8-2018

Divide and Recombine for Large and Complex Data: Model Likelihood Functions using MCMC and TRMM Big Data Analysis

Qi Liu
Purdue University

DIVIDE AND RECOMBINE FOR LARGE AND COMPLEX DATA: MODEL LIKELIHOOD FUNCTIONS USING MCMC AND TRMM BIG DATA ANALYSIS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Qi Liu

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2018

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. William S. Cleveland, Chair

Department of Statistics

Dr. Anindya Bhadr

Department of Statistics

Dr. Vinayak Rao

Department of Statistics

Dr. Ryan Hafen

Department of Statistics

Dr. Wen-wen Tung

Department of Earth, Atmospheric, and Planetary Sciences

Approved by:

Dr. Jun Xie

Graduate Chair of the Statistics Graduate Program

"Stay Hungry. Stay Foolish."

– Steve Jobs

ACKNOWLEDGMENTS

It is a long journey to get Ph.D. degree and I could not have finished it without the help and support of many people.

I was fortunate to have my advisor, Dr. William S. Cleveland, guide me during the PhD journey. He provided the most important thing a grad student might need – understanding and constant feedback, especially in the beginning. His attitude to pursuit of perfection in data visualization, critical thinking, and unique way of looking at data, have significantly influenced me during my Ph.D. study. Also, I would like to thank the other committee members, Dr. Vinayak Rao, Dr. Anindya Bhadra, Dr. Wen-wen Tung and Dr. Ryan Hafen, for their help and valuable advice. I have learned a lot from all of you and would not be where I am today without all your help.

Especially, I need to express my gratitude and deep appreciation to all of my former and present colleagues: Ryan, Saptarshi, Jin, Xiang, Jeremiah, Ashrith, Jianfu, Yang, Philip, Barret, Xiaosu, Jeremy, Aritra, Yuying, and Jiasen. Xiaosu, thank you for sharing your understanding of Hadoop and Rhipe. Barret, thank you for your technical support and helping me improve programming skills when I was struggling to learn Rhipe. Philip and Jiasen, thank you for sharing your initial thoughts about likelihood modeling project.

My thanks must go to all my dear friends at Purdue, who have made these years most enjoyable and memorable. Hilda, thank you for all your thoughtful suggestions and all the fun we have had as classmates and roommates. I will miss the time spent with you. Yuying, thank you for all stimulating discussion and unselfish help to me. Sophie, Rongrong, Ayu, Faye, Yumin, Boqian, Yixuan, Yixi, Jincheng, Raquel, thank you for your constructive suggestions when I resort for help. And I thank you all for spending your valuable time with me.

In particular, I am grateful to my parents and my sister for the love, support, and constant encouragement I have gotten over the years, and for letting me pursue my dream. You are the salt of the earth, and I undoubtedly could not have done this without you.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

SYMBOLS

T_i	Trend component decomposed by STL model at time i
S_i	Seasonal component decomposed by STL model at time i
R_i	Reminder component decomposed by STL model at time i
n_p	Seasonal periodicity
n_{inner}	Number of iterations in an inner loop of STL
n_{outer}	Number of iterations in an outer loop of STL
T_i^k	Trend component decomposed by STL model at time i in the end of k-th iteration
S_i^k	Seasonal component decomposed by STL model at time i in the end of k-th iteration
R_i^k	Reminder component decomposed by STL model at time i in the end of k-th iteration
s_{window}	Smooth window for the seasonal component
s_{degree}	Smooth degree for the seasonal component
t_{window}	Smooth window for the trend component
t_{degree}	Smooth degree for the trend component
$B(.)$	Bi-square weight function
m	log2 of the number of subset observations
r	log2 of the number of subsets
p	the number of the covariate variables
R^2_{McF}	McFadden R^2
Loc_i	Rain status or heavy rain status of the i-th neighborhood location (1 or 0)

ABBREVIATIONS

D&R	Divide and Recombine
Rhipe	R and Hadoop Integrated Programming Environment
HDFS	Hadoop Distributed File System
PDF	Probability Density Function
CDF	Cumulative Distribution Function
SN	Skew-normal
MLE	Maximum Likelihood Estimate
MCMC	Markov Chain Monte Carlo
MH	Metropolis-Hastings
STL	Seasonal Trend Decomposition using Loess
EM	Expectation Maximization
BLB	Bags of Little Bootstrap
TRON	Trust Region Newton Method
SVM	Support Vector Machine
LME	Likelihood Modeling Estimate
DM	Data Model
LM	Likelihood Modeling
MM	Moment Matching
NMM	Normal D&R Estimate Using MM Method
SNMM	Skew-normal D&R Estimate Using MM Method
SSNMM	Simplified Skew-normal D&R Estimate Using MM Method
K-S test	Kolmogorov-Smirnov test
CPA	Contour Probability Algorithm
TRMM	Tropical Rainfall Measuring Mission

TMPA	TRMM Multi-satellite Precipitation Analysis (Version 7 3B42)
UTC	Coordinated Universal Time
GB	Gigabyte
ROC	Receiver Operating Characteristic
AUC	The Area under the ROC Curve
NA	Not Available
SRS	Simple Random Sampling
EDA	Exploratory Data Analysis

ABSTRACT

Liu, Qi PhD, Purdue University, August 2018. Divide and Recombine for Large and Complex Data: Model Likelihood Functions using MCMC and TRMM Big Data Analysis . Major Professor: William S. Cleveland.

Divide & Recombine (D&R) is a powerful and practical statistical framework for the analysis of large and complex data. In D&R, big data are divided into subsets, each analytic method is applied to subsets with no communication among subsets, and the outputs are recombined to form a result of the analytic method for the entire data. This enables deep analysis and practical computational performance. The aim of this thesis is to provide an innovative D&R procedure to model likelihood of the generalized linear model for large data sets using Markov chain Monte Carlo (MCMC) methods and to present an analysis of Tropical Rainfall Measuring Mission (TRMM) data utilizing the DeltaRho D&R computational environment.

The first chapter briefly introduces DeltaRho computation environment, followed by the introduction of univariate and multivariate skew-normal distribution and the derivation of parameter estimation using sample moments. Then a very basic introduction to MCMC sampling is provided as the MCMC sampling method could be used to characterize the posterior distribution in Chapter 3. Finally, the chapter is closed by a nonparametric procedure for decomposing a seasonal time series into seasonal, trend and remainder components – STL.

In the second chapter, an innovate D&R procedure is proposed to compute likelihood functions of data-model (DM) parameters for big data. The likelihood-model (LM) is a parametric probability density function of the DM parameters. The density parameters are estimated by fitting the density to MCMC draws from each subset DM likelihood function, and then the fitted densities are recombined. The procedure

is illustrated using normal and skew-normal LMs for the logistic regression DM on simulated data. Also, a novel diagnostic method is developed to measure the degree of the similarity between fitted density and the true likelihood function, with a real data application illustrated in the later section.

In the last chapter, the focus is to present an analysis of TRMM big data utilizing the DeltaRho D&R computational environment. First, the exploratory data analysis is conducted to investigate the spatial patterns of precipitation and the seasonal behaviors of rain rates at different time scales. Then, spatio-temporal logistic models are constructed to explain the variation of 3-hr precipitation occurrence in automation for 460,800 locations, followed by model diagnostics and model inference. Furthermore, more advanced predictive models– two-stage logistic regression model, spatial-temporal autologistic regression model, and neighbor recurrent logistic regression model– are developed to forecast the probability of 3-hr precipitation occurrence at all locations. Finally, the chapter is ended with the application of spatio-temporal logistic models on daily heavy rainfall data.

1. BACKGROUND

1.1 Divide and Recombine (D&R) for Large Complex Data

1.1.1 D&R Statistical Framework

D&R [1] is a powerful and practical statistical framework for the analysis of large and complex data. The data are divided into subsets. Analytic methods are applied to each of the subsets, and the outputs of each method are recombined to form a result for the entire data.

First, the data are divided into subsets. Computationally, each subset is a small dataset. The division methods can be either defined by the analyst such as random division or based on a conditional variable in the dataset itself. For instance, if the dataset is a spatial temporal data, then it is reasonable to divide the data either by the time unit or by the location unit.

There are two categories of analytic methods: statistical methods (including machine learning methods), whose output is numeric and categorical, and visualization methods, whose output is visual. In practise, due to the enormous number of the subsets, only a sample of visual displays of subset can be evaluated carefully [2]. When a statistical analysis method is applied to each subset of the division, it is an embarrassingly parallel computation which means there is no communication between each subset.

Finally, the analysis results are recombined together with a selected recombination method. It can be a computational method which is applied to the outputs across all subsets to generate the final result for the whole dataset, or it can just simply combine the results of each subsets. For a statistical analytic method, the recombination results in numeric and categorical values. For example, suppose we carry out linear

regression on subsets. The outputs are the estimates of the regression coefficients, and covariance matrix of the estimates. The recombination can be simple average of the subset coefficient estimates, or means weighted by estimates of their variances. For a visualization method, the recombination is a visual display that assembles the panels for viewing across subsets.

The D&R methods used for an analytic method are critical to the success of the D&R result. We seek optimal division and recombination methods that suit the analysis task at hand. For some problems, we can obtain D&R results which is the exactly the same as what we could have when the analytic method is applied to all data directly. In many cases, however, we can only obtain recombination results that serve as approximations to the ground true.

1.1.2 DeltaRho Computation Environment

DeltaRho [3] is a computational environment to carry out D&R. It consists of two parts: the front end and back end. The front end is R [4], which is a free software environment for statistical computing and graphic. The back end is the Hadoop distributed, parallel computational environment [5] which is an open-source software framework used for distributed storage and processing of datasets of big data using the MapReduce [6] programming model. RHIPE [7], the R and Hadoop Integrated Programming Environment, builds the bridge between R and Hadoop. RHIPE allows an R user to apply D&R to large complex data wholly from within R. This saves the analyst enormous time and efforts to manage the details of the Hadoop database management and parallel processing. The only thing that the analyst needs to conduct is to specific R code for the three D&R tasks:

- divide the into subsets (D[dr] computations)
- apply the analytic method to each subset (A[dr] computations)

- recombine the outputs of the A[dr] computations and write results to the HDFS (R[dr] computations)

The data analyst writes R code to divide the data into subsets, and that create R objects containing the subsets, usually one object per subset. The code is an input to RHIPE R commands that communicate with Hadoop. The subset R objects are distributed by Hadoop across the nodes of the cluster in the HDFS. Then the analyst gives R code to RHIPE to apply an analytic method to each subset, and that create R objects containing the outputs of the method applications. The A[dr] outputs are R[dr] inputs. The analyst gives R code to RHIPE in order to recombine the R[dr] inputs, and create R objects containing the R[dr] outputs. For the RHIPE-Hadoop computation framework, A[dr] computations on the subsets are embarrassingly parallel, which means no communication between the parallel computations, the simplest possible parallel processing.

RHIPE R commands can have Hadoop write outputs of D[dr], A[dr], and R[dr] computations to the HDFS. D[dr] output objects are always written because they create division subsets which will be used multiple times in the data analysis procedure. R[dr] outputs are almost always written to the HDFS because they tend to be either a final answer for a method, or data that need to be further analyzed to get a final answer. A[dr] computations are sometimes written, but are typically not when they are just the means to the recombination end. Whether written or not, the A[dr] and R[dr] computations can be run simultaneously. Embarrassingly parallel computations that are run by Hadoop consist of the same R code being applied to each object in a collection of objects. Hadoop assigns a core to compute on an object. There are typically far more objects than cores. When a core finishes its computation on an object, Hadoop assigns it to a new object. To minimize overall elapsed read/write time when objects are read from the HDFS, the Hadoop scheduling algorithm seeks to assign a core of a node as close as possible to the node on which an object is stored. In other words, Hadoop brings the core to the data, rather than the other way around.

1.2 Skew-normal

1.2.1 Univariate Case

To illustrate how to estimate parameters of the skew-normal, we introduce some basic definitions and relevant properties of the skew-normal (SN) family (Azzalini and Valle [8]). The skew-normal density function, in one-dimensional case, is given by

$$f_1(\theta|\xi,\omega^2,\alpha) = \frac{2}{\sqrt{2\pi\omega^2}} \exp\left(-\frac{(\theta-\xi)^2}{2\omega^2}\right) \Phi(\alpha(\frac{\theta-\xi}{\omega})),\ \xi,\alpha \in \mathbb{R}, \omega \in \mathbb{R}^+,$$

where Φ denotes the cumulative distribution function (CDF) of the standard normal distribution; ξ, ω, and α are the location, scale, and shape parameters, respectively. We say $\Theta \sim SN(\xi,\omega^2,\alpha)$ if random variable Θ has density function $f_1(\theta|\xi,\omega^2,\alpha)$.

