$D10)
D10.bubble=bubble(Insitu_200301.XY[D10.sel, "D10"])
D11.sel <- !is.na(Insitu_200301.XY$D11)
D11.bubble=bubble(Insitu_200301.XY[D11.sel, "D11"])
```

-1-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
D12.sel <- !is.na(Insitu_200301.XY$D12)
D12.bubble=bubble(Insitu_200301.XY[D12.sel, "D12"])
D13.sel <- !is.na(Insitu_200301.XY$D13)
D13.bubble=bubble(Insitu_200301.XY[D13.sel, "D13"])
D14.sel <- !is.na(Insitu_200301.XY$D14)
D14.bubble=bubble(Insitu_200301.XY[D14.sel, "D14"])
D15.sel <- !is.na(Insitu_200301.XY$D15)
D15.bubble=bubble(Insitu_200301.XY[D15.sel, "D15"])
D16.sel <- !is.na(Insitu_200301.XY$D16)
D16.bubble=bubble(Insitu_200301.XY[D16.sel, "D16"])
D17.sel <- !is.na(Insitu_200301.XY$D17)
D17.bubble=bubble(Insitu_200301.XY[D17.sel, "D17"])
D18.sel <- !is.na(Insitu_200301.XY$D18)
D18.bubble=bubble(Insitu_200301.XY[D18.sel, "D18"])
D19.sel <- !is.na(Insitu_200301.XY$D19)
D19.bubble=bubble(Insitu_200301.XY[D19.sel, "D19"])
D20.sel <- !is.na(Insitu_200301.XY$D20)
D20.bubble=bubble(Insitu_200301.XY[D20.sel, "D20"])
D21.sel <- !is.na(Insitu_200301.XY$D21)
D21.bubble=bubble(Insitu_200301.XY[D21.sel, "D21"])
D22.sel <- !is.na(Insitu_200301.XY$D22)
D22.bubble=bubble(Insitu_200301.XY[D22.sel, "D22"])
D23.sel <- !is.na(Insitu_200301.XY$D23)
D23.bubble=bubble(Insitu_200301.XY[D23.sel, "D23"])
D24.sel <- !is.na(Insitu_200301.XY$D24)
D24.bubble=bubble(Insitu_200301.XY[D24.sel, "D24"])
D25.sel <- !is.na(Insitu_200301.XY$D25)
D25.bubble=bubble(Insitu_200301.XY[D25.sel, "D25"])
D26.sel <- !is.na(Insitu_200301.XY$D26)
D26.bubble=bubble(Insitu_200301.XY[D26.sel, "D26"])
D27.sel <- !is.na(Insitu_200301.XY$D27)
D27.bubble=bubble(Insitu_200301.XY[D27.sel, "D27"])
D28.sel <- !is.na(Insitu_200301.XY$D28)
D28.bubble=bubble(Insitu_200301.XY[D28.sel, "D28"])
D29.sel <- !is.na(Insitu_200301.XY$D29)
D29.bubble=bubble(Insitu_200301.XY[D29.sel, "D29"])
D30.sel <- !is.na(Insitu_200301.XY$D30)
D30.bubble=bubble(Insitu_200301.XY[D30.sel, "D30"])
D31.sel <- !is.na(Insitu_200301.XY$D31)
D31.bubble=bubble(Insitu_200301.XY[D31.sel, "D31"])
##################################################################################### hist
D1.hist=hist(log1p(Insitu_200301.XY$D1))
D2.hist=hist(log1p(Insitu_200301.XY$D2))
D3.hist=hist(log1p(Insitu_200301.XY$D3))
D4.hist=hist(log1p(Insitu_200301.XY$D4))
D5.hist=hist(log1p(Insitu_200301.XY$D5))
D6.hist=hist(log1p(Insitu_200301.XY$D6))
D7.hist=hist(log1p(Insitu_200301.XY$D7))
D8.hist=hist(log1p(Insitu_200301.XY$D8))
D9.hist=hist(log1p(Insitu_200301.XY$D9))
D10.hist=hist(log1p(Insitu_200301.XY$D10))
D11.hist=hist(log1p(Insitu_200301.XY$D11))
D12.hist=hist(log1p(Insitu_200301.XY$D12))
D13.hist=hist(log1p(Insitu_200301.XY$D13))
D14.hist=hist(log1p(Insitu_200301.XY$D14))
D15.hist=hist(log1p(Insitu_200301.XY$D15))
D16.hist=hist(log1p(Insitu_200301.XY$D16))
D17.hist=hist(log1p(Insitu_200301.XY$D17))
D18.hist=hist(log1p(Insitu_200301.XY$D18))
D19.hist=hist(log1p(Insitu_200301.XY$D19))
D20.hist=hist(log1p(Insitu_200301.XY$D20))
D21.hist=hist(log1p(Insitu_200301.XY$D21))
D22.hist=hist(log1p(Insitu_200301.XY$D22))
D23.hist=hist(log1p(Insitu_200301.XY$D23))
D24.hist=hist(log1p(Insitu_200301.XY$D24))
D25.hist=hist(log1p(Insitu_200301.XY$D25))
D26.hist=hist(log1p(Insitu_200301.XY$D26))
D27.hist=hist(log1p(Insitu_200301.XY$D27))
D28.hist=hist(log1p(Insitu_200301.XY$D28))
D29.hist=hist(log1p(Insitu_200301.XY$D29))
D30.hist=hist(log1p(Insitu_200301.XY$D30))
```

-2-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
D31.hist=hist(log1p(Insitu_200301.XY$D31))
##################################################################################### var
D1.svar <- variogram(log1p(D1)~1, Insitu_200301.XY[D1.sel,])
D1.vgm <- fit.variogram(D1.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D1),
na.rm=TRUE)/3, model="Lin"))
D1.plot=plot(D1.svar, D1.vgm)
D2.svar <- variogram(log1p(D2)~1, Insitu_200301.XY[D2.sel,])
D2.vgm <- fit.variogram(D2.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D2),
na.rm=TRUE)/3, model="Lin"))
D2.plot=plot(D2.svar, D2.vgm)
D3.svar <- variogram(log1p(D3)~1, Insitu_200301.XY[D3.sel,])
D3.vgm <- fit.variogram(D3.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D3),
na.rm=TRUE)/3, model="Lin"))
D3.plot=plot(D3.svar, D3.vgm)
D4.svar <- variogram(log1p(D4)~1, Insitu_200301.XY[D4.sel,])
D4.vgm <- fit.variogram(D4.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D4),
na.rm=TRUE)/3, model="Lin"))
D4.plot=plot(D4.svar, D4.vgm)
D5.svar <- variogram(log1p(D5)~1, Insitu_200301.XY[D5.sel,])
D5.vgm <- fit.variogram(D5.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D5),
na.rm=TRUE)/3, model="Lin"))
D5.plot=plot(D5.svar, D5.vgm)
D6.svar <- variogram(log1p(D6)~1, Insitu_200301.XY[D6.sel,])
D6.vgm <- fit.variogram(D6.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D6),
na.rm=TRUE)/3, model="Lin"))
D6.plot=plot(D6.svar, D6.vgm)