Suppose $\Theta \sim SN(\xi,\omega^2,\alpha)$ and $\Theta = \xi + \omega Z$, then

$$Z = (\Theta - \xi)/\omega,$$

which is the "normalized" random variable with a distribution $SN(0,1,\alpha)$. It's worth noting that Z has non-zero mean if $\alpha \neq 0$. More specifically, the mean, variance, and skewness of Z are

$$\mu_Z = b\delta, \quad \sigma_Z^2 = 1-\mu_Z^2, \quad \gamma_Z = \frac{4-\pi}{2}\frac{\mu_Z^3}{(1-\mu_Z^2)^{3/2}},$$

where $b = \sqrt{2/\pi}$ and $\delta = \alpha/\sqrt{(1+\alpha^2)}$. Therefore, the mean, variance and skewness of Θ are

$$\mu_\Theta = E[\Theta] = \xi + \omega\mu_Z, \tag{1.1}$$

$$\sigma_\Theta^2 = var[\Theta] = \omega^2(1-\mu_Z^2), \tag{1.2}$$

$$\gamma_\Theta = E\left[(\frac{\Theta-\mu_\Theta}{\sigma_\Theta})^3\right] = \frac{4-\pi}{2}\frac{\mu_Z^3}{(1-\mu_Z^2)^{3/2}}, \tag{1.3}$$

which form the centered parametrization of $SN(\xi,\omega^2,\alpha)$. Also these three equations imply the way to estimate parameters of $SN(\xi,\omega,\alpha)$. Given a random sample

$\theta_1, \theta_2, \cdots, \theta_n$ from distribution $SN(\xi, \omega, \alpha)$, we can calculate sample mean $\hat{\mu_\Theta}$, sample variance $\hat{\sigma_\Theta^2}$ and sample skewness $\hat{\gamma_\Theta}$. By solving equations (1.3), (1.2), (1.1), sequentially, we obtain

$$\hat{\mu}_Z = \frac{\hat{c}}{\sqrt{1+\hat{c}^2}}, \tag{1.4}$$

$$\hat{\alpha} = \frac{\hat{\mu}_Z}{\sqrt{b^2 - \hat{\mu}_Z^2}}, \tag{1.5}$$

$$\hat{\omega}^2 = \frac{\hat{\sigma}_\Theta^2}{1-\hat{\mu}_Z^2}, \tag{1.6}$$

$$\hat{\xi} = \hat{\mu}_\Theta - \hat{\omega}\hat{\mu}_Z, \tag{1.7}$$

where $\hat{c} = (\frac{2\hat{\gamma}_\Theta}{4-\pi})^{1/3}$.

The parameters estimation is straightforward when the sample is available. However, not all sample can successfully derive estimates of the parameters. As a matter of fact,

$$\delta \in (-1, 1) \Longrightarrow \mu_Z \in (-b, b).$$

Therefore,

$$\gamma_\Theta \in (-\frac{4-\pi}{2}\frac{b^3}{(1-b^2)^{3/2}}, \frac{4-\pi}{2}\frac{b^3}{(1-b^2)^{3/2}}) \approx (-0.9952717, 0.9952717).$$

If $\hat{\gamma}_\Theta$ derived from the sample falls in above region, then we call $(\hat{\mu}_\Theta, \hat{\sigma}_\Theta^2, \hat{\gamma}_\Theta)$ admissible; otherwise inadmissible. As the normal density function is a special case of the skew-normal density function with $\alpha = 0$. If a normal density is considered as a candidate approximate function for the logistic likelihood function, then the parameters of the normal density can be easily estimated by the sample mean and the sample standard error.

1.2.2 Multivariate Case

The Multivariate SN distribution has been widely discussed by Azzalini, Dalla Valle and Capitanio. Similar to the univariate case, the p-dimensional SN density function is defined by

$$f_p(\theta|\xi,\Omega,\alpha) = \frac{2}{\sqrt{(2\pi)^p|\Omega|}} \exp\left(-\frac{1}{2}(\theta-\xi)^\intercal\Omega^{-1}(\theta-\xi)\right)\Phi(\alpha^\intercal\omega^{-1}(\theta-\xi)),\ \xi,\alpha\in\mathbb{R}^p,\Omega\in\mathbb{R}^{p\times p},$$

where Ω is a $p\times p$ positive definite matrix, ξ is a vector location parameter, α is a vector shape parameter, and ω is a diagonal matrix formed by the square root of the diagonal of Ω. We say $\Theta\sim SN(\xi,\Omega,\alpha)$ if a multivariate random variable Θ has density function $f_p(\theta|\xi,\Omega,\alpha)$.

To derive the estimating formulas, let $\Theta=\xi+\omega Z$. Then

$$Z=\omega^{-1}(\Theta-\xi),$$

which is the 'normalized' variable with distribution $SN(0,\overline{\Omega},\alpha)$, where $\overline{\Omega}=\omega^{-1}\Omega\omega^{-1}$. It is worth noting that the diagonal elements of $\overline{\Omega}$ are all ones. Let $b=\sqrt{2/\pi}$, $\delta=(1+\alpha^\intercal\overline{\Omega}\alpha)^{-1/2}\overline{\Omega}\alpha$ and $\gamma_{zi}=\frac{4-\pi}{2}\frac{\mu_{zi}^3}{(1-\mu_{zi}^2)^{3/2}}$, then

$$\mu_Z=E[Z]=b\delta,\quad \Sigma_Z=var[Z]=\overline{\Omega}-\mu_Z\mu_Z^\intercal,\quad \gamma_Z=(\gamma_{z1},\ldots,\gamma_{zp}).$$

Therefore, it is trivial that

$$\begin{aligned}\mu_\Theta &= E[\Theta]=\xi+\omega\mu_Z,\\ \Sigma_\Theta &= var[\Theta]=\omega\Sigma_Z\omega=\Omega-\omega\mu_Z\mu_Z^\intercal\omega,\\ \gamma_\Theta &= \gamma_Z.\end{aligned}$$

The derivation of the parameters estimation for the multivariate skew-normal density is similar to univariate case. To simplify the notation, let $\sigma_Z=\sqrt{diag(\Sigma_Z)}$ and $\sigma_\Theta=\sqrt{diag(\Sigma_\Theta)}$, i.e. the square root of the diagonal of the variance matrix of Z and Θ, respectively. Given a multivariate random variable sample $\theta_1,\ldots,\theta_n$ drawn from distribution $SN(\xi,\Omega,\alpha)$, sample mean $\hat{\mu}_\Theta$, sample covariance $\hat{\Sigma}_\Theta$, and

component-wise skewness $\hat{\gamma}_\Theta$ can be easily computed. Then $\hat{\mu}_Z$ can be obtained by using (1.4). Therefore, the parameters could be estimated as follows:

$$\hat{\delta} = \hat{\mu}_Z / b, \quad \hat{\sigma}_Z = \sqrt{diag(I - \hat{\mu}_Z \hat{\mu}_Z^\intercal)}, \tag{1.8}$$

$$\hat{\omega} = diag(\hat{\sigma}_Z^{-1} \hat{\sigma}_\Theta), \quad \hat{\xi} = \hat{\mu}_\Theta - \omega \hat{\mu}_Z, \tag{1.9}$$

$$\hat{\Omega} = \hat{\Sigma}_\Theta + \hat{\omega} \hat{\mu}_Z \hat{\mu}_Z^\intercal \hat{\omega}, \quad \hat{\alpha} = \frac{\hat{\overline{\Omega}}^{-1} \hat{\delta}}{\sqrt{1 - \hat{\delta}^\intercal \hat{\overline{\Omega}}^{-1} \hat{\delta}}}, \tag{1.10}$$

where $diag(\hat{\sigma}_Z^{-1} \hat{\sigma}_\Theta)$ is a main diagonal matrix with components $(\hat{\sigma}_Z^{-1} \hat{\sigma}_\Theta)_{ii}$, $i = 1, \cdots, p$.

There are several properties of this estimation method. First of all, this method enables us to estimate parameters of the multivariate skew normal in a closed form, rather than in an iterative approach, which greatly reduces the computational cost. The estimation procedure for the multivariate case is an extended version of the univariate case since the multivariate case reduces to the univariate case when p = 1. Given (ξ, Ω, α), there must exist only one corresponding (μ, Σ, γ). However, not vice versa. As a matter of fact, the corresponding (ξ, Ω, α) may not exist even though (μ, Σ, γ) satisfy the constraint that Σ is positive definite. Additional constraints should include

$$\gamma_{\Theta i} \in (-\frac{4-\pi}{2} \frac{b^3}{(1-b^2)^{3/2}}, \frac{4-\pi}{2} \frac{b^3}{(1-b^2)^{3/2}}) \approx (-0.9952717, 0.9952717), \quad i = 1, \cdots, p,$$

$$1 - \delta^\intercal \overline{\Omega}^{-1} \delta > 0.$$

For the first constraint, it is implicit in the genesis of the multivariate skew-normal random variable. Because the marginal distribution of a subset of the components of the multivariate skew normal random variable is still a skew-normal random variable (Azzalini & Dalla Valle [9]). For the second constraint, it is straightforward. In order to obtain the parameters estimates, we resample the data until (ξ, Ω, α) can be estimated. In chapter 2, we will assume the sample of the logistic likelihood function is a good approximate sample of the SN distribution. Simulation studies show that (ξ, Ω, α) usually can be successfully estimated with a sample drawn from the subset

logistic likelihood for the first time when the subset likelihood function is not too flat. Due to that the likelihood is flat around the neighborhood of the maximum likelihood estimate (MLE) when the number of observations in a subset is small, the skewness of a sample drawn from a flat density function is very sensitive to the sample.

1.3 Markov Chain Monte Carlo Sampling

Markov chain Monte Carlo (MCMC) methods are a class of computer driven sampling methods ([10], [11], [12]). They enable one to characterize a distribution by randomly sampling values out of the distribution without knowing all of the distributions mathematical properties. A particular strength of MCMC is that it can be used to draw samples from distributions even when all that is known about the distribution is how to calculate the density for different samples [13].

The MCMC has two properties: Monte-Carlo and Markov chain. Monte-Carlo is a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results. The fundamental idea is to use randomness to solve problems that might be deterministic in principle. For example, a Monte-Carlo approach would be to draw a large number of random samples from a normal distribution, and calculate the sample mean of those, rather than finding the mean of a normal distribution by directly calculating it from the distributions equations. The advantage of the Monte-Carlo method is obvious: calculating the mean of a large sample of numbers can be much easier than calculating the mean directly from the normal distributions equations. This benefit is most remarkable when random samples are easy to draw, and when the distributions equations are hard to compute in other ways.

The Markov chain property of MCMC is the idea that the random samples are generated by a special sequential process. A Markov chain is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state in the previous event. The basic principle is that once this chain

has run sufficiently long enough it will find its way to the targeted distribution of interest, and we can obtain statistics of interest by using samples.

MCMC is a strategy for generating samples while exploring the state space using a Markov chain mechanism. This mechanism is constructed so that the chain spends more time in the most important regions. The difficult problem of constructing a Markov chain with the desired properties is to determine how many steps are needed to converge to the stationary distribution within an acceptable error. For different stationary distributions, appropriate MCMC algorithms should be chosen wisely to generate samples. The Metropolis-Hastings (MH) algorithm ([14], [15]), and the Gibbs sampler [16] are two most popular MCMC methods. The Gibbs sampler can considered as a special case of the MH algorithm. More theoretical results about MCMC can be found in [17].

MCMC is particularly useful in Bayesian inference due to that posterior distributions are often difficult to work with through analytic methods. More specifically, MCMC enables the user to approximate aspects of posterior distributions that cannot be directly calculated such as random samples from the posterior, and posterior means. Bayesian inference uses the information provided by observed data about a (set of) parameter(s), formally the likelihood, to update a prior state of beliefs about a (set of) parameter(s) to become a posterior state of beliefs about a (set of) parameter(s). Formally, Bayes rule is defined as

$$p(\beta|D) \propto p(D|\beta)p(\beta)$$

where β indicates a (set of) parameter(s) of interest and D indicates the data, $p(\beta|D)$ indicates the posterior of β given the data, $p(D|\beta)$ indicates the likelihood of the data given β, and $p(\beta)$ indicates the prior of β. The symbol $\propto$ means is proportional to.

The important point for this exposition is that the likelihood of the data given the parameter(s) of interest can be considered as our target posterior distribution with an uniform prior on the parameters. In this case, sampling from the likelihood is via MCMC: drawing a sequence of samples from the posterior (likelihood). In the

case of logistic regression model, Polson et al. [18] propose a new data-augmentation strategy, which leads to a simple and effective method for sampling from the logistic likelihood function. This method is fully automatic, with no tuning needed to get optimal performance. It is therefore sufficiently fast and reliable to be used as a black-box sampling routine in the models involving the logit link. Such logistic sampling, which is implemented by the R function "logit" in the R package **BayesLogit**, is applied for the subset likelihood sampling in the chapter 2. Other computationally efficient methods to draw approximate posterior samples can be found in [19] [20] [21] [22] [23] [24] [20] [25] [26] [27] [28].

1.4 Seasonal Trend Decomposition using Loess (STL)

STL [29] is a nonparametric procedure for decomposing a seasonal time series into three components: trend, seasonal and remainder. It is a powerful design for seasonal time series, which is based on a series of applications of the locally weighted regression. STL also enables analysts to specify amounts of seasonal and trend smoothing which range from a small amount of smoothing to a large amount.

1.4.1 Basic Procedure

Suppose a time series $\{Y_i\}_{i=1}^N$, where N is the total number of observations. STL decomposes it into the trend component, the seasonal component, and the remainder component which are denoted by T_i, S_i and R_i, respectively, for i = 1 to N. Then

$$Y_i = T_i + S_i + R_i.$$

In this procedure, the seasonal periodicity n_p is supposed to be predefined based on a prior knowledge of the time series. For example, the data we will demonstrate in Chapter 3 is the about monthly rain rate. There are 12 observations in each annual period, so the n_p is equal to 12. All smoothing operation are based on loess method [30]. There are two smoothing parameters for each smoothing operation

(seasonal smooth and trend smooth): window size and the degree. The window size specifies the number of observations used in local smoothing while the degree indicates the degree of locally-fitted polynomial which should be 0 or 1 (2 is optional for stlplus [31]).

Generally, the procedure of STL of decomposing a time series into these three components consists of two recursive procedures: an inner loop nested inside an outer loop. For the inner loop and the outer loop, the number of iterations n_{inner} and n_{outer} are two parameters which should be specified. Each iteration of the outer loop includes the inner loop and a computation of robustness weights which will be used in the next run of inner loop to reduce the influence of abnormal behavior on the seasonal and trend components.

The inner loop is the procedure to estimate seasonal and trend components iteratively. Suppose S_i^k, T_i^k for i = 1 to N are the seasonal and trend components at the end of k-th iteration. For the iteration k+1 in the inner loop, a detrended series is computed by subtracting T_i^k ($T_i^k = 0$ for k = 0) from Y_i. Then the seasonal component is obtained by applying smoothing operation on each cycle sub-series of the detrended series with given seasonal window s_{window} and seasonal degree s_{degree}. For the monthly rain rate data, each sub-series would be one of a collection of a sub-series of all January values, a sub-series of all February values, etc. Once the seasonal component is computed, a deseasonalized series is calculated by subtracting S_i^{k+1} from Y_i. The trend component is estimated by applying the smoothing operation on deseasonalized series with predefined trend window t_{window} and trend degree t_{degree}. It is worth noting that the seasonal fitting procedure and the trend fitting procedure compete with each other in the variation explanation of the original time series. A low-pass filter is applied to smooth cycle sub-series before the trend component estimation procedure.

After the inner loop, the remainder is calculated as follows:

$$\hat{R}_i = Y_i - \hat{S}_i - \hat{T}_i$$

We can define a weight for each observed time point using $\hat{R}_i$ for i = 1 to N. There might be some extreme observations in the original time series which result in very large $|\hat{R}_i|$. Let

$$h = 6 \times \text{median}(|\hat{R}_i|).$$

Then the robustness weight at the time point i is

$$\rho_i = B(\frac{|\hat{R}_i|}{h})$$

where B is the bi-square weight function:

$$B(x) = \begin{cases} (1-x^2)^2 & \text{if } 0 \leq x < 1 \\ 0 & \text{if } x > 1. \end{cases}$$

So the weights for each observation will be used in the next inner loop. Collectively, the inner loop and robustness computation form the outer loop and it iterate n_{outer} times. More detailed explanations of the whole procedure and other parameters can be found in [29] [31].

1.4.2 Choosing Turning Parameters

We briefly introduce 7 main parameters in the STL procedure in the previous section. They are: the seasonal periodicity n_p, the number of iterations n_{inner} and n_{outer} for the inner loop and outer loop, seasonal window s_{window} and seasonal degree s_{degree}, trend window t_{window} and trend degree t_{degree}. It is quite straightforward to specify the seasonal periodicity n_p based on the common sense of a time series. For example, $n_p = 365$ for the daily temperature due to yearly periodicity. With respect to the iteration times, $n_{inner} = 1$ or 2 is sufficient in general while $n_{outer} = 10$ provides near certainty of convergence in [29]. To be safe, we can specify a larger value for both n_{inner} and n_{outer}.

As discussed in [29], the turning procedure of seasonal window s_{window}, seasonal degree s_{degree}, trend window t_{window}, and trend degree t_{degree} can be very tricky. Each

sub-series becomes smoother as s_{window} increases. Both s_{window} and s_{degree} determine the variation in the data that makes up the seasonal component. Definitely, the choice of these two parameters depends on the characteristics of the series. According to [29], s_{window} should be odd and at least 7. If s_{window} is specified as "periodic", then each sub-series is constant and seasonal degree is redundant. In addition, visualization diagnostic plots are used to help data analysts to decide the value. The seasonal diagnostic plot demonstrates both the estimated seasonal component $\hat{S}_i$ and the detrended component $\hat{S}_i + \hat{R}_i$ against time i conditional on each sub-series. This plot can help us to balance the bias-variance trade-off in the seasonal smoothing procedure.

On the other hand, the choice of t_{window} often is restricted by the needs of the decomposition. There are two roles of the trend component in helping to estimate the seasonal component. One is to eliminate persistent, long-term variation in the data. Therefore, t_{window} is necessary to get large enough that the smoother misses even persistent effects. Another is to play a role in robustness iterations. Collectively, we need to choose t_{window} such that

$$t_{window} \geq \frac{1.5 n_p}{1 - 1.5 s_{window}^{-1}}.$$

The diagnostic plot can also be applied to determine the trend window t_{window}, and trend degree t_{degree}.

1.5 Overview of Later Chapters

In chapter 2, an innovate D&R procedure to compute likelihood functions of generalized linear regression models for big data is proposed. The likelihood-model (LM) is a parametric probability density function of the DM parameters. The density parameters are estimated by fitting the density to MCMC draws from each subset DM likelihood function, and then the fitted densities are recombined. In section 2, normal and skew-normal are presented to illustrate the choice of LM, followed by the recombination methods to formulate an approximate all-data likelihood using approximate

subset likelihoods in section 3. LM diagnostic method – contour probability algorithm is discussed in detail in section 4. Section 5 provides a real data example illustrating that the skew-normal likelihood modeling better captures the posterior density, and presents the performance of the likelihood modeling for a variety of simulated datasets. Section 6 is a concluding discussion.

Chapter 3 presents a case study: an analysis of TRMM big data using D&R methods. First, the exploratory data analysis is conducted to investigate the spatial patterns of precipitation and the seasonal behaviors of rain rates at different time scales. Then, spatio-temporal logistic models are constructed to explain the variation of 3-hr precipitation occurrence in automation for 460,800 locations, followed by model diagnostics and model inference. Furthermore, more advanced predictive models– two-stage logistic regression model, spatial-temporal autologistic regression model, and neighbor recurrent logistic regression model– are developed to forecast the probability of 3-hr precipitation occurrence at all locations. Finally, the chapter is ended with the application of spatio-temporal logistic models on daily heavy rainfall data.

2. MODEL LIKELIHOOD FUNCTIONS USING MCMC

2.1 Introduction

2.1.1 Motivation and Related Works

Statistical inference on big data is becoming increasingly important in an era when data is easily accessible and its volume grows exponentially. When a training set or an observation set becomes too large for a single machine to process, one approach to address this problem is subsampling. Kleiner et al. [32] proposed the bags of little bootstrap (BLB) approach which is a combination of subsampling, the m-out-of-n bootstrap, and the bootstrap. Ma et al. [33] presented a leveraging method in which one samples a small proportion of the data from the full sample and then performs intended computations using the small subsamples as a surrogate. Liang et al. [34] proposed a resampling-based stochastic approximation method of which at each iteration, a small subsample is drawn from the full dataset, and then the current estimate of the parameters is updated accordingly under the framework of stochastic approximation. However, these methods suffer either slow convergence rates or not full use of data.

Another solution is to divide big data into multiple small data sets and store them in multiple machines. One of the intuitive methods to address the big data challenges is to implement corresponding computing algorithms across multiple machines. It is well known that many statistical maximum likelihood estimation (MLE) problems are ultimately solved by iterative algorithms such as the Fisher's scoring algorithm or expectation maximization (EM) algorithm of Dempster et al. [35]. For example, computing MLEs of the parameters in logistic regression is a typical problem solved by iterative algorithms. In the simplest forms of these algorithms, each iterative

step requires the whole data. For small or medium data sets, we can load the whole data into memory and implement interactive algorithms to compute MLEs. However, it becomes problematic when the data are too big for a single processor because it is computationally expensive and time-consuming to combine the messages across multiple machines for each iterative step, regardless of the size of the messages being passed (Scott et al. [24]).

The third approach is to analyze data within the D&R framework. The D&R is a statistical approach to analyze large complex data by dividing the data into subsets, applying analytic methods to each subset independently with no communication among subsets, and recombining all subset results to form a result of the analytical method for the entire data [1]. In general, the first two approaches are implemented in Apache Spark [36] in which a dataset is cached in memory, while the third method is executed using MapReduce in the Hadoop ecosystem [5]. Apache Spark has in-memory cache property that makes it faster when the iterative algorithms are implemented. The primary difference between data analysis within Spark and within MapReduce processing system in Hadoop is the frequency of the communication between the nodes. Lin et al. [37] considered a distributed version of the trust region Newton method (TRON) to solve logistic regression and linear support vector machine (SVM) in Spark. Therefore, the frequency of the communication between the nodes depends on the number of iterations. In contrast, there is only one final step requiring communication among multiple nodes when using the D&R approach. Moreover, Spark does not have its own distributed system, and it processes data in memory, which means Spark requires a greater investment in memory than Hadoop does.

In the D&R paradigm, Scott et al. [24] proposed the consensus Monte Carlo algorithm that performs distributed approximate Bayesian analyses with minimal communication. The idea is to break the data into subsets, distribute each subset to a node which does a full Monte Carlo simulation from a posterior distribution given its own data, and then combine the posterior simulation from each node to

produce a set of global draws representing the consensus belief among all nodes. This method can be applied to draw a consensus sample of the posterior distribution of the coefficients in logistic regression. However, further recombination methods should be explored to accommodate models for which posterior distribution moves away from Gaussianity, especially when the dimension of the coefficients is high. Similarly, there are demands for innovative methods that provide an appropriate approximation of the coefficients in regression or classification models for distributed data with minimal communication.

Instead of computing the exact MLE of parameters in regression or classification problems for big data with many iterations across multiple cluster nodes, an appropriate approximation of an acceptable error within the D&R framework can be promising. Assume all observations are independent and identically distributed (i.i.d.). Take the logistic regression as an example, we want to seek an approximate likelihood function for the coefficient parameters. There are several reasons why we might want a likelihood in addition to the point estimate, or confidence intervals for the parameters. First, the likelihood is a natural device for combining information across observations: in particular, the likelihood for independent observations is just the product of the individual observation likelihoods. Second, prior information for parameters may be combined with the likelihood to produce a Bayesian posterior distribution for inference.

To find approximate likelihood methods for distributed large-scale data, Gautier [38] studied D&R methods for likelihood-based model fitting. More specifically, the analyst applies a division method to the data, and then parallelly computes MLE for each subset. Using the subset MLEs as well as the observed Fisher information, an analyst can fit a likelihood model on each subset. Finally, the fitted all-data likelihood is formulated by multiplying the fitted subset likelihoods. Gautier defined the maximizer of the fitted all-data likelihood as the likelihood modeling estimates (LMEs). This method is equivalent to approximate the subset likelihood function by using a normal density with a mean (the subset MLE), and variance matrix (inverse

of the observed Fisher information), up to a constant multiplier. There are two disadvantages for Gautier's method. The most serious limitation, however, is that it is based purely on the aspects of the true distribution at a specific value of the variable, and so can fail to capture important global properties. Furthermore, the inference based on the normality might be not reliable if the departure from normal assumption of the subset likelihood is serious. For example, the model can be very complex and the subset data based on some divisions might be not large enough.

For subset likelihood modeling to succeed in statistical inference within the D&R framework, the fitted likelihood should retain as much information as possible about the observed subset likelihood. In this paper, we propose a new strategy to model the subset likelihood. We consider the subset likelihood as a probability distribution function up to a multiplier constant, then draw a sample of a reasonable size from the distribution by using MCMC sampling methods. And the fitted subset likelihood is estimated by using the sample from the observed subset likelihood. Finally, all-data likelihood function is approximated by the product of fitted likelihoods of the divided subsets. This method is characterized by capturing the likelihood information by using the sample. The quality of the information greatly depends on how well the sample reflect the subset likelihood function. Therefore, the sampling method is of great importance.

2.1.2 Main Idea

The fundamental idea for the likelihood modeling within D&R framework using MCMC is as follows. Suppose that the data consist of N independent observations. Each observation contains explanatory variables $x_i \in \mathbb{R}^p$ (including intercept) and response variable y_i. The likelihood function for data model (DM) parameters on the data is a function of coefficient parameters θ given by

$$L(\theta) = \prod_{i=1}^{N} L(\theta|x_i, y_i)$$

We assume that the dataset (X, Y) is too large to reside in a singe machine. Therefore, it is divided into R subsets: $(X_1, Y_1), \ldots, (X_R, Y_R)$, each with M observations, such that $(x_{(s)i}, y_{(s)i})$ is the i-th observation of the subset (X_s, Y_s). Thus, the all-data likelihood function is given by

$$L(\theta) = \prod_{s=1}^{R} L_{(s)}(\theta), \tag{2.1}$$

which we refer to as the independent product equation, where $L_{(s)}(\theta)$ is the subset likelihood function defined by

$$L_{(s)}(\theta) = L(\theta|X_s, Y_s) = \prod_{i=1}^{M} L(\theta|x_{(s)i}, y_{(s)i}).$$

This equation indicates that under the independence assumption, the likelihood of the full data can be represented by the product of subset likelihood functions. In likelihood modeling (LM), we work with some parameterized class of distributions $g(\theta|\phi)$, where ϕ is the parameter of density function (e.g. mean and covariance matrix in the Gaussian density function). For each subset, the density parameters for pre-chosen density family are estimated by fitting the density to MCMC draws from each subset DM likelihood function. Then

$$g_{(s)}(\theta|\hat{\phi}) \approx C_s \times L_{(s)}(\theta).$$

Finally, the full-data likelihood function can be approximated by the product of the subset fitted density functions, up to a multiplicative constant.

$$L(\theta) \approx \prod_{s=1}^{R} \frac{1}{C_s} g_{(s)}(\theta|\hat{\phi}) = C \times \prod_{s=1}^{R} g_{(s)}(\theta|\hat{\phi}). \tag{2.2}$$

There are many candidate distributions $g(\theta|\phi)$, just as there are many models for DM. Of course, one thing is attempting to try is normal density as the likelihood function tends to normal when n becomes big. There are two fundamental questions:

1. How to assess whether some candidate distribution well approximates the subset likelihood function?

2. How close to the full-data likelihood function the approximated recombined likelihood function is?

To answer these two questions, we propose the contour probability algorithm to visually quantify the distance between two unnormalized density functions. The model diagnostics are applied to both subset likelihood modeling and the final all-data likelihood modeling.