D7.svar <- variogram(log1p(D7)~1, Insitu_200301.XY[D7.sel,])
D7.vgm <- fit.variogram(D7.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D7),
na.rm=TRUE)/3, model="Lin"))
D7.plot=plot(D7.svar, D7.vgm)
D8.svar <- variogram(log1p(D8)~1, Insitu_200301.XY[D8.sel,])
D8.vgm <- fit.variogram(D8.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D8),
na.rm=TRUE)/3, model="Lin"))
D8.plot=plot(D8.svar, D8.vgm)
D9.svar <- variogram(log1p(D9)~1, Insitu_200301.XY[D9.sel,])
D9.vgm <- fit.variogram(D9.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D9),
na.rm=TRUE)/3, model="Lin"))
D9.plot=plot(D9.svar, D9.vgm)
D10.svar <- variogram(log1p(D10)~1, Insitu_200301.XY[D10.sel,])
D10.vgm <- fit.variogram(D10.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D10),
na.rm=TRUE)/3, model="Lin"))
D10.plot=plot(D10.svar, D10.vgm)
D11.svar <- variogram(log1p(D11)~1, Insitu_200301.XY[D11.sel,])
D11.vgm <- fit.variogram(D11.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D11),
na.rm=TRUE)/3, model="Lin"))
D11.plot=plot(D11.svar, D11.vgm)
D12.svar <- variogram(log1p(D12)~1, Insitu_200301.XY[D12.sel,])
D12.vgm <- fit.variogram(D12.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D12),
na.rm=TRUE)/3, model="Lin"))
D12.plot=plot(D12.svar, D12.vgm)
D13.svar <- variogram(log1p(D13)~1, Insitu_200301.XY[D13.sel,])
D13.vgm <- fit.variogram(D13.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D13),
na.rm=TRUE)/3, model="Lin"))
D13.plot=plot(D13.svar, D13.vgm)
D14.svar <- variogram(log1p(D14)~1, Insitu_200301.XY[D14.sel,])
D14.vgm <- fit.variogram(D14.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D14),
na.rm=TRUE)/3, model="Lin"))
D14.plot=plot(D14.svar, D14.vgm)
D15.svar <- variogram(log1p(D15)~1, Insitu_200301.XY[D15.sel,])
D15.vgm <- fit.variogram(D15.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D15),
na.rm=TRUE)/3, model="Lin"))
D15.plot=plot(D15.svar, D15.vgm)
D16.svar <- variogram(log1p(D16)~1, Insitu_200301.XY[D16.sel,])
D16.vgm <- fit.variogram(D16.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D16),
na.rm=TRUE)/3, model="Lin"))
D16.plot=plot(D16.svar, D16.vgm)
D17.svar <- variogram(log1p(D17)~1, Insitu_200301.XY[D17.sel,])
D17.vgm <- fit.variogram(D17.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D17),
na.rm=TRUE)/3, model="Lin"))
D17.plot=plot(D17.svar, D17.vgm)
D18.svar <- variogram(log1p(D18)~1, Insitu_200301.XY[D18.sel,])
```

-3-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
D18.vgm <- fit.variogram(D18.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D18),
na.rm=TRUE)/3, model="Lin"))
D18.plot=plot(D18.svar, D18.vgm)
D19.svar <- variogram(log1p(D19)~1, Insitu_200301.XY[D19.sel,])
D19.vgm <- fit.variogram(D19.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D19),
na.rm=TRUE)/3, model="Lin"))
D19.plot=plot(D19.svar, D19.vgm)
D20.svar <- variogram(log1p(D20)~1, Insitu_200301.XY[D20.sel,])
D20.vgm <- fit.variogram(D20.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D20),
na.rm=TRUE)/3, model="Lin"))
D20.plot=plot(D20.svar, D20.vgm)
D21.svar <- variogram(log1p(D21)~1, Insitu_200301.XY[D21.sel,])
D21.vgm <- fit.variogram(D21.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D21),
na.rm=TRUE)/3, model="Lin"))
D21.plot=plot(D21.svar, D21.vgm)
D22.svar <- variogram(log1p(D22)~1, Insitu_200301.XY[D22.sel,])
D22.vgm <- fit.variogram(D22.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D22),
na.rm=TRUE)/3, model="Lin"))
D22.plot=plot(D22.svar, D22.vgm)
D23.svar <- variogram(log1p(D23)~1, Insitu_200301.XY[D23.sel,])
D23.vgm <- fit.variogram(D23.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D23),
na.rm=TRUE)/3, model="Lin"))
D23.plot=plot(D23.svar, D23.vgm)
D24.svar <- variogram(log1p(D24)~1, Insitu_200301.XY[D24.sel,])
D24.vgm <- fit.variogram(D24.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D24),
na.rm=TRUE)/3, model="Lin"))
D24.plot=plot(D24.svar, D24.vgm)
D25.svar <- variogram(log1p(D25)~1, Insitu_200301.XY[D25.sel,])
D25.vgm <- fit.variogram(D25.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D25),
na.rm=TRUE)/3, model="Lin"))
D25.plot=plot(D25.svar, D25.vgm)
D26.svar <- variogram(log1p(D26)~1, Insitu_200301.XY[D26.sel,])
D26.vgm <- fit.variogram(D26.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D26),
na.rm=TRUE)/3, model="Lin"))
D26.plot=plot(D26.svar, D26.vgm)
D27.svar <- variogram(log1p(D27)~1, Insitu_200301.XY[D27.sel,])
D27.vgm <- fit.variogram(D27.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D27),
na.rm=TRUE)/3, model="Lin"))
D27.plot=plot(D27.svar, D27.vgm)
D28.svar <- variogram(log1p(D28)~1, Insitu_200301.XY[D28.sel,])
D28.vgm <- fit.variogram(D28.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D28),
na.rm=TRUE)/3, model="Lin"))
D28.plot=plot(D28.svar, D28.vgm)
D29.svar <- variogram(log1p(D29)~1, Insitu_200301.XY[D29.sel,])
D29.vgm <- fit.variogram(D29.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D29),