2.1.3 Overview of Later Sections

The remainder of this chapter is organized as follows. In section 2, normal and skew-normal families are presented to illustrate the choice of LM. Section 3 addresses how to merge approximate subset likelihoods to formulate an approximate all-data likelihood. And the likelihood modeling algorithm is proposed for the skew-normal family. LM diagnostic method – contour probability algorithm is discussed in detail in section 4. Section 5 provides a real data example illustrating that the skew-normal likelihood modeling better captures the posterior density, and presents the performance of the likelihood modeling for a variety of simulated datasets. Section 6 is a concluding discussion.

2.2 The Choice of LM

Model building procedure can also be used for LM, including diagnostic methods to check how well LM fits the subset likelihoods and full-data likelihood. This is just like model building and checking for the DM, although the details for the diagnostics are not the same.

There are many candidates, just as there are many models for DM. Normal and skew-normal are presented here as illustrations. The modeling building and checking can, as with a DM, lead to insight about a better LM.

2.2.1 Normal Family

One thing which is attempting to try is normal density as the likelihood function tends to normal when n becomes big. Our objective is to find

$$N(\theta|\mu, \Sigma) \to L(\theta|X_s, Y_s)$$

where μ and Σ are the mean and covariance matrix of the normal distribution.

There are two approaches to estimate the parameters in the normal density function. One is to match the mode of the normal density to the mode for the subset likelihood function, which is computed by maximum likelihood estimation (MLE); and estimate the covariance matrix as a function of the Hessian matrix evaluated at the MLE. We refer this method as **Local Information** (Local) method. This method is equivalent to approximate the subset likelihood function by using a normal density with a mean (the subset MLE), and variance matrix (inverse of the observed Fisher information), up to a constant multiplier.

$$\hat{\mu} = \operatorname{argmax}_{\theta} l(\theta|X_r, Y_r)$$
$$\hat{\Sigma} = \mathcal{I}^{-1}$$

where $\mathcal{I}$ is the observed Fisher information. Another approach is to generate a sample according to the stationary function $L(\theta|X_s, Y_s)$ using Markov chain Monte Carlo (MCMC) methods, and estimate $(\hat{\mu}, \hat{\Sigma})$ using the sample moments. We call it the **Moment Matching** (MM) method.

The inference based on the normality might be not reliable if the subset likelihood seriously departs from the normal density, especially when the model can be very complex and the subset data based on some divisions might be not large enough. Therefore, we propose a more general density family – skew-normal (SN) family to model likelihoods.

2.2.2 Skew-normal Family

Generally, the MM and the MLE (Local) methods are two widely used methods for estimation of population density parameters. The MM is preferable to the Local method for the skew-normal family due to following reasons. For statistical inference, one concerns the behavior of the likelihood function and other related quantities for a sample from the SN distribution in the neighborhood of $\alpha = 0$ (the shape parameter in the skew-normal density function), a value of particular relevance since there the SN family reduces to the normal one. First, a sort of non-quadratic shape of the log-likelihood function has been exhibited with many data in Azzalini et al. [8]. Another unpleasant phenomenon is that, at $\alpha = 0$, the expected Fisher information is singular, even if all parameters are identifiable. Moreover, closed-form solutions for the maximum likelihood estimator do not exist. Therefore, we estimate parameters of the skew-normal using the MM method instead of the Local method.

As discussed in Chapter 1, the p-dimensional SN density function is defined by

$$f_p(\theta|\xi,\Omega,\alpha) = \frac{2}{\sqrt{(2\pi)^p|\Omega|}} \exp\left(-\frac{1}{2}(\theta-\xi)^\intercal\Omega^{-1}(\theta-\xi)\right)\Phi(\alpha^\intercal\omega^{-1}(\theta-\xi)),\ \xi,\alpha\in\mathbb{R}^p,\Omega\in\mathbb{R}^{p\times p},$$

where Ω is a $p \times p$ positive definite matrix, ξ is a vector location parameter, α is a vector shape parameter, and ω is a diagonal matrix formed by the square root of the diagonal of Ω. We say $\Theta \sim SN(\xi,\Omega,\alpha)$ if a multivariate random variable Θ has density function $f_p(\theta|\xi,\Omega,\alpha)$.

Given a sample generated from $L(\theta|X_s,Y_s)$ using MCMC methods, sample mean $\hat{\mu}_\Theta$, sample covariance $\hat{\Sigma}_\Theta$, and component-wise skewness $\hat{\gamma}_\Theta$ can be easily computed. There is a mapping:

$$(\hat{\xi},\hat{\Omega},\hat{\alpha}) \to (\hat{\mu}_\Theta,\hat{\Sigma}_\Theta,\hat{\gamma}_\Theta).$$

However, not vice versa. In order to obtain the parameters estimates, we resample the data until $(\hat{\xi},\hat{\Omega},\hat{\alpha})$ can be estimated. The detail derivations for the parameter estimation of the skew-normal density can be found in Chapter 1.

2.3 Recombination

In this section, we will address how to merge approximate subset likelihoods to formulate an approximate all-data likelihood function such that the overall quality of inference is resonable and accepatable comparing the one for the true likelihood function. The subset likelihood is, in general, a nontrivial function of all of the data in a given subset as it can not be expressed without reading all of the data. Therfore, the subset likelihood modelling is introduced to model each subset likelihood on some distribution family such that each fitted subset likelihood can be expressed by only a small number of distribution parameters, up to a multiplicative constant (left bottom to left top in Figure 2.1). The approximation of full-data likelihood is the product of approximate subset likelihoods (right bottom to right top in Figure 2.1). We will investigate two likelihood models in detail: skew-normal model and normal model.

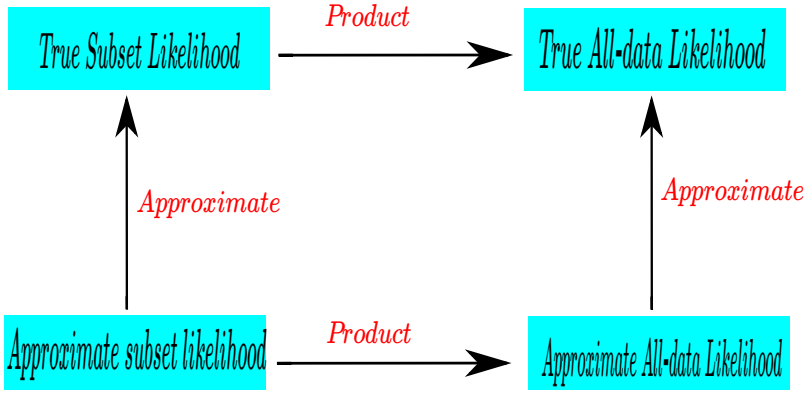


Fig. 2.1.: A diagram of likelihood modeling for big data

2.3.1 Normal Moment Matching Estimation

Recall that the likelihood function for each subset is given by

$$L_{(s)}(\theta) = \prod_{i=1}^{M} L(\theta | x_{(s)i}, y_{(s)i}).$$

which is a function of θ. Assume that subset likelihood function $L_{(s)}(\theta)$ is approximated by the normal density function $N(\theta|\hat{\mu}_s, \hat{\Sigma}_s)$, up to a multiplicative constant. The all-data likelihood function is approximated by

$$L^{Norm}(\theta) = C_1 \prod_{s=1}^{R} N(\theta|\hat{\mu}_s, \hat{\Sigma}_s),$$

Which is also normal density function, up to a multiplicative constant; and where C_1 is a constant. Therefore, the recombined approximate log likelihood for the normal model is

$$l^{Norm}(\theta) = \log L^{Norm}(\theta) = c_1 - \frac{1}{2}(\theta - \hat{\mu})^{\intercal}\hat{\Sigma}^{-1}(\theta - \hat{\mu}),$$

where c_1 is a constant; and

$$\hat{\Sigma}^{-1} = \sum_{s=1}^{R} \hat{\Sigma}_{(s)}^{-1}, \quad \hat{\mu} = \hat{\Sigma} \sum_{s=1}^{R} \hat{\Sigma}_{(s)}^{-1}\hat{\mu}_{(s)}.$$

Here $(\hat{\mu}_{(s)}, \hat{\Sigma}_{(s)})$ are sample mean and sample covariance matrix of the MCMC draws from the subset likelihood function.

Definition 2.3.1 *The normal D&R estimate using the MM method (NMM) is defined by*

$$\hat{\theta}_{NMM} = \arg\max_{\theta} l^{Norm}(\theta) = \hat{\mu}.$$

2.3.2 Skew-normal Moment Matching Estimation

Consider the subset likelihood model is limited in the skew-normal family, then $L_{(s)}(\theta)$ is approximated by the skew-normal $SN(\theta|\hat{\xi}_{(s)}, \hat{\Omega}_{(s)}, \hat{\alpha}_{(s)})$, up to a multiplicative constant. Therefore, the all-data likelihood function is approximated by

$$L^{SN}(\theta) = C_2 \prod_{s=1}^{R} SN(\theta|\hat{\xi}_{(s)}, \hat{\Omega}_{(s)}, \hat{\alpha}_{(s)}).$$

Where C_2 is a constant. The recombined approximate log likelihood for the skew-normal model is

$$l^{SN}(\theta) = \sum_{s=1}^{R} \log SN(\theta|\hat{\xi}_{(s)}, \hat{\Omega}_{(s)}, \hat{\alpha}_{(s)}) = c_2 - \frac{1}{2}(\theta - \hat{\xi})^{\intercal}\hat{\Omega}^{-1}(\theta - \hat{\xi}) + \sum_{s=1}^{R} \log \Phi\left(\hat{\lambda}_{(s)}^{\intercal}(\theta - \hat{\xi}_{(s)})\right), \tag{2.3}$$

where c_2 is a constant and

$$\hat{\Omega}^{-1} = \sum_{s=1}^{R} \hat{\Omega}_{(s)}^{-1},$$
$$\hat{\lambda}_{(s)}^{\intercal} = \hat{\alpha}_{(s)}^{\intercal} \hat{\omega}_{(s)}^{-1},$$
$$\hat{\xi} = \hat{\Omega} \sum_{s=1}^{R} \hat{\Omega}_{(s)}^{-1} \hat{\xi}_{(s)}.$$

$(\hat{\xi}_{(s)}, \hat{\Omega}_{(s)}^{-1}, \hat{\alpha}_{(s)})$ is estimated by using formulas (1.5)-(1.7) in the chapter 1 if $p = 1$ or (1.8)-(1.10) if $p > 1$; and $\hat{\omega}_{(s)}$ is the diagonal matrix formed by the square root of the diagonal of $\hat{\Omega}_{(s)}$.

Definition 2.3.2 *The skew-normal D&R estimate using the MM method (SNMM) is defined by*

$$\hat{\theta}_{SNMM} = \arg\max_{\theta} l^{SN}(\theta). \tag{2.4}$$

Actually, $l^{SN}(\theta)$ is a concave function because it is the sum of log skew normal density functions which are concave. Therefore, the recombined approximate log-likelihood for the skew-normal model is unimodal, which guarantees the local optimum is the global optimum.

To prove that the multivariate skew-normal density is concave, we assume $\theta \sim SN(\xi, \Omega, \alpha)$. Then the log density function is

$$\log f(\theta) = -\frac{1}{2} \log \left(\frac{1}{4} (2\pi)^p |\Omega| \right) - \frac{1}{2} (\theta - \xi)^{\intercal} \Omega^{-1} (\theta - \xi) + \log \Phi \left(\lambda^{\intercal} (\theta - \xi) \right),$$

where $\lambda^{\intercal} = \alpha^{\intercal} \omega^{-1}$. The first and second order relevant derivatives respect to θ are

$$\frac{\partial}{\partial \theta_k} \log f(\theta) = -(\theta - \xi)^{\intercal} \Omega_{\cdot k}^{-1} + \frac{\lambda_k \phi(\lambda^{\intercal}(\theta - \xi))}{\Phi(\lambda^{\intercal}(\theta - \xi))},$$

$$H_{j,k} = \frac{\partial^2}{\partial \theta_j \theta_k} \log f(\theta) = -\Omega_{jk}^{-1} + \lambda_j \lambda_k \frac{\phi'(\lambda^{\intercal}(\theta - \xi)) \Phi(\lambda^{\intercal}(\theta - \xi)) - \phi^2(\lambda^{\intercal}(\theta - \xi))}{\Phi^2(\lambda^{\intercal}(\theta - \xi))},$$

Where

$$\phi(\lambda^\intercal(\theta-\xi)) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\lambda^\intercal(\theta-\xi))^2},$$

$$\Phi(\lambda^\intercal(\theta-\xi)) = \int_{-\infty}^{\lambda^\intercal(\theta-\xi)} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx,$$

$$\phi'(\lambda^\intercal(\theta-\xi)) = \frac{-1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\lambda^\intercal(\theta-\xi))^2} \lambda^\intercal(\theta-\xi).$$

The $\log f(\theta)$ is concave if and only if Hessian matrix H is negative semidefinite. Let

$$g(t) = \frac{\phi'(t)\Phi(t) - \phi^2(t)}{\Phi^2(t)} = -\frac{\phi(t)(t\Phi(t) + \phi(t))}{\Phi^2(t)}.$$

It is trivial to prove that $t\Phi(t) + \phi(t) \geq 0, t \in \mathbb{R}$. Therefore, it is straightforward that $g(t) \leq 0, t \in \mathbb{R}$ and

$$v^T H v = -v^T \Omega^{-1} v + g(\lambda^\intercal(\theta-\xi))(\lambda^\intercal v)^2 < 0, v \in \mathbb{R}^p / \{0\}.$$

From the general theory about the MLE, the sampling distribution of a MLE is approximately normal. And the asymptotic estimated covariance matrix for the coefficient parameter estimates is obtained from the Fisher scoring estimation method. Specifically, the asymptotic covariance matrix is given by a function of the information matrix. Based on above approximate log likelihood function, the observed Fisher information matrix can be estimated by

$$\mathcal{I} = -\frac{\partial^2}{\partial\theta\partial\theta^T} l^{SN}(\theta) = \hat{\Omega}^{-1} - \sum_{s=1}^{R} \frac{\phi'_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)}))\Phi_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)})) - \phi^2_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)}))}{\Phi^2_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)}))} \hat{\lambda}_{(s)}\hat{\lambda}_{(s)}^\intercal,$$

where

$$\phi_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)})) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)}))^2},$$

$$\Phi_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)})) = \int_{-\infty}^{\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)})} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx,$$

$$\phi'_{(s)}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)})) = \frac{-1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)}))^2} \hat{\lambda}_{(s)}^\intercal(\theta-\hat{\xi}_{(s)})).$$

Therefore,

$$\hat{\theta}_{\text{SNMM}} \longrightarrow^{L} N(\theta, \mathcal{I}^{-1}). \tag{2.5}$$

In real world applications, the optimizer of (2.3) is not easy to compute when the number of subsets R is large. For this scenario, we propose a simplified version of the recombined log likelihood for the skew-normal model as follows:

$$l^{SSN}(\theta) = c - \frac{1}{2}(\theta - \hat{\xi})^{\intercal}\hat{\Omega}^{-1}(\theta - \hat{\xi}) + R \times \log \Phi\left(\hat{\lambda}_A^{\intercal}(\theta - \hat{\xi}_A)\right),$$

where

$$\hat{\lambda}_A^{\intercal} = \frac{\sum_{s=1}^{R} \hat{\lambda}_{(s)}^{\intercal}}{R},$$
$$\hat{\xi}_A = \sum_{s=1}^{R} \hat{\xi}_{(s)}/R.$$

Definition 2.3.3 *The simplified skew-normal D&R estimate using the MM method (SSNMM) is defined by*

$$\hat{\theta}_{SSNMM} = \arg\max_{\theta} l^{SSN}(\theta). \tag{2.6}$$

From a Bayesian perspective, the likelihood function is proportional to the posterior density function when the prior is the uniform distribution. Therefore, the recombined likelihood function provides a good approximate posterior density function, which can be used to perform statistical inference such as posterior mean estimation, credible interval computation and hypothesis testing.

Based on the above derivation, we summarize the likelihood model fitting procedure using skew-normal density as follows. In general, the distribution family that analysts choose to model the subset likelihood depends on both the data itself and data model. Therefore, it is critical to develop diagnostic methods which enable analysts to judge whether the choice of distribution family is valid.

2.4 LM Diagnostics – Contour Probability Algorithm

For univariate likelihood functions, the visible comparison between approximate likelihood and true likelihood can be achieved by plotting log likelihood ratio over a

Algorithm 1 Likelihood Model Fitting Procedure using Skew-normal Density

Require: $X, Y \quad \{X \in \mathbb{R}^{N\times p}$ and $Y \in R^N\}$

Divide (X, Y) into R submatrix $X_i \in \mathbb{R}^{M_i\times p}, Y_i \in R^{M_i}, i = 1, \ldots, R$

The following for loop is computed in parallel

for $s = 1 : R$ **do**

Generate MCMC draws according to the stationary function $L_{(s)}(\theta)$

Estimate $(\hat{\xi}_{(s)}, \hat{\omega}_{(s)}, \hat{\alpha}_{(s)})$ using MCMC draws

end for

Recombine subset approximate likelihoods to formulate the log of approximate likelihood $l^{SN}(\theta)$

Calculate the SNMM $\hat{\theta}_{\text{SNMM}}$ based on (2.4), and its covariance matrix $Cov(\hat{\theta}_{\text{SNMM}})$ using the observed Fisher information

return $(\hat{\theta}_{\text{SNMM}}, Cov(\hat{\theta}_{\text{SNMM}}))$

neighborhood of the MLE. In contrast, it is a big challenge to visualize how close one likelihood function is to another likelihood function when the dimension of the parameter vector is high. In the case of one-dimensional distributions, the Kolmogorov-Smirnov (K-S) test by Massey 1951 [39], is based on the maximum distance between the cumulative distribution functions of two histograms or probability densities. The K-S test is non-parametric and independent of the shapes of the underlying distributions. However, it does not generalize naturally to higher dimensions, and there is no widely accepted test for comparing N-dimensional distributions [40]. Another popular method is the likelihood ratio test. However, for our case, it requires computing normalizing constant of the likelihood function, which is computationally intense and numerically unstable for high dimensional functions, such as the logistic likelihood function, with a huge number of observations.

A new method is proposed to measure the similarity between approximate multivariate likelihood function and the true multivariate likelihood function without calculating the corresponding normalizing constants. Instead of using the difference between the empirical distribution function of the sample of the approximate likelihood function and the cumulative distribution function of the true likelihood distribution, we consider a series of probabilities that samples, which are drawn from the approximate likelihood, fall in regions bounded by predefined high dimensional ellipsoids, respectively. What is the contour probability? Why can contour probabilities measure the difference between two likelihood functions?

The idea of the contour probability is motivated by the Monte Carlo method. Take a univariate normal density function as an example. In Figure 2.2, the upper panel is a plot for the function $f(x) = e^{-\frac{x^2}{2}}$. Suppose the normalizing constant C is unknown even though it is known to be $\sqrt{2\pi}$, how to calculate $E = \int_a^{-a} \frac{f(x)}{C} dx$? The principle of the Monte Carlo method [41] for approximating E is to generate a sample $(x_1, \cdots, x_n)$ from the $f(x)$ and propose the empirical average as an approximation

$$\hat{E} = \frac{\sum_{i=1}^{n} I_{|x_i|<|a|}}{n}.$$

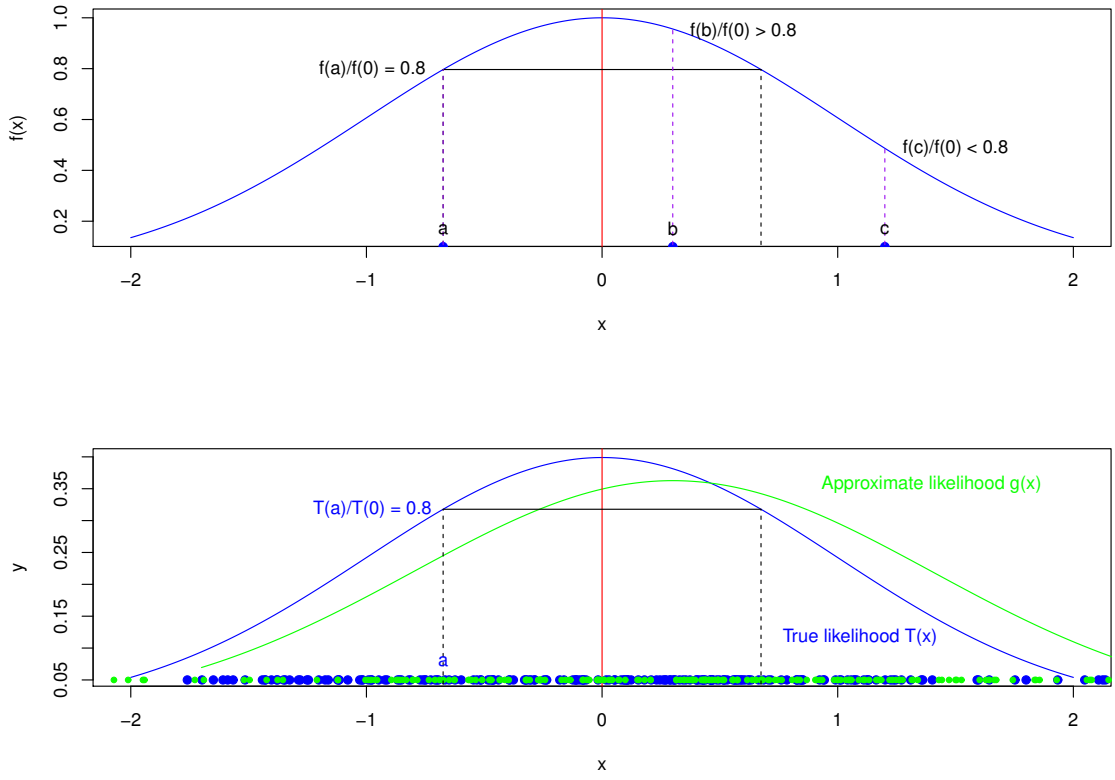


Fig. 2.2.: The upper panel displays the plot for $f(x) = e^{-\frac{x^2}{2}}$. In the lower panel, T(x) is the reference density function, which is the standard normal density function, while g(x) is the approximate density function which is the normal density function with mean 0.3 and standard error 1.1. The blue dots on the bottom are a random sample generated from T(x) and the green ones are from g(x).

As $f(x)$ is concave, it is equivalent to

$$\hat{E} = \frac{\sum_{i=1}^{n} I_{f(x_i)/f(0)>0.8}}{n}$$

where I is an indicator function. For a given ratio $h \in (0,1)$, $A_h = \{x | f(x)/f(0) > h\}$ is a region bounded by a contour, and there is only one corresponding probability $E_h = \int_{A_h} \frac{f(x)}{C} dx$. Therefore, there is a mapping

$$CP : h \in (0,1) \to E_h \in (0,1)$$

It is worth noting that the probability is estimated by using the sample generated from the target function, without knowing the normalizing constant. Also, this method can be naturally generalized to multivariate concave positive functions.

In order to demonstrate how the contour probabilities can measure the difference between two functions, we consider the probability density function of N(0,1) and N(0.3,1) as the reference function and the approximate function, respectively, which are displayed in the lower panel of Figure 2.2. Assume a sample $(x_1, \cdots, x_n)$ and a sample $(y_1, \cdots, y_n)$ are drawn from $T(x)$ and $g(y)$, respectively. For a given $h = 0.8$, $A_h = \{x|T(x)/T(0) > h\} = (a, -a)$. Then $E_T = \int_a^{-a} T(x)dx$ and $E_g = \int_a^{-a} g(y)dy$ can be estimated by

$$\hat{E_T} = \frac{\sum_{i=1}^n I_{|x_i|<|a|}}{n} \Longleftrightarrow \hat{E_T} = \frac{\sum_{i=1}^n I_{T(x_i)/T(0)>0.8}}{n}$$
$$\hat{E_g} = \frac{\sum_{i=1}^n I_{|y_i|<|a|}}{n} \Longleftrightarrow \hat{E_g} = \frac{\sum_{i=1}^n I_{T(y_i)/T(0)>0.8}}{n}$$

Therefore, there will be a pair of probabilities $(\hat{E_T}(h), \hat{E_g}(h))$ for any given ratio $h \in (0, 1)$. A series of points $(\hat{E_T}(h), \hat{E_g}(h))$ are supposed to lie around the straight line $y = x$ in that $\hat{E_g}$ is supposed to be close to $\hat{E_T}$ if g(x) well approximates T(x). Alternatively, if the contour probability difference is plotted against the contour probability of T(x), i.e. $(\hat{E_g}(h) - \hat{E_T}(h), \hat{E_T}(h))$, the points should be not far away from $y = 0$.

All of above reasoning suggests the contour probability algorithm (CPA) in algorithm 2. Assume $L(\theta)$ and $L^{approx}(\theta)$ are the true likelihood function and approximate likelihood function, respectively and $L(\theta)$ is unimodal.

2.5 Real Data and Simulated Experiments

This section proceeds through a real data example illustrating the contour probability algorithm and simulated examples for logistic regression to assess the performance of likelihood modeling on big data.

Algorithm 2 Contour Probability Algorithm (CPA)

Require: $h_i \in (0,1), i = 1, \cdots, k$, $L(\theta)$ and $L^{approx}(\theta)$

Draw a sample $(\theta_1, \cdots, \theta_{n1})$ and a sample $(\theta_1^a, \cdots, \theta_{n2}^a)$ from $L(\theta)$ and $L^{approx}(\theta)$, respectively

Compute MLE of $L(\theta)$ denoted by $\hat{\theta}_{\text{MLE}}$

for $i = 1 : k$ **do**

Count the number of the points $\tilde{\theta}$ satisfying

$$\frac{L(\tilde{\theta})}{L(\hat{\theta}_{\text{MLE}})} > h_i \Longleftrightarrow l(\tilde{\theta}) - l(\hat{\theta}_{\text{MLE}}) > log(h_i)$$

in both the approximate likelihood sample and the true likelihood sample, denoted by a_i and t_i, respectively.

$A_i := \frac{a_i}{n_2}, T_i := \frac{t_i}{n_1}$

end for

return $A = (A_1, \cdots, A_k), T = (T_1, \cdots, T_k)$,

2.5.1 Data and Model

We use one simple example to show how skew-normal likelihood modeling can capture more information of subset likelihoods or subset posterior densities. The data are the summary of exit polls in 58 counties in California (see Appendix A.1). The polls were conducted several hours before the end of the primary on June 7, 2016, with the total number of sampled people in each county fixed by design. The final goal is to predict Hillary Clintons vote share in each county, as well as her vote share in California overall. Here we are only interested in the performance of the likelihood modeling on the selected data model for this dataset. The data include following variables.

- Fips (j): The Federal Information Processing Standard (FIPS) code that uniquely identifies a county in the United States.
- Total voters (N_j): The total number of registered voters in the California Democratic primary.
- Sample voters (n_j): The total number of voters in the exit poll.
- Sample Clinton (y_j): The total number of votes for Clinton in the exit poll.