na.rm=TRUE)/3, model="Lin"))
D29.plot=plot(D29.svar, D29.vgm)
D30.svar <- variogram(log1p(D30)~1, Insitu_200301.XY[D30.sel,])
D30.vgm <- fit.variogram(D30.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D30),
na.rm=TRUE)/3, model="Lin"))
D30.plot=plot(D30.svar, D30.vgm)
D31.svar <- variogram(log1p(D31)~1, Insitu_200301.XY[D31.sel,])
D31.vgm <- fit.variogram(D31.svar, model=vgm(nugget=var(log1p(Insitu_200301.XY$D31),
na.rm=TRUE)/3, model="Lin"))
D31.plot=plot(D31.svar, D31.vgm)
############################################################################### qqline
D1.qqnorm=qqnorm(Insitu_200301$D1,main="200301_D1")
D2.qqnorm=qqnorm(Insitu_200301$D2,main="200301_D2")
D3.qqnorm=qqnorm(Insitu_200301$D3,main="200301_D3")
D4.qqnorm=qqnorm(Insitu_200301$D4,main="200301_D4")
D5.qqnorm=qqnorm(Insitu_200301$D5,main="200301_D5")
D6.qqnorm=qqnorm(Insitu_200301$D6,main="200301_D6")
D7.qqnorm=qqnorm(Insitu_200301$D7,main="200301_D7")
D8.qqnorm=qqnorm(Insitu_200301$D8,main="200301_D8")
D9.qqnorm=qqnorm(Insitu_200301$D9,main="200301_D9")
D10.qqnorm=qqnorm(Insitu_200301$D10,main="200301_D10")
D11.qqnorm=qqnorm(Insitu_200301$D11,main="200301_D11")
D12.qqnorm=qqnorm(Insitu_200301$D12,main="200301_D12")
D13.qqnorm=qqnorm(Insitu_200301$D13,main="200301_D13")
D14.qqnorm=qqnorm(Insitu_200301$D14,main="200301_D14")
D15.qqnorm=qqnorm(Insitu_200301$D15,main="200301_D15")
```

-4-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
D16.qqnorm=qqnorm(Insitu_200301$D16,main="200301_D16")
D17.qqnorm=qqnorm(Insitu_200301$D17,main="200301_D17")
D18.qqnorm=qqnorm(Insitu_200301$D18,main="200301_D18")
D19.qqnorm=qqnorm(Insitu_200301$D19,main="200301_D19")
D20.qqnorm=qqnorm(Insitu_200301$D20,main="200301_D20")
D21.qqnorm=qqnorm(Insitu_200301$D21,main="200301_D21")
D22.qqnorm=qqnorm(Insitu_200301$D22,main="200301_D22")
D23.qqnorm=qqnorm(Insitu_200301$D23,main="200301_D23")
D24.qqnorm=qqnorm(Insitu_200301$D24,main="200301_D24")
D25.qqnorm=qqnorm(Insitu_200301$D25,main="200301_D25")
D26.qqnorm=qqnorm(Insitu_200301$D26,main="200301_D26")
D27.qqnorm=qqnorm(Insitu_200301$D27,main="200301_D27")
D28.qqnorm=qqnorm(Insitu_200301$D28,main="200301_D28")
D29.qqnorm=qqnorm(Insitu_200301$D29,main="200301_D29")
D30.qqnorm=qqnorm(Insitu_200301$D30,main="200301_D30")
D31.qqnorm=qqnorm(Insitu_200301$D31,main="200301_D31")
################################################################################# qqline.log
D1.qqlog=qqnorm(log1p(Insitu_200301.XY$D1),main="200301_D1")
D2.qqlog=qqnorm(log1p(Insitu_200301.XY$D2),main="200301_D2")
D3.qqlog=qqnorm(log1p(Insitu_200301.XY$D3),main="200301_D3")
D4.qqlog=qqnorm(log1p(Insitu_200301.XY$D4),main="200301_D4")
D5.qqlog=qqnorm(log1p(Insitu_200301.XY$D5),main="200301_D5")
D6.qqlog=qqnorm(log1p(Insitu_200301.XY$D6),main="200301_D6")
D7.qqlog=qqnorm(log1p(Insitu_200301.XY$D7),main="200301_D7")
D8.qqlog=qqnorm(log1p(Insitu_200301.XY$D8),main="200301_D8")
D9.qqlog=qqnorm(log1p(Insitu_200301.XY$D9),main="200301_D9")
D10.qqlog=qqnorm(log1p(Insitu_200301.XY$D10),main="200301_D10")
D11.qqlog=qqnorm(log1p(Insitu_200301.XY$D11),main="200301_D11")
D12.qqlog=qqnorm(log1p(Insitu_200301.XY$D12),main="200301_D12")
D13.qqlog=qqnorm(log1p(Insitu_200301.XY$D13),main="200301_D13")
D14.qqlog=qqnorm(log1p(Insitu_200301.XY$D14),main="200301_D14")
D15.qqlog=qqnorm(log1p(Insitu_200301.XY$D15),main="200301_D15")
D16.qqlog=qqnorm(log1p(Insitu_200301.XY$D16),main="200301_D16")
D17.qqlog=qqnorm(log1p(Insitu_200301.XY$D17),main="200301_D17")
D18.qqlog=qqnorm(log1p(Insitu_200301.XY$D18),main="200301_D18")
D19.qqlog=qqnorm(log1p(Insitu_200301.XY$D19),main="200301_D19")
D20.qqlog=qqnorm(log1p(Insitu_200301.XY$D20),main="200301_D20")
D21.qqlog=qqnorm(log1p(Insitu_200301.XY$D21),main="200301_D21")
D22.qqlog=qqnorm(log1p(Insitu_200301.XY$D22),main="200301_D22")
D23.qqlog=qqnorm(log1p(Insitu_200301.XY$D23),main="200301_D23")
D24.qqlog=qqnorm(log1p(Insitu_200301.XY$D24),main="200301_D24")
D25.qqlog=qqnorm(log1p(Insitu_200301.XY$D25),main="200301_D25")
D26.qqlog=qqnorm(log1p(Insitu_200301.XY$D26),main="200301_D26")
D27.qqlog=qqnorm(log1p(Insitu_200301.XY$D27),main="200301_D27")
D28.qqlog=qqnorm(log1p(Insitu_200301.XY$D28),main="200301_D28")
D29.qqlog=qqnorm(log1p(Insitu_200301.XY$D29),main="200301_D29")
D30.qqlog=qqnorm(log1p(Insitu_200301.XY$D30),main="200301_D30")
D31.qqlog=qqnorm(log1p(Insitu_200301.XY$D31),main="200301_D31")
################################################################################# Ordinary
kriging
setwd("D:/Interpolation/kriging")
NL32station_XY<-read.csv("./input/NL32station_XY.csv")
# fix the coordinates:
NL32station_XY$X <- NL32station_XY$X..km. * 1000
NL32station_XY$Y <- NL32station_XY$Y..km. * 1000
coordinates(NL32station_XY)<-~X+Y
################################################################################# kriging
setwd("D:/Interpolation/kriging")