The data from counties $j = 1, ..., 58$, are assumed to follow independent binomial distributions:

$$y_j|\theta_j \sim Binomial(n_j, \theta_j), \quad j = 1, \ldots, 58,$$

with the number of sample votes, n_j, known. The parameters θ_j are assumed to be independent samples from a beta distribution:

$$\theta_j|\alpha, \beta \sim Beta(\alpha, \beta),$$

and we shall assign a noninformative hyper-prior distribution to reflect our ignorance about the unknown hyper-parameters. However, we must check that the posterior distribution is proper. One reasonable choice of the hyper-prior density of (α, β) is

$$(\alpha, \beta) \sim (\alpha + \beta)^{-5/2}.$$

The corresponding posterior density is proper as long as $0 < y_j < n_j$ for at least one experiment j [42]. Combining the sampling model for the observable y_j's and the prior distribution yields the joint posterior distribution of all the parameters and hyper-parameters, which can be expressed as follows

$$\begin{aligned} p(\alpha, \beta, \theta_1, \ldots, \theta_J) &\propto p(\alpha, \beta) \prod_{i=1}^{J} Binomial(y_i|\theta_i) Beta(\theta_i|\alpha, \beta) \\ &\propto (\alpha+\beta)^{-5/2} \prod_{i=1}^{J} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta_i^{\alpha+y_i-1}(1-\theta_i)^{n_i+\beta-y_i-1}. \end{aligned}$$

Thus we can write the marginal posterior density of the hyper-parameters as

$$p(\alpha, \beta|y) \propto (\alpha+\beta)^{-5/2} \prod_{i=1}^{J} \int \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta_i^{\alpha+y_i-1}(1-\theta_i)^{n_i+\beta-y_i-1} d\theta_i \tag{2.7}$$

$$\propto (\alpha+\beta)^{-5/2} (\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)})^J \prod_{i=1}^{J} \frac{\Gamma(\alpha+y_i)\Gamma(\beta+n_i-y_i)}{\Gamma(\alpha+\beta+n_i)} \tag{2.8}$$

2.5.2 Approximate Methods for Posterior Distribution

In this section, **Local Information**, **Moment Matching** methods with the normal family, and **Moment Matching** with the SN family are applied to approximate the posterior density.

Figure 2.3 shows the comparison between the posterior distribution of the hyper-parameters (α, β) and its approximate densities. The MM skew-normal approximation can capture the skewness of the posterior distribution while the MM normal and Local normal cannot. The distances between the mode of the true posterior and the one for the MM skew-normal approximation, MM normal, and Local normal are 0.87, 2.91, and 0, respectively.

Besides the comparison of the joint density, the comparison of the marginal density is also of interest. Figure 2.4 is a plot of the quantiles of a marginal sample from the approximate densities against the quantiles of a marginal sample from the true posterior density with a sample size 10000. Panels in the first column are Q-Q plots of marginal densities of the MM skew-normal approximate density against the

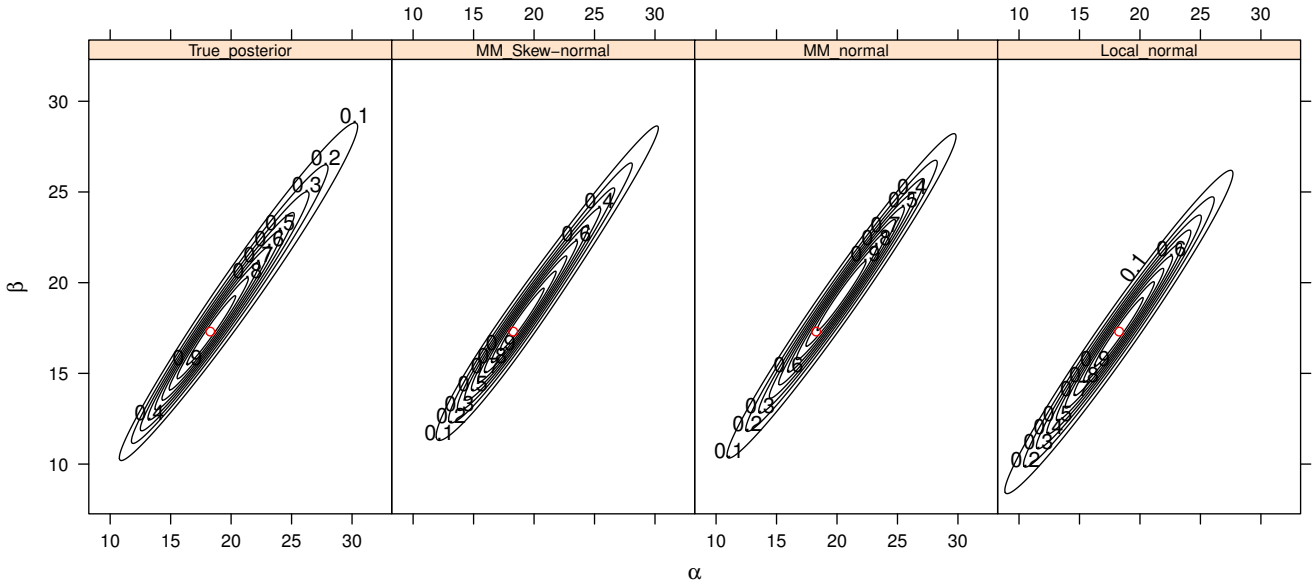


Fig. 2.3.: Comparison between the true posterior density and approximate densities. The red point in each panel is the mode of the true posterior distribution.

ones for the true posterior density. The second and third columns are for the MM normal approximate density and the Local normal approximate density against the true posterior density, respectively. Panels in the first row represent the marginal Q-Q plot for the parameter β while the ones in the second row are for α. If the two sets come from the same distribution, the points should fall approximately along the red reference line. Obviously, the MM skew-normal approximate density well approximates the true density while there is an unignorable departure from the MM normal approximation to the true density. The Local normal approximation is even worse.

Figure 2.5 demonstrates the summary comparisons of the marginal density of the hyper-parameters α and β between the true density and its approximations. Based on Figure 2.5, we can conclude that the approximation performance of the MM skew-normal approximation approach is better than the ones for the MM normal and Local normal methods in terms of the closeness of median, 50% intervals, and 95% intervals.

In order to have a deeper insight of the difference between the true posterior density and the approximation densities, we compute contour probabilities for three

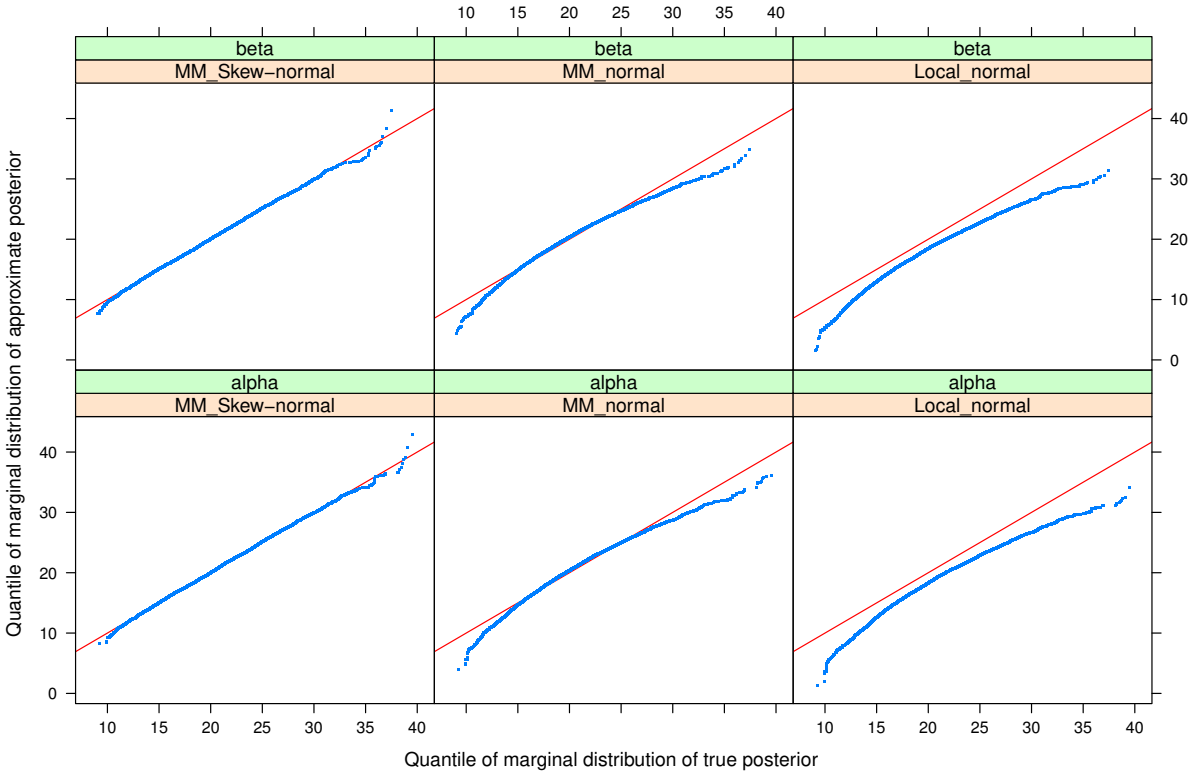


Fig. 2.4.: Pair quantile comparisons among the true posterior density and its approximate densities. The red line is a 45-degree reference line in each panel.

approximate density and true posterior function using CPA when h_i's are chosen such that $T_i \in (0.05, 0.1, \cdots, 0.95)$. Contour probability differences between approximate densities and the true posterior density are plotted against the true contour probability. Figure 2.6 indicates that the MM skew-normal approximation method significantly outperforms the MM normal and the Local normal methods.

2.5.3 Simulated Experiments

In this section, the goal is to see the performance of the likelihood modeling for logistic regresson model on a distributed data, comparing to all-data likelihood on a singe machine of the same data. Thus the data will have to be small enough for a

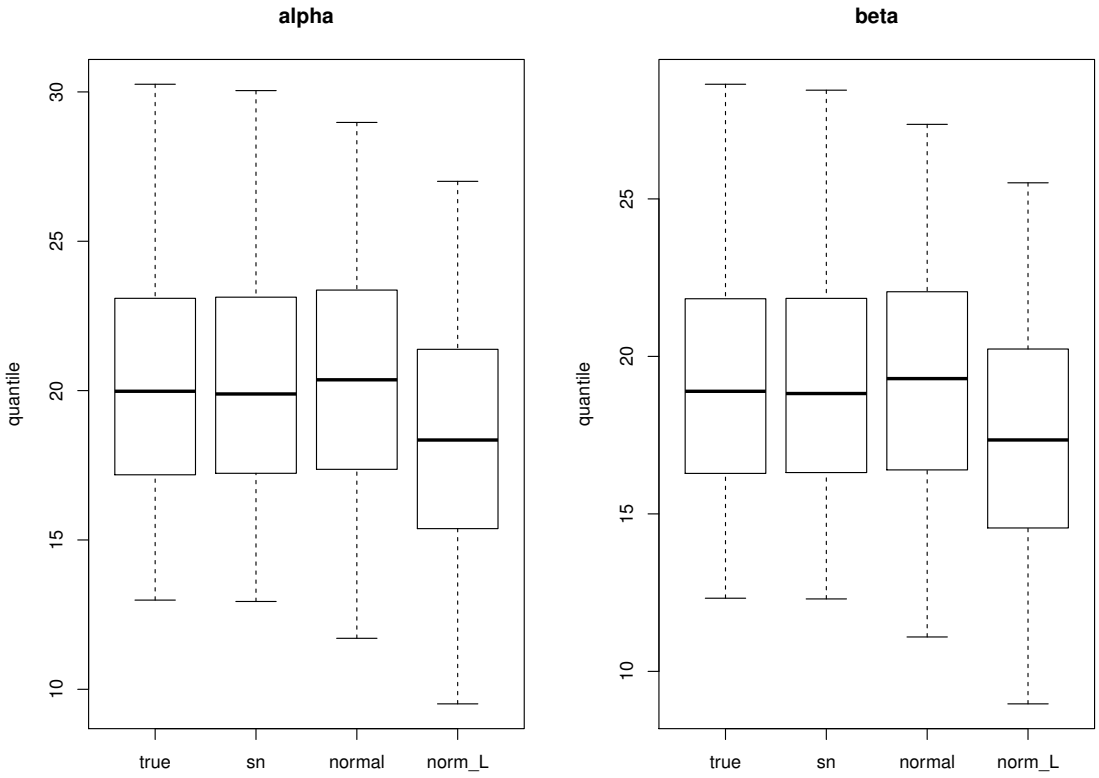


Fig. 2.5.: Estimates (medians, 50% intervals, and 95% intervals) of the marginal hyper-parameters. "true" represents the estimates from the true marginal density, "sn" stands for the estimates from the marginal density of the MM skew-normal approximation, "normal" indicates the estimates from the marginal density of the MM normal approximation, "*normal_L*" implies the estimates from the marginal density of the Local normal approximation.

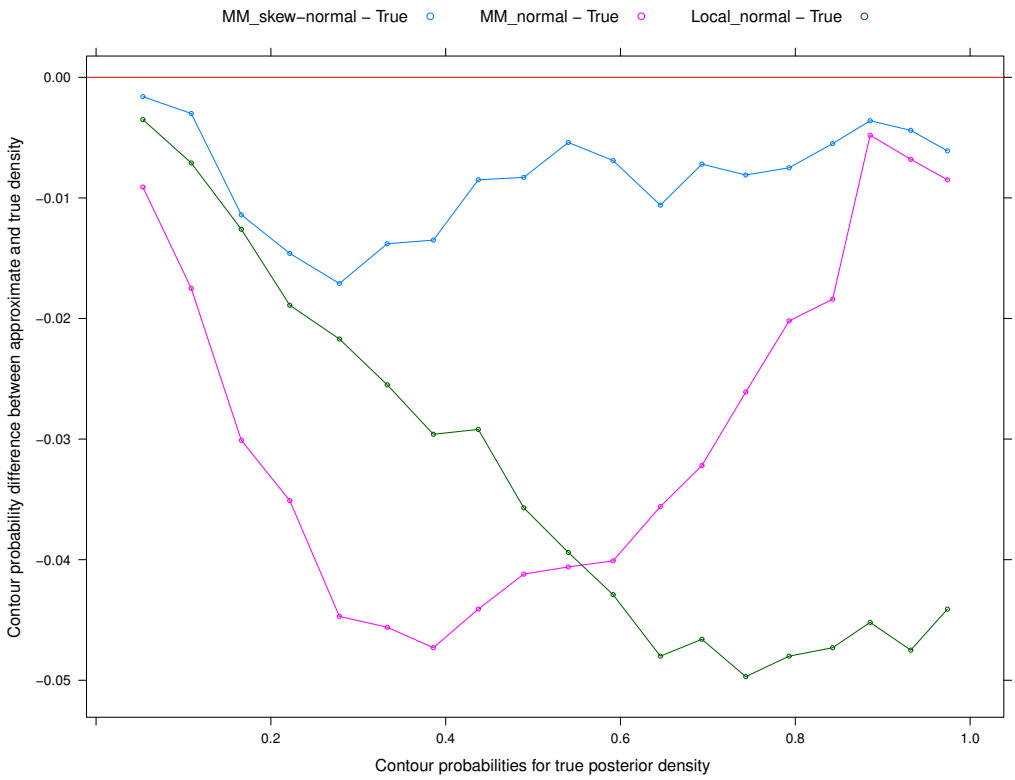


Fig. 2.6.: Contour probability differences between approximate densities and the true posterior density under series of regions bounded by ellipsoids

single machine run to be possible. To assess the performance of likelihood modeling on distributed data for the logistic regression, we set up the experiments as follows:

- run: the number of simulations
- m: log2 of the number of subset observations
- r: log2 of the number of subsets
- p: the number of the covariate variables
- Coefficient vector $\theta = (1, \cdots, 1)$
- Design matrix X with each row $x_i \overset{iid}{\sim} N^p(0, 1)$,
- Response variable Y with the element $y_i \sim Bernoulli(1/(1 + \exp^{(-x_i^T \theta)}))$

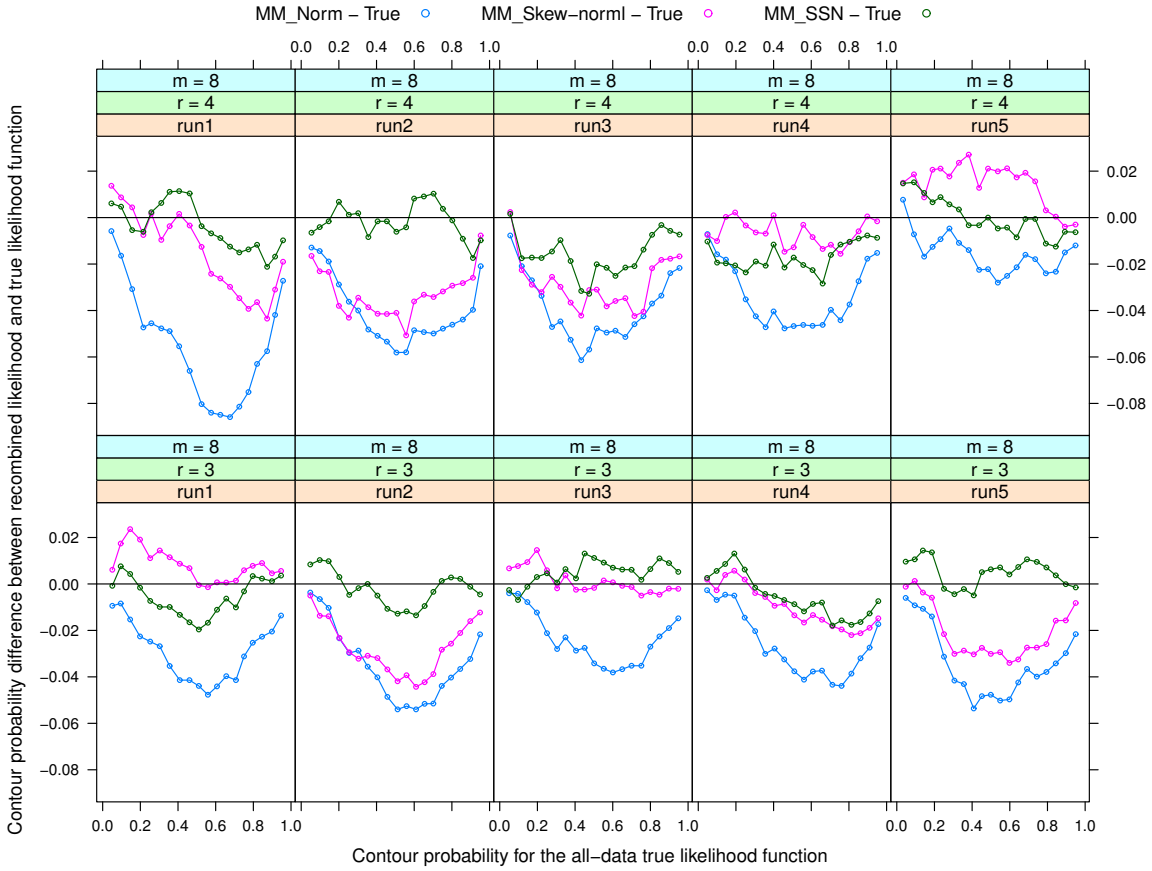


Fig. 2.7.: Scatter plots of the contour probability differences between approximate likelihoods and the true likelihood, against the true contour probability in the cases of $m = 8$, $r = 3, 4$,$run = c(1, 2, \cdots, 5)$, and $\theta = (1, 1, 1, 1, 1)$

For each combination of (m, r, run), the true likelihood function can be computed when data are generated with $p = 5$ and stored in a single machine. In contrast, the MM skew-normal approximate likelihood, MM simplified skew-normal likelihood (MM_SSN), and MM normal likelihood are estimated using the likelihood modeling algorithm when the same data are stored in a distributed cluster. Then, contour probabilities for both approximate likelihoods and true likelihood are estimated using the CPA. Figure 2.7 displays plots of the contour probability differences against the true contour probability for cases$m = 8$, $r = 3, 4$,$run = c(1, 2, \cdots, 5)$. It is straightforward that the smaller the absolute contour probability difference is, the closer to the true

likelihood function the approximate likelihood function is. The contour probabilities of the true likelihood range from 0.05 to 0.95 with a step size 0.05. Based on all panels, we can make a conclusion that the SN family are preferable to the normal family. And the MM simplified skew-normal model can be a good alternative candidate to replace the MM skew-normal model when we want to reduce computation workload for a large r.

2.5.4 Computation Performance

An analyst not only cares about how close to the true likelihood the approximate likelihood is, but also cares how fast it is to compute the approximate likelihood. Here, we compare the running time to draw 10,000 samples from the true likelihood using MCMC and the approximate likelihood on the same size of data.

Table 2.1.: Computation Performance. a) Running time (in hours) of the naive MCMC algorithm and likelihood modeling algorithm on clusters of different number of nodes for the case p = 8, 2^r = 600,000, m = 7, iterations = 10,000. b) Running time (in seconds) on different size of data using likelihood modeling on the cluster of 10 nodes.

(a)

	Number of Nodes		
Methods	10	50	500
Multi-machine MCMC	164.2	5	2.75
Likelihood Modeling	2.04		

(b)

	r		
m	8	11	14
8	126(3.96)	128(7.81)	661(7.90)
10	534(6.1)	546(4.3)	2598(6.01)
12	2104(52.1)	2165(107)	10210(279)

Scott 2013 [24] presents timings from a multi-machine MCMC algorithm for a single layer hierarchical logistic regression model on a 500-machine cluster and a 50-machine cluster. The running time to complete the job on a cluster of 500 machines

and 50 machines is 2.75 hours and 5 hours, respectively. Scott concludes that a ten-fold reduction in computing resources only produced a two-fold increase in compute time. In contrast, we run similar simulation experiments on a cluster of 10 machines using the likelihood modeling algorithm and MCMC algorithm (see Table 2.1 (a)). All experiments are implemented on the WSC Cluster which consists of 10 nodes with total 200 cores, 128 GB RAM, 128.9 TB disk and 10 Gbps Ethernet interconnect. And all machines are running R version 3.3.1, Java 1.7.0_07b10, Cloudera Hadoop 0.20.2-cdh3u5 and Rhipe 0.75 [1]. The likelihood modeling algorithm reduced computation time in 80 folds with the same cluster setting. There might be a smarter way of setting up MCMC algorithm to reduce computation time. The bottleneck of the multi-machine MCMC algorithm is that the iterative algorithm is implemented as a chain of jobs where the output from each job is used as input to the next job.

The next test case is to run experiments to assess computation performance of the likelihood modeling algorithm. The test cases are all combinations of $r = (8, 11, 14), m = c(8, 10, 12)$ for run = 3, p = 10. The value in each cell at Table 2.1 (b) is the average of three runs while the value in parenthesis is the corresponding standard deviation of the three runs. It is noticing that the running time does not increase much when r increases from 8 to 11 with m fixed. Given m, the running time for r=14 is around 5 times the one for r = 11. The one possible explanation is that jobs for r = 11 make full use of all containers while there are some idle containers when running jobs for r = 8.

2.6 Discussion

We have proposed an innovative D&R procedure to model the likelihood of generalized linear regression models on distributed datasets. There are many candidate models for likelihoods, just as there are many models for DM. Normal family and skew-normal family have been investigated to illustrate the likelihood modeling procedure. Also, we discussed two methods to estimate parameters of the given likelihood model

family: MM with MCMC draws and Local method. Moreover, the contour probability algorithm has been introduced to measure the similarity between approximate multivariate likelihood function and the true multivariate likelihood function. In terms of accuracy, the MM skew-normal likelihood model outperforms normal likelihood model in the application of CPA on Exit Poll data. On the computation of point view, the likelihood modeling definitely speeds up computation for generalized linear models, keeping the inference capability for big data. As the likelihood modeling procedure is designed to work in the D&R framework.

In summary, the likelihood modeling algorithm can provide a relatively accurate estimate of the MLE of the parameters in the generalized linear model; it is well aligned with modern parallel and distributed computing architectures and is scalable to very large datasets.

Nevertheless, the likelihood modeling has some limitations. First of all, LM is constructed under the assumption that all observations are independent. Second, MCMC sampling method is used to generate a sample based on the subset likelihood function. There is a trade-off between computation time and the effective sample, especially in high dimension space. There are two possible future work. One of the potential future works is to modify methods within the D&R framework for non-iid data. Another follow-up work is to investigate more efficient strategies to capture information of the subset likelihood.

3. MODELING FOR TRMM BIG DATA

3.1 Introduction

Rainfall is one of the most eminent and complex atmospheric phenomena. It is complex as it involves interaction of several atmospheric processes and is vital for the survival and sustenance of the earth. Precipitation patterns and rainfall time series forecasting are two of the most important issues in many real-world applications such tropical cyclones, extreme weather, floods, landslides, climate prediction, soil moisture, agriculture, freshwater availability and world health. Due to the complexity of the atmospheric processes that generate rainfall, it is very challenge for researchers to propose good global models.

In general, there are two kinds of models for rainfall. First, the conceptual physical approach entails using the fundamental laws of physics to represent and explain the hydrological processes governing the behavior of the hydrosystem. Another is statistical models that are created based upon historical observations and the climatological conditions for specific locations. A statistical model for rainfall has at least two useful properties: (1) it can describe the relationship between rainfall at a given location and other weather-related variables, such as climate variables and rainfall observed at other nearby locations, in order to reduce the unexplained variation in rainfall amounts, and (2) it provides a principled way to quantify the uncertainty that accompanies rainfall processes.

From a statistical point of view, one of challenges in precipitation modeling is that the probability distribution of precipitation depends on the space-time averaging scale [43] as precipitation has a high spatial and temporal variability. In general, precipitation data are measured as averages over space-time scales determined by the mechanism and resolution. The rainfall variability decreases with increasing space

Table 3.1.: The percentage of rainfall occurrences for different averaging time windows from 10-minute to 91-day, where 30-day and 91-day represent the monthly and seasonal cases, respectively.

Time	10-min	15-min	30-min	1-hr	3-hr	6-hr	1-day	1-week	30-day	91-day
Percentage	1.77	2.55	4.91	6.47	10.42	14.77	32.71	88.57	99.76	100

or time doman over which an average is taken. This leads to the fact that temporal and areal average are very similar. Hudlow and Patterson [44] showed for the rainfall during GATE measured by radar that the hourly rainfall averaged over an area of $28\times28km^2$ has a very similar statistical distribution as the daily rainfall for a $4\times4km^2$ area. Also, Table 3.1 demonstrates the percentage of nonzero rainfall for different averaging time windows over 12 irregularly sites in Virginia, Maryland, and North Carolina [45]. By analyzing rain rates on different space-time averaging scales, it is easy to see that precipitation statistics are strongly scale dependent [46]. For example, the range of spatial dependence for monthly rain rates is much larger than that for hourly rain rates.