D1.kriging=expm1(krige(log1p(D1)~1, Insitu_200301.XY[D1.sel,], NL32station_XY,
D1.vgm)$var1.pred)
D2.kriging=expm1(krige(log1p(D2)~1, Insitu_200301.XY[D2.sel,], NL32station_XY,
D2.vgm)$var1.pred)
D3.kriging=expm1(krige(log1p(D3)~1, Insitu_200301.XY[D3.sel,], NL32station_XY,
D3.vgm)$var1.pred)
D4.kriging=expm1(krige(log1p(D4)~1, Insitu_200301.XY[D4.sel,], NL32station_XY,
D4.vgm)$var1.pred)
D5.kriging=expm1(krige(log1p(D5)~1, Insitu_200301.XY[D5.sel,], NL32station_XY,
D5.vgm)$var1.pred)
D6.kriging=expm1(krige(log1p(D6)~1, Insitu_200301.XY[D6.sel,], NL32station_XY,
D6.vgm)$var1.pred)
D7.kriging=expm1(krige(log1p(D7)~1, Insitu_200301.XY[D7.sel,], NL32station_XY,
```

-5-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
D7.vgm)$var1.pred)
D8.kriging=expm1(krige(log1p(D8)~1, Insitu_200301.XY[D8.sel,], NL32station_XY,
D8.vgm)$var1.pred)
D9.kriging=expm1(krige(log1p(D9)~1, Insitu_200301.XY[D9.sel,], NL32station_XY,
D9.vgm)$var1.pred)
D10.kriging=expm1(krige(log1p(D10)~1, Insitu_200301.XY[D10.sel,], NL32station_XY,
D10.vgm)$var1.pred)
D11.kriging=expm1(krige(log1p(D11)~1, Insitu_200301.XY[D11.sel,], NL32station_XY,
D11.vgm)$var1.pred)
D12.kriging=expm1(krige(log1p(D12)~1, Insitu_200301.XY[D12.sel,], NL32station_XY,
D12.vgm)$var1.pred)
D13.kriging=expm1(krige(log1p(D13)~1, Insitu_200301.XY[D13.sel,], NL32station_XY,
D13.vgm)$var1.pred)
D14.kriging=expm1(krige(log1p(D14)~1, Insitu_200301.XY[D14.sel,], NL32station_XY,
D14.vgm)$var1.pred)
D15.kriging=expm1(krige(log1p(D15)~1, Insitu_200301.XY[D15.sel,], NL32station_XY,
D15.vgm)$var1.pred)
D16.kriging=expm1(krige(log1p(D16)~1, Insitu_200301.XY[D16.sel,], NL32station_XY,
D16.vgm)$var1.pred)
D17.kriging=expm1(krige(log1p(D17)~1, Insitu_200301.XY[D17.sel,], NL32station_XY,
D17.vgm)$var1.pred)
D18.kriging=expm1(krige(log1p(D18)~1, Insitu_200301.XY[D18.sel,], NL32station_XY,
D18.vgm)$var1.pred)
D19.kriging=expm1(krige(log1p(D19)~1, Insitu_200301.XY[D19.sel,], NL32station_XY,
D19.vgm)$var1.pred)
D20.kriging=expm1(krige(log1p(D20)~1, Insitu_200301.XY[D20.sel,], NL32station_XY,
D20.vgm)$var1.pred)
D21.kriging=expm1(krige(log1p(D21)~1, Insitu_200301.XY[D21.sel,], NL32station_XY,
D21.vgm)$var1.pred)
D22.kriging=expm1(krige(log1p(D22)~1, Insitu_200301.XY[D22.sel,], NL32station_XY,
D22.vgm)$var1.pred)
D23.kriging=expm1(krige(log1p(D23)~1, Insitu_200301.XY[D23.sel,], NL32station_XY,
D23.vgm)$var1.pred)
D24.kriging=expm1(krige(log1p(D24)~1, Insitu_200301.XY[D24.sel,], NL32station_XY,
D24.vgm)$var1.pred)
D25.kriging=expm1(krige(log1p(D25)~1, Insitu_200301.XY[D25.sel,], NL32station_XY,
D25.vgm)$var1.pred)
D26.kriging=expm1(krige(log1p(D26)~1, Insitu_200301.XY[D26.sel,], NL32station_XY,
D26.vgm)$var1.pred)
D27.kriging=expm1(krige(log1p(D27)~1, Insitu_200301.XY[D27.sel,], NL32station_XY,
D27.vgm)$var1.pred)
D28.kriging=expm1(krige(log1p(D28)~1, Insitu_200301.XY[D28.sel,], NL32station_XY,
D28.vgm)$var1.pred)
D29.kriging=expm1(krige(log1p(D29)~1, Insitu_200301.XY[D29.sel,], NL32station_XY,
D29.vgm)$var1.pred)
D30.kriging=expm1(krige(log1p(D30)~1, Insitu_200301.XY[D30.sel,], NL32station_XY,
D30.vgm)$var1.pred)
D31.kriging=expm1(krige(log1p(D31)~1, Insitu_200301.XY[D31.sel,], NL32station_XY,
D30.vgm)$var1.pred)
kriging=list(D1.kriging,D2.kriging,D3.kriging,D4.kriging,D5.kriging,D6.kriging,D7.kriging,D8.k
riging,D9.kriging,D10.kriging,D11.kriging,D12.kriging,D13.kriging,D14.kriging,D15.kriging,D16.
kriging,D17.kriging,D18.kriging,D19.kriging,D20.kriging,D21.kriging,D22.kriging,D23.kriging,D2
4.kriging,D25.kriging,D26.kriging,D27.kriging,D28.kriging,D29.kriging,D30.kriging,D31.kriging,
nrow=32,ncol=32)
sink("200301.txt")
print(kriging)
sink()
########################################################################################## ordinary
kriging
setwd("D:/Interpolation/kriging")
D1.OK <- krige(log1p(D1)~1, Insitu_200301.XY[D1.sel,], grids25km["nlmask25km"], D1.vgm,
nmin=10, nmax=50)
D1.OK$pred <- expm1(D1.OK$var1.pred)
write.asciigrid(D1.OK["pred"], "200301okD1.asc")
pre200301D1=D1.OK["pred"]
pre_200301D1=as.data.frame(pre200301D1)
write.table(pre_200301D1,file="pre_200301D1.txt")
D1.m=mean(expm1(D1.OK$var1.pred + 1.96*sqrt(D1.OK$var1.var))-expm1(D1.OK$var1.pred -
1.96*sqrt(D1.OK$var1.var)), na.rm=TRUE)/2
D1.sd=sd(Insitu_200301.XY$D1, na.rm=TRUE)
D2.OK <- krige(log1p(D2)~1, Insitu_200301.XY[D2.sel,], grids25km["nlmask25km"], D2.vgm,
```

-6-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
nmin=10, nmax=50)
D2.OK$pred <- expm1(D2.OK$var1.pred)
write.asciigrid(D2.OK["pred"], "200301okD2.asc")
pre200301D2=D2.OK["pred"]
pre_200301D2=as.data.frame(pre200301D2)
write.table(pre_200301D2,file="pre_200301D2.txt")
D2.m=mean(expm1(D2.OK$var1.pred + 1.96*sqrt(D2.OK$var1.var))-expm1(D2.OK$var1.pred -
1.96*sqrt(D2.OK$var1.var)), na.rm=TRUE)/2
D2.sd=sd(Insitu_200301.XY$D2, na.rm=TRUE)
D3.OK <- krige(log1p(D3)~1, Insitu_200301.XY[D3.sel,], grids25km["nlmask25km"], D3.vgm,