Another challenge arises due to a particular feature of precipitation fields. A mixed distribution with a point mass probability of zeros is often used to describe the frequent occurrence of rainfall zeros [47]. The spatio-temporal dependence in rainfall zeros is a critical aspect of any space-time stochastic model for precipitation. For the daily precipitation, Zheng and Katz [48] proposed a approach for modeling the spatial dependence in rainfall occurrence using the previous state information at multiple sites. Hughes and Guttorp [49] used non-homogeneous hidden Markov model to relate atmospheric circulation to precipitation occurrence at 30 rain-gauge stations in south-western Australia.

Modeling the spatio-temporal dependence is necessary to better characterize the movement or the spatial patterns of the precipitation over short time scales. Although much progress has been achieved in the development of precipitation modeling, the

generation of multisite precipitation sequences with realistic spatial dependence remains a challenge even for the daily time scale. Precipitation models in previous works are commonly developed for daily data and mostly focus on reproducing means of the precipitation [45]. Devi et al. [50] applied different neural network models such as feed forward back propagation neural network (BPN), cascade-forward back propagation neural network (CBPN), distributed time delay neural network (DTDNN) and nonlinear autoregressive exogenous network (NARX), and compared their forecasting capabilities for daily rainfall prediction at Nilgiris and Coonoor. Mislan et al. [51] investigated that BPN algorithm has provided a good model to predict monthly rainfall in Tenggarong, East Kalimantan - Indonesia.

In this Chapter, we focus on building explanatory models for 3-hr rainfall occurrence based on the joint use of spatial and temporal features, and predictive models which can produce the conditional rain probabilities given historical data at the center location and its neighborhood. The investigation uses the Tropical Rainfall Measuring Mission (TRMM) version 7 3B42 Multi-satellite Precipitation Analysis (TMPA) data.

The remainder of this chapter is organized as follows. We briefly introduce TRMM data and the goal of data analysis in section 2 and 3, respectively. Data preparation procedures such as handling missing values and sampling methods are discussed in section 4, followed by the exploratory data analysis in section 5. Section 6 illustrates the procedure of builiding explanatory models for 3-hr rainfall occurrence. In section 7, we develop two-stage logistic regression models, Markov random field model, and neighbor recurrent logistic regression model to forcast 3-hr rainfall occurrence. Then, we extend the application of the spatial temporal logistic regression model to the extreme weather– daily heavy rainfall in section 8. The chapter is closed with the conclusion in section 9.

3.2 Data

The Tropical Rainfall Measuring Mission (TRMM), a joint mission of NASA and the Japan Aerospace Exploration Agency, was launched in 1997 to study rainfall for weather and climate research. The TRMM is the first coordinated international effort to provide reliable rainfall measurement from space. The data [52] are estimated by using a calibration-based sequential scheme for combining precipitation estimates from multiple satellites, as well as gauge analyses where feasible. They consist of 3-hourly precipitation rates (mm/hr) from 1998-01-01 00:00 UTC to 2015-04-30 21:00 UTC (50632 time steps) on a fixed degree latitude-longitude grid (0.25 x 0.25), globally from 50S to 50N (1440 x 400 locations). The total size of the dataset is $50632 \times 1440 \times 400 \times 8/2^{30} = 217.289$ GB.

We can view data in two different perspectives: by time and by location. For the fixed time t, the subset data consist of 3-hourly rain rates at $0.25^\circ \times 0.25^\circ$ latitude-longitude resolution from 50S to 50N. If we divide the whole data by the location. For each location, the data are a 3-hourly rain rate time series of length 50632 with coverage from 1998-01-01 00 to 2015-04-30 21 UTC.

It shows that data quality on high latitudes $40^\circ - 50^\circ N(S)$ is inconsistent with ones on lower latitudes $0^\circ - 40^\circ N(S)$. Also, there are a high number of missing observations and a large length of consecutive missing runs in 1998 due to the lack of satellite over the Indian Ocean for half of the year and probably a few days of missing data from another geostationary satellite over Asia [53]. Therefore, we restrict our data analysis on the location from $40^\circ S$ to $40^\circ N$ rather than from $50^\circ S$ to $50^\circ N$, and eliminated the data of the first half year. This results in a great reduce in the percentage of missing observations. The final data consist of 49,184 observations at each of 460,800 locations. Figure 3.1 shows levelplot of the log2 of rain rate average across all locations $0^\circ - 40^\circ N(S)$. For each location, the rain rate average is mean of the time series from 1998-07-01 00 to 2015-04-30 21 UTC, with missing observations ignored in cases when missing values are present.

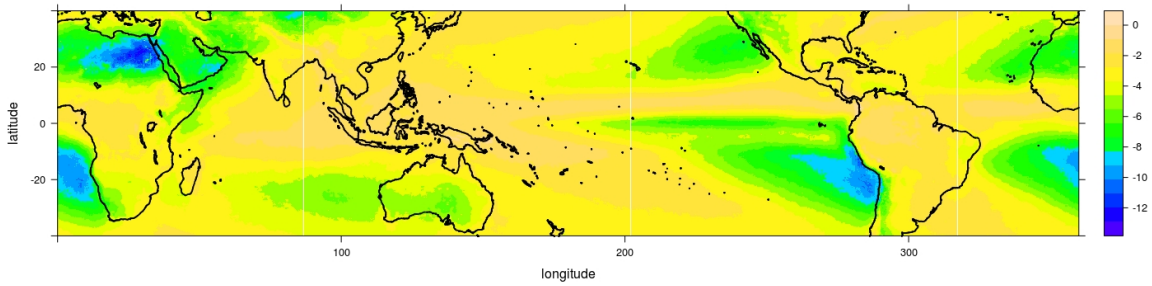


Fig. 3.1.: Levelplot of log2 of mean of rain rates over time

It is quite challenging to conduct data analysis on TRMM big data. First, the quality of data is a big concern, because rain rate is measured indirectly through multiple sensors flying on a variety of satellites. The TRMM Multi-satellite Precipitation Analysis (TMPA) [52] provides reasonable performance at monthly scales, although it is shown to have precipitation rate-dependent low bias due to lack of sensitivity to low precipitation rates over the ocean in one of the input products. In terms of shorter time scales such as daily scale and 3-hourly scale, the TMPA estimates demonstrate considerably more uncertainty.

Another challenge comes from a particular property of precipitation. The precipitation displays small-scale variability and highly non-normal statistical behavior that requires frequent, closely spaced observations for adequate representation. The precipitation pattern varies considerately from continents to oceans, from forests to deserts, and from mountains to flat lands. A mixed distribution with a point mass probability of zeros is often used to describe the frequent occurrence of rainfall zeros [47]. The spatio-temporal dependence in rainfall zeros is a critical aspect of any space-time stochastic model for precipitation.

3.3 Goal

From a statistical point of view, two objectives of data analysis on TRMM big data are: 1) Build an explanatory model to explain the variation of the response (3-hr rainfall occurrence); 2) Develop a predictive model for 3-hr rainfall occurrence. The explanatory modeling primarily focuses the goal of explaining the response with multiple explanatory variables. On the other hand, we define predictive modeling as the process of applying a statistical model or data mining algorithm to data for the purpose of predicting new or future observations. In particular, we focus on 3-hr rainfall occurrence prediction, where the goal is to predict rain probabilities in next 3-hr given its historical data.

The process of explanatory modeling is quite different from the one for predictive modeling. Galit Shmueli et al. [54] provide a thorough discussion of a variety of differences between explanatory and predictive modeling, from its sources and its purpose to the practical implications of the distinction at each step in the modeling process. From bias and variance perspective, explanatory modeling focuses on minimizing bias to obtain the most accurate representation of the underlying theory. In contrast, predictive modeling seeks to minimize the combination of bias and estimation variance, occasionally sacrificing theoretical accuracy for improved empirical precision. In classical inference, the explanatory model focuses on in-sample estimates by explained-variance metrics of the entire data sample, while predictive model focuses on out-of-sample estimates by assessing prediction performance metrics on unseen data samples which are not used during model fitting [55].

While explanatory power provides information about the strength of an underlying causal relationship, it does not imply its predictive power. One effect assesses to be statistically significant by a p-value may sometimes not yield successful predictability based on cross-validation, and vice versa. In Figure 3.2 [56], differences between 100 brain measurements (data points) drawn from each of two groups are evaluated using two-sample t-tests ("P-value") and classification ("Classification"), where data points

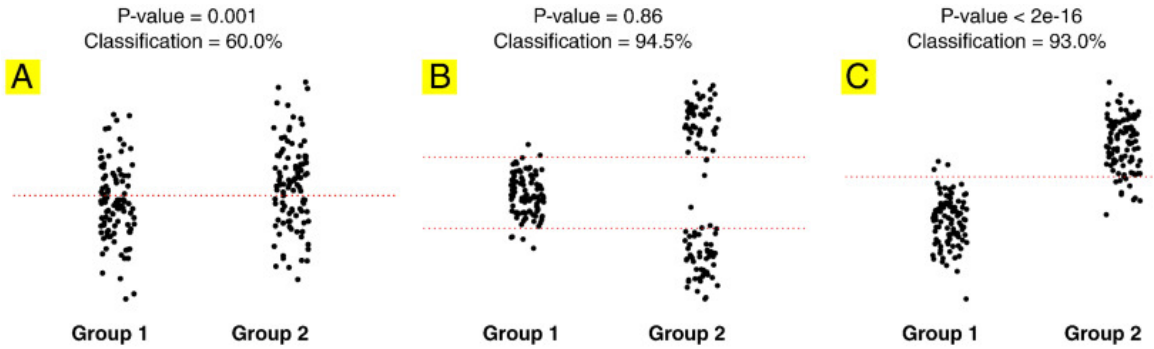


Fig. 3.2.: Examples for that classical statistical inference and classification performance can lead to diverging conclusions

on either side of the dotted lines are predicted as being from different groups. In three cases with different data distributions, (A) t-test was statistically significant, while classification accuracy was poor, (B) t-test was not statistically significant, while classification accuracy was high, (C) t-test was statistically significant and classification accuracy was high. This toy example illustrates that null-hypothesis rejection and pattern recognition constitute two different statistical analyses that do not necessarily judge data distributions by the same aspects. Hence, group effects as assessed by significant p-values do not always entail a high classification performance, and vice versa.

Before developing explanatory models, a natural question is which model best fits the data. The basic principles of model selection are 1) simple models have low variance, but risk bias; 2) More complicated models reduce bias and fit the sample data better, but can be highly variable and do not necessarily generalize to the population better; 3) Automatic model selection approaches and criteria can be informative, provided that we use the results cautiously and continue to think about the scientific meaning and plausibility of the models under consideration.

It should come as no surprise that many approaches have been proposed over the years for dealing with this key issue. Both frequentist and Bayesian statisticians have made great contributions on developing model selection methods including informa-

tion criterion (AIC and BIC), subset selection procedures (stepwise selection, best subset selection), shrinkage methods (Ridge and Lasso), cross-validation, goodness of fit tests (deviance goodness of fit test, Pearson chi-square goodness of fit test and Hosmer-Lemeshow test).

In explanatory modeling, model validation is to validate that the model fits the data $\{X, Y\}$. And the top priority in terms of model performance in explanatory modeling is assessing explanatory power, which measures the strength of relationship indicated by the model. Generally, R^2-type values can be used to indicate the level of explanatory power in linear regression as it indicates the proportion of variation of the response which is explained by the model.

Several pseudo R^2 measures for logistic regression are logically analogous to ordinary linear regression R^2 measures. There are many different ways to calculate R^2 for logistic regression and, unfortunately, no consensus on which one is best [57]. Mittlbock and Schemper [58] reviewed 12 different measures; Menard [59] considered several others. McFaddens R^2 is perhaps the most popular pseudo R^2 of them all. In the TRMM data analysis, we will develop explanatory models on all available data based on pseudo R^2 measures, and conduct model validation, model evaluation.

On the other hand, we do not really care how well the method works on the training data in predictive modeling. Rather, we are interested in the accuracy of the predictions that we obtain when we apply our method to unseen test data. As explanatory power does not imply its predictive power [54], R^2 measures are not appropriate for predictive modeling.

In predictive modeling, the biggest danger to generalization is overfitting the training data. Hence validation consists of evaluating the degree of overfitting, by comparing the performance of the model on the training and holdout sets. If performance is significantly better on the training set, overfitting is implied. Assessment of this performance is extremely important, since it guides the choice of learning method or model, and gives us a measure of the quality of the ultimately chosen model. Therefore, we divide the data set into two parts: training data and test data. First, we fit

candidate models to a set of the training data which consist of 3-hr rain rates from 1998 to 2013. Then we apply learning models on test data which include observations from 2014 to 2015 to check the predictive power of candidate models.

In terms of measures of predictive power, the most critical metric regards how well the model does in predicting the dependent variable on test observations. The fitted value for a logistic regression model is an estimate of the observation's class membership probability to which different thresholds may be applied to predict class membership. It might happen that model one is better than model two when one threshold is chosen, while model two is preferable if another threshold is selected. In order to compare the overall prediction performance of different models, we use the receiving operating characteristic (ROC). ROC [60] is a measure of classifier performance. Using the proportion of positive data points that are correctly considered as positive and the proportion of negative data points that are mistakenly considered as positive, we generate a graphic that shows the trade-off between the rate at which you can correctly predict something with the rate of incorrectly predicting something. Ultimately, we are concerned about the area under the ROC curve (AUC). This metric ranges from 0.50 to 1.00, and values above 0.80 indicate that the model does a good job in discriminating between the two categories which comprise our response variable.

3.4 Data Preparation

The raw TRMM data is a collection of NetCDF files. Each of file contains 3-hr rain rates of all locations and other metadata. The first step is to extract 3-hr rain rates from NetCDF and transfer them to HDFS as key-value pairs. Here, the key is the time and the value is a matrix of rain rates. We call this version as by-time division. As discussed in the previous section, our goal is to build models for 3-hr rainfall occurrence. It is necessary to generate a by-location division. Using RHIPE