nmin=10, nmax=50)
D3.OK$pred <- expm1(D3.OK$var1.pred)
write.asciigrid(D3.OK["pred"], "200301okD3.asc")
pre200301D3=D3.OK["pred"]
pre_200301D3=as.data.frame(pre200301D3)
write.table(pre_200301D3,file="pre_200301D3.txt")
D3.m=mean(expm1(D3.OK$var1.pred + 1.96*sqrt(D3.OK$var1.var))-expm1(D3.OK$var1.pred -
1.96*sqrt(D3.OK$var1.var)), na.rm=TRUE)/2
D3.sd=sd(Insitu_200301.XY$D3, na.rm=TRUE)
D4.OK <- krige(log1p(D4)~1, Insitu_200301.XY[D4.sel,], grids25km["nlmask25km"], D4.vgm,
nmin=10, nmax=50)
D4.OK$pred <- expm1(D4.OK$var1.pred)
write.asciigrid(D4.OK["pred"], "200301okD4.asc")
pre200301D4=D4.OK["pred"]
pre_200301D4=as.data.frame(pre200301D4)
write.table(pre_200301D4,file="pre_200301D4.txt")
D4.m=mean(expm1(D4.OK$var1.pred + 1.96*sqrt(D4.OK$var1.var))-expm1(D4.OK$var1.pred -
1.96*sqrt(D4.OK$var1.var)), na.rm=TRUE)/2
D4.sd=sd(Insitu_200301.XY$D4, na.rm=TRUE)
D5.OK <- krige(log1p(D5)~1, Insitu_200301.XY[D5.sel,], grids25km["nlmask25km"], D5.vgm,
nmin=10, nmax=50)
D5.OK$pred <- expm1(D5.OK$var1.pred)
write.asciigrid(D5.OK["pred"], "200301okD5.asc")
pre200301D5=D5.OK["pred"]
pre_200301D5=as.data.frame(pre200301D5)
write.table(pre_200301D5,file="pre_200301D5.txt")
D5.m=mean(expm1(D5.OK$var1.pred + 1.96*sqrt(D5.OK$var1.var))-expm1(D5.OK$var1.pred -
1.96*sqrt(D5.OK$var1.var)), na.rm=TRUE)/2
D5.sd=sd(Insitu_200301.XY$D5, na.rm=TRUE)
D6.OK <- krige(log1p(D6)~1, Insitu_200301.XY[D6.sel,], grids25km["nlmask25km"], D6.vgm,
nmin=10, nmax=50)
D6.OK$pred <- expm1(D6.OK$var1.pred)
write.asciigrid(D6.OK["pred"], "200301okD6.asc")
pre200301D6=D6.OK["pred"]
pre_200301D6=as.data.frame(pre200301D6)
write.table(pre_200301D6,file="pre_200301D6.txt")
D6.m=mean(expm1(D6.OK$var1.pred + 1.96*sqrt(D6.OK$var1.var))-expm1(D6.OK$var1.pred -
1.96*sqrt(D6.OK$var1.var)), na.rm=TRUE)/2
D6.sd=sd(Insitu_200301.XY$D6, na.rm=TRUE)
D7.OK <- krige(log1p(D7)~1, Insitu_200301.XY[D7.sel,], grids25km["nlmask25km"], D7.vgm,
nmin=10, nmax=50)
D7.OK$pred <- expm1(D7.OK$var1.pred)
write.asciigrid(D7.OK["pred"], "200301okD7.asc")
pre200301D7=D7.OK["pred"]
pre_200301D7=as.data.frame(pre200301D7)
write.table(pre_200301D7,file="pre_200301D7.txt")
D7.m=mean(expm1(D7.OK$var1.pred + 1.96*sqrt(D7.OK$var1.var))-expm1(D7.OK$var1.pred -
1.96*sqrt(D7.OK$var1.var)), na.rm=TRUE)/2
D7.sd=sd(Insitu_200301.XY$D7, na.rm=TRUE)
D8.OK <- krige(log1p(D8)~1, Insitu_200301.XY[D8.sel,], grids25km["nlmask25km"], D8.vgm,
nmin=10, nmax=50)
D8.OK$pred <- expm1(D8.OK$var1.pred)
write.asciigrid(D8.OK["pred"], "200301okD8.asc")
pre200301D8=D8.OK["pred"]
pre_200301D8=as.data.frame(pre200301D8)
write.table(pre_200301D8,file="pre_200301D8.txt")
D8.m=mean(expm1(D8.OK$var1.pred + 1.96*sqrt(D8.OK$var1.var))-expm1(D8.OK$var1.pred -
1.96*sqrt(D8.OK$var1.var)), na.rm=TRUE)/2
D8.sd=sd(Insitu_200301.XY$D8, na.rm=TRUE)
D9.OK <- krige(log1p(D9)~1, Insitu_200301.XY[D9.sel,], grids25km["nlmask25km"], D9.vgm,
nmin=10, nmax=50)
```

-7-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
D9.OK$pred <- expm1(D9.OK$var1.pred)
write.asciigrid(D9.OK["pred"], "200301okD9.asc")
pre200301D9=D9.OK["pred"]
pre_200301D9=as.data.frame(pre200301D9)
write.table(pre_200301D9,file="pre_200301D9.txt")
D9.m=mean(expm1(D9.OK$var1.pred + 1.96*sqrt(D9.OK$var1.var))-expm1(D9.OK$var1.pred -
1.96*sqrt(D9.OK$var1.var)), na.rm=TRUE)/2
D9.sd=sd(Insitu_200301.XY$D9, na.rm=TRUE)
D10.OK <- krige(log1p(D10)~1, Insitu_200301.XY[D10.sel,], grids25km["nlmask25km"], D10.vgm,
nmin=10, nmax=50)
D10.OK$pred <- expm1(D10.OK$var1.pred)
write.asciigrid(D10.OK["pred"], "200301okD10.asc")
pre200301D10=D10.OK["pred"]
pre_200301D10=as.data.frame(pre200301D10)
write.table(pre_200301D10,file="pre_200301D10.txt")
D10.m=mean(expm1(D10.OK$var1.pred + 1.96*sqrt(D10.OK$var1.var))-expm1(D10.OK$var1.pred -
1.96*sqrt(D10.OK$var1.var)), na.rm=TRUE)/2
D10.sd=sd(Insitu_200301.XY$D110, na.rm=TRUE)
D11.OK <- krige(log1p(D11)~1, Insitu_200301.XY[D11.sel,], grids25km["nlmask25km"], D11.vgm,
nmin=10, nmax=50)
D11.OK$pred <- expm1(D11.OK$var1.pred)
write.asciigrid(D11.OK["pred"], "200301okD11.asc")
pre200301D11=D11.OK["pred"]
pre_200301D11=as.data.frame(pre200301D11)
write.table(pre_200301D11,file="pre_200301D11.txt")
D11.m=mean(expm1(D11.OK$var1.pred + 1.96*sqrt(D11.OK$var1.var))-expm1(D11.OK$var1.pred -
1.96*sqrt(D11.OK$var1.var)), na.rm=TRUE)/2
D11.sd=sd(Insitu_200301.XY$D11, na.rm=TRUE)
D12.OK <- krige(log1p(D12)~1, Insitu_200301.XY[D12.sel,], grids25km["nlmask25km"], D12.vgm,
nmin=10, nmax=50)
D12.OK$pred <- expm1(D12.OK$var1.pred)
write.asciigrid(D12.OK["pred"], "200301okD12.asc")
pre200301D12=D12.OK["pred"]
pre_200301D12=as.data.frame(pre200301D12)
write.table(pre_200301D12,file="pre_200301D12.txt")
D12.m=mean(expm1(D12.OK$var1.pred + 1.96*sqrt(D12.OK$var1.var))-expm1(D12.OK$var1.pred -
1.96*sqrt(D12.OK$var1.var)), na.rm=TRUE)/2
D12.sd=sd(Insitu_200301.XY$D12, na.rm=TRUE)
D13.OK <- krige(log1p(D13)~1, Insitu_200301.XY[D13.sel,], grids25km["nlmask25km"], D13.vgm,
nmin=10, nmax=50)
D13.OK$pred <- expm1(D13.OK$var1.pred)
write.asciigrid(D13.OK["pred"], "200301okD13.asc")
pre200301D13=D13.OK["pred"]
pre_200301D13=as.data.frame(pre200301D13)
write.table(pre_200301D13,file="pre_200301D13.txt")
D13.m=mean(expm1(D13.OK$var1.pred + 1.96*sqrt(D13.OK$var1.var))-expm1(D13.OK$var1.pred -
1.96*sqrt(D13.OK$var1.var)), na.rm=TRUE)/2
D13.sd=sd(Insitu_200301.XY$D13, na.rm=TRUE)
D14.OK <- krige(log1p(D14)~1, Insitu_200301.XY[D14.sel,], grids25km["nlmask25km"], D14.vgm,
nmin=10, nmax=50)
D14.OK$pred <- expm1(D14.OK$var1.pred)
write.asciigrid(D14.OK["pred"], "200301okD14.asc")
pre200301D14=D14.OK["pred"]
pre200301D14=as.data.frame(pre200301D14)
write.table(pre200301D14,file="pre200301D14.txt")
D14.m=mean(expm1(D14.OK$var1.pred + 1.96*sqrt(D14.OK$var1.var))-expm1(D14.OK$var1.pred -
1.96*sqrt(D14.OK$var1.var)), na.rm=TRUE)/2
D14.sd=sd(Insitu_200301.XY$D14, na.rm=TRUE)
D15.OK <- krige(log1p(D15)~1, Insitu_200301.XY[D15.sel,], grids25km["nlmask25km"], D15.vgm,
nmin=10, nmax=50)
D15.OK$pred <- expm1(D15.OK$var1.pred)
write.asciigrid(D15.OK["pred"], "200301okD15.asc")
pre200301D15=D15.OK["pred"]
pre_200301D15=as.data.frame(pre200301D15)
write.table(pre_200301D15,file="pre_200301D15.txt")
D15.m=mean(expm1(D15.OK$var1.pred + 1.96*sqrt(D15.OK$var1.var))-expm1(D15.OK$var1.pred -
1.96*sqrt(D15.OK$var1.var)), na.rm=TRUE)/2
D15.sd=sd(Insitu_200301.XY$D15, na.rm=TRUE)
D16.OK <- krige(log1p(D16)~1, Insitu_200301.XY[D16.sel,], grids25km["nlmask25km"], D16.vgm,
nmin=10, nmax=50)
D16.OK$pred <- expm1(D16.OK$var1.pred)
```

-8-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
write.asciigrid(D16.OK["pred"], "200301okD16.asc")
pre200301D16=D16.OK["pred"]
pre_200301D16=as.data.frame(pre200301D16)
write.table(pre_200301D16,file="pre_200301D16.txt")
D16.m=mean(expm1(D16.OK$var1.pred + 1.96*sqrt(D16.OK$var1.var))-expm1(D16.OK$var1.pred -
1.96*sqrt(D16.OK$var1.var)), na.rm=TRUE)/2
D16.sd=sd(Insitu_200301.XY$D16, na.rm=TRUE)
D17.OK <- krige(log1p(D17)~1, Insitu_200301.XY[D17.sel,], grids25km["nlmask25km"], D17.vgm,
nmin=10, nmax=50)
D17.OK$pred <- expm1(D17.OK$var1.pred)
write.asciigrid(D17.OK["pred"], "200301okD17.asc")
pre200301D17=D17.OK["pred"]
pre_200301D17=as.data.frame(pre200301D17)
write.table(pre_200301D17,file="pre_200301D17.txt")
D17.m=mean(expm1(D17.OK$var1.pred + 1.96*sqrt(D17.OK$var1.var))-expm1(D17.OK$var1.pred -
1.96*sqrt(D17.OK$var1.var)), na.rm=TRUE)/2
D17.sd=sd(Insitu_200301.XY$D17, na.rm=TRUE)
D18.OK <- krige(log1p(D18)~1, Insitu_200301.XY[D18.sel,], grids25km["nlmask25km"], D18.vgm,
nmin=10, nmax=50)
D18.OK$pred <- expm1(D18.OK$var1.pred)
write.asciigrid(D18.OK["pred"], "200301okD18.asc")
pre200301D18=D18.OK["pred"]
pre_200301D18=as.data.frame(pre200301D18)
write.table(pre_200301D18,file="pre_200301D18.txt")
D18.m=mean(expm1(D18.OK$var1.pred + 1.96*sqrt(D18.OK$var1.var))-expm1(D18.OK$var1.pred -
1.96*sqrt(D18.OK$var1.var)), na.rm=TRUE)/2
D18.sd=sd(Insitu_200301.XY$D18, na.rm=TRUE)
D19.OK <- krige(log1p(D19)~1, Insitu_200301.XY[D19.sel,], grids25km["nlmask25km"], D19.vgm,
nmin=10, nmax=50)
D19.OK$pred <- expm1(D19.OK$var1.pred)
write.asciigrid(D19.OK["pred"], "200301okD19.asc")
pre200301D19=D19.OK["pred"]
pre_200301D19=as.data.frame(pre200301D19)
write.table(pre_200301D19,file="pre_200301D19.txt")
D19.m=mean(expm1(D19.OK$var1.pred + 1.96*sqrt(D19.OK$var1.var))-expm1(D19.OK$var1.pred -
1.96*sqrt(D19.OK$var1.var)), na.rm=TRUE)/2
D19.sd=sd(Insitu_200301.XY$D19, na.rm=TRUE)
D20.OK <- krige(log1p(D20)~1, Insitu_200301.XY[D20.sel,], grids25km["nlmask25km"], D20.vgm,
nmin=10, nmax=50)
D20.OK$pred <- expm1(D20.OK$var1.pred)
write.asciigrid(D20.OK["pred"], "200301okD20.asc")
pre200301D20=D20.OK["pred"]
pre_200301D20=as.data.frame(pre200301D20)
write.table(pre_200301D20,file="pre_200301D20.txt")
D20.m=mean(expm1(D20.OK$var1.pred + 1.96*sqrt(D20.OK$var1.var))-expm1(D20.OK$var1.pred -
1.96*sqrt(D20.OK$var1.var)), na.rm=TRUE)/2
D20.sd=sd(Insitu_200301.XY$D20, na.rm=TRUE)
D21.OK <- krige(log1p(D21)~1, Insitu_200301.XY[D21.sel,], grids25km["nlmask25km"], D21.vgm,
nmin=10, nmax=50)
D21.OK$pred <- expm1(D21.OK$var1.pred)
write.asciigrid(D21.OK["pred"], "200301okD21.asc")
pre200301D21=D21.OK["pred"]
pre_200301D21=as.data.frame(pre200301D21)
write.table(pre_200301D21,file="pre_200301D21.txt")
D21.m=mean(expm1(D21.OK$var1.pred + 1.96*sqrt(D21.OK$var1.var))-expm1(D21.OK$var1.pred -
1.96*sqrt(D21.OK$var1.var)), na.rm=TRUE)/2
D21.sd=sd(Insitu_200301.XY$D21, na.rm=TRUE)
D22.OK <- krige(log1p(D22)~1, Insitu_200301.XY[D22.sel,], grids25km["nlmask25km"], D22.vgm,
nmin=10, nmax=50)
D22.OK$pred <- expm1(D22.OK$var1.pred)
write.asciigrid(D22.OK["pred"], "200301okD22.asc")
pre200301D22=D22.OK["pred"]
pre_200301D22=as.data.frame(pre200301D22)
write.table(pre_200301D22,file="pre_200301D22.txt")
D22.m=mean(expm1(D22.OK$var1.pred + 1.96*sqrt(D22.OK$var1.var))-expm1(D22.OK$var1.pred -
1.96*sqrt(D22.OK$var1.var)), na.rm=TRUE)/2
D22.sd=sd(Insitu_200301.XY$D22, na.rm=TRUE)
D23.OK <- krige(log1p(D23)~1, Insitu_200301.XY[D23.sel,], grids25km["nlmask25km"], D23.vgm,
nmin=10, nmax=50)
D23.OK$pred <- expm1(D23.OK$var1.pred)
write.asciigrid(D23.OK["pred"], "200301okD23.asc")
```

-9-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
pre200301D23=D23.OK["pred"]
pre_200301D23=as.data.frame(pre200301D23)
write.table(pre_200301D23,file="pre_200301D23.txt")
D23.m=mean(expm1(D23.OK$var1.pred + 1.96*sqrt(D23.OK$var1.var))-expm1(D23.OK$var1.pred -
1.96*sqrt(D23.OK$var1.var)), na.rm=TRUE)/2
D23.sd=sd(Insitu_200301.XY$D23, na.rm=TRUE)
D24.OK <- krige(log1p(D24)~1, Insitu_200301.XY[D24.sel,], grids25km["nlmask25km"], D24.vgm,
nmin=10, nmax=50)
D24.OK$pred <- expm1(D24.OK$var1.pred)
write.asciigrid(D24.OK["pred"], "200301okD24.asc")
pre200301D24=D24.OK["pred"]
pre_200301D24=as.data.frame(pre200301D24)
write.table(pre_200301D24,file="pre_200301D24.txt")
D24.m=mean(expm1(D24.OK$var1.pred + 1.96*sqrt(D24.OK$var1.var))-expm1(D24.OK$var1.pred -
1.96*sqrt(D24.OK$var1.var)), na.rm=TRUE)/2
D24.sd=sd(Insitu_200301.XY$D24, na.rm=TRUE)
D25.OK <- krige(log1p(D25)~1, Insitu_200301.XY[D25.sel,], grids25km["nlmask25km"], D25.vgm,
nmin=10, nmax=50)
D25.OK$pred <- expm1(D25.OK$var1.pred)
write.asciigrid(D25.OK["pred"], "200301okD25.asc")
pre200301D25=D25.OK["pred"]
pre_200301D25=as.data.frame(pre200301D25)
write.table(pre_200301D25,file="pre_200301D25.txt")
D25.m=mean(expm1(D25.OK$var1.pred + 1.96*sqrt(D25.OK$var1.var))-expm1(D25.OK$var1.pred -
1.96*sqrt(D25.OK$var1.var)), na.rm=TRUE)/2
D25.sd=sd(Insitu_200301.XY$D25, na.rm=TRUE)
D26.OK <- krige(log1p(D26)~1, Insitu_200301.XY[D26.sel,], grids25km["nlmask25km"], D26.vgm,
nmin=10, nmax=50)
D26.OK$pred <- expm1(D26.OK$var1.pred)
write.asciigrid(D26.OK["pred"], "200301okD26.asc")
pre200301D26=D26.OK["pred"]
pre_200301D26=as.data.frame(pre200301D26)
write.table(pre_200301D26,file="pre_200301D26.txt")
D26.m=mean(expm1(D26.OK$var1.pred + 1.96*sqrt(D26.OK$var1.var))-expm1(D26.OK$var1.pred -
1.96*sqrt(D26.OK$var1.var)), na.rm=TRUE)/2
D26.sd=sd(Insitu_200301.XY$D26, na.rm=TRUE)
D27.OK <- krige(log1p(D27)~1, Insitu_200301.XY[D27.sel,], grids25km["nlmask25km"], D27.vgm,
nmin=10, nmax=50)
D27.OK$pred <- expm1(D27.OK$var1.pred)
write.asciigrid(D27.OK["pred"], "200301okD27.asc")
pre200301D27=D27.OK["pred"]
pre_200301D27=as.data.frame(pre200301D27)
write.table(pre_200301D27,file="pre_200301D27.txt")
D27.m=mean(expm1(D27.OK$var1.pred + 1.96*sqrt(D27.OK$var1.var))-expm1(D27.OK$var1.pred -
1.96*sqrt(D27.OK$var1.var)), na.rm=TRUE)/2
D27.sd=sd(Insitu_200301.XY$D27, na.rm=TRUE)
D28.OK <- krige(log1p(D28)~1, Insitu_200301.XY[D28.sel,], grids25km["nlmask25km"], D28.vgm,
nmin=10, nmax=50)
D28.OK$pred <- expm1(D28.OK$var1.pred)
write.asciigrid(D28.OK["pred"], "200301okD28.asc")
pre200301D28=D28.OK["pred"]
pre_200301D28=as.data.frame(pre200301D28)
write.table(pre_200301D28,file="pre_200301D28.txt")
D28.m=mean(expm1(D28.OK$var1.pred + 1.96*sqrt(D28.OK$var1.var))-expm1(D28.OK$var1.pred -
1.96*sqrt(D28.OK$var1.var)), na.rm=TRUE)/2
D28.sd=sd(Insitu_200301.XY$D28, na.rm=TRUE)
D29.OK <- krige(log1p(D29)~1, Insitu_200301.XY[D29.sel,], grids25km["nlmask25km"], D29.vgm,
nmin=10, nmax=50)
D29.OK$pred <- expm1(D29.OK$var1.pred)
write.asciigrid(D29.OK["pred"], "200301okD29.asc")
pre200301D29=D29.OK["pred"]
pre_200301D29=as.data.frame(pre200301D29)
write.table(pre_200301D29,file="pre_200301D29.txt")
D29.m=mean(expm1(D29.OK$var1.pred + 1.96*sqrt(D29.OK$var1.var))-expm1(D29.OK$var1.pred -
1.96*sqrt(D29.OK$var1.var)), na.rm=TRUE)/2
D29.sd=sd(Insitu_200301.XY$D29, na.rm=TRUE)
D30.OK <- krige(log1p(D30)~1, Insitu_200301.XY[D30.sel,], grids25km["nlmask25km"], D30.vgm,
nmin=10, nmax=50)
D30.OK$pred <- expm1(D30.OK$var1.pred)
write.asciigrid(D30.OK["pred"], "200301okD30.asc")
pre200301D30=D30.OK["pred"]
```

-10-

E:\Interpolation_code\kriging\input\Q1-31.R 2016年2月6日 17:40

```
pre_200301D30=as.data.frame(pre200301D30)
write.table(pre_200301D30,file="pre_200301D30.txt")
D30.m=mean(expm1(D30.OK$var1.pred + 1.96*sqrt(D30.OK$var1.var))-expm1(D30.OK$var1.pred -
1.96*sqrt(D30.OK$var1.var)), na.rm=TRUE)/2
D30.sd=sd(Insitu_200301.XY$D30, na.rm=TRUE)
D31.OK <- krige(log1p(D31)~1, Insitu_200301.XY[D31.sel,], grids25km["nlmask25km"], D31.vgm,
nmin=10, nmax=50)
D31.OK$pred <- expm1(D31.OK$var1.pred)
write.asciigrid(D31.OK["pred"], "200301okD31.asc")
pre200301D31=D31.OK["pred"]
pre_200301D31=as.data.frame(pre200301D31)
write.table(pre_200301D31,file="pre_200301D31.txt")
D31.m=mean(expm1(D31.OK$var1.pred + 1.96*sqrt(D31.OK$var1.var))-expm1(D31.OK$var1.pred -
1.96*sqrt(D31.OK$var1.var)), na.rm=TRUE)/2
D31.sd=sd(Insitu_200301.XY$D31, na.rm=TRUE)
kriging.m=list(D1.m,D2.m,D3.m,D4.m,D5.m,D6.m,D7.m,D8.m,D9.m,D10.m,D11.m,D12.m,D13.m,D14.m,D15.
m,D16.m,D17.m,D18.m,D19.m,D20.m,D21.m,D22.m,D23.m,D24.m,D25.m,D26.m,D27.m,D28.m,D29.m,D30.m,D3
1.m,nrow=32,ncol=32)
sink("m200301.txt")
print(kriging.m)
sink()
kriging.sd=list(D1.sd,D2.sd,D3.sd,D4.sd,D5.sd,D6.sd,D7.sd,D8.sd,D9.sd,D10.sd,D11.sd,D12.sd,D13
.sd,D14.sd,D15.sd,D16.sd,D17.sd,D18.sd,D19.sd,D20.sd,D21.sd,D22.sd,D23.sd,D24.sd,D25.sd,D26.sd
,D27.sd,D28.sd,D29.sd,D30.sd,D31.sd,nrow=32,ncol=32)
sink("sd200301.txt")
print(kriging.sd)
sink()
##################################################################################
##################################################################################
##################################################################################
```

4. Matlab code for integrating the complex data to clear 3 hourly data for everyday over 11 years according to 32 stations order.

E:\Insitu data\again.m 2016年2月6日 17:36

```
record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Berkhout'}];
elseif data2(index,1)==251

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Hoorn(Terschelling)'}];
elseif data2(index,1)==257

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'WijkaanZee'}];
elseif data2(index,1)==260

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'De Bilt'}];
elseif data2(index,1)==267

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Stavoren'}];
elseif data2(index,1)==269

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Lelystad'}];
elseif data2(index,1)==270

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Leeuwarden'}];
elseif data2(index,1)==273

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Marknesse'}];
```

-2-

```
elseif data2(index,1)==275

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Deelen'}];
elseif data2(index,1)==277

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Lauwersoog'}];
elseif data2(index,1)==278

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Heino'}];
elseif data2(index,1)==279

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Hoogeveen'}];
elseif data2(index,1)==280

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Eelde'}];
elseif data2(index,1)==283

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Hupsel'}];
elseif data2(index,1)==286

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'NieuwBeerta'}];
elseif data2(index,1)==290

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
```

E:\Insitu data\again.m 2016年2月6日 17:36

```
),{'Twenthe'}];
elseif data2(index,1)==310

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Vlissingen'}];
elseif data2(index,1)==319

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Westdorpe'}];
elseif data2(index,1)==323

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Wilhelminadorp'}];
elseif data2(index,1)==330

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'HoekvanHolland'}];
elseif data2(index,1)==344

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Rotterdam'}];
elseif data2(index,1)==348

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'Cabauw'}];
elseif data2(index,1)==350

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
),{'GilzeRijen'}];
elseif data2(index,1)==356

record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
```

-4-

E:\Insitu data\again.m 2016年2月6日 17:36

```
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'Herwijnen'}];
        elseif data2(index,1)==370

        record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
        data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
        index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
        22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
        ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'Eindhoven'}];
        elseif data2(index,1)==375

        record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
        data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
        index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
        22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
        ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'Volkel'}];
        elseif data2(index,1)==377

        record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
        data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
        index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
        22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
        ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'Ell'}];
        elseif data2(index,1)==380

        record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
        data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
        index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
        22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
        ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'Maastricht'}];
        elseif data2(index,1)==391

        record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
        data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
        index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
        22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
        ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'Arcen'}];
    else

        record1=[data2(index,1),data2(index,8),data2(index,9),data2(index,10),data2(index,11),
        data2(index,12),data2(index,13),data2(index,14),data2(index,15),data2(index,16),data2(
        index,17),data2(index,18),data2(index,19),data2(index,20),data2(index,21),data2(index,
        22),data2(index,23),data2(index,24),data2(index,25),data2(index,26),data2(index,27),da
        ta2(index,28),data2(index,29),data2(index,30),data2(index,31),data2(index,32),data2(in
        dex,33),data2(index,34),data2(index,35),data2(index,36),data2(index,37),data2(index,38
        ),{'None'}];
            end
    outputdata=[outputdata;record1]
    end
        %output for the non hh value data, which also
            if hh-1<10
               c=['0',int2str(hh-1)];
            else
               c=int2str(hh-1);
            end
             % this is the output file floder
              filename=['C:\Users\BAO\Desktop\result\',a,b,c]
              head1=[{'# DEZE GEGEVENS MOGEN VRIJ WORDEN GEBRUIKT MITS DE VOLGENDE
              BRONVERMELDING WORDT GEGEVEN:'}]
              head2=[{'# KONINKLIJK NEDERLANDS METEOROLOGISCH INSTITUUT (KNMI)'}]
              head4=[{'# THESE DATA CAN BE USED FREELY PROVIDED THAT THE FOLLOWING SOURCE IS
```

-5-

```
        ACKNOWLEDGED:'}]
        head5=[{'# ROYAL NETHERLANDS METEOROLOGICAL INSTITUTE'}]
        head7=[{'# Dagelijkse stationsneerslag'}]

        head8=['#',{'NR'},'1','2','3','4','5','6','7','8','9','10','11','12','13','14','
        15','16','17','18','19','20','21','22','23','24','25','26','27','28','29','30','
        31','Locatie']
       if(~isempty(outputdata))
        xlswrite(filename,head1,1,'A1')
        xlswrite(filename,head2,1,'A2')
        xlswrite(filename,head4,1,'A4')
        xlswrite(filename,head5,1,'A5')
        xlswrite(filename,head7,1,'A7')
        xlswrite(filename,head8,1,'A8')
        xlswrite(filename,outputdata,1,'B9')
       end
      end
       recordhhnone=find(isnan(data(:,5)))
       yearmonthhhhnone=intersect(yearmonth,recordhhnone)
       recordsnone=data(yearmonthhhhnone,:)
       cnone='None'
       filenamenone=['C:\Users\BAO\Desktop\result\',a,b,cnone]
        head1=[{'# DEZE GEGEVENS MOGEN VRIJ WORDEN GEBRUIKT MITS DE VOLGENDE
        BRONVERMELDING WORDT GEGEVEN:'}]
        head2=[{'# KONINKLIJK NEDERLANDS METEOROLOGISCH INSTITUUT (KNMI)'}]
        head4=[{'# THESE DATA CAN BE USED FREELY PROVIDED THAT THE FOLLOWING SOURCE IS
        ACKNOWLEDGED:'}]
        head5=[{'# ROYAL NETHERLANDS METEOROLOGICAL INSTITUTE'}]
        head7=[{'# Dagelijkse stationsneerslag'}]


        head8=['#',{'NR'},'1','2','3','4','5','6','7','8','9','10','11','12','13','14','
        15','16','17','18','19','20','21','22','23','24','25','26','27','28','29','30','
        31','Locatie']
        outputdatanone=[{'here'},'is','missing','data','in',int2str(recordsnone)]
      if(~isempty(outputdatanone))
       xlswrite(filename,head1,1,'A1')
        xlswrite(filename,head2,1,'A2')
        xlswrite(filename,head4,1,'A4')
        xlswrite(filename,head5,1,'A5')
        xlswrite(filename,head7,1,'A7')
        xlswrite(filename,head8,1,'A8')
        xlswrite(filenamenone,outputdatanone,1,'B9')
       end
     end


end
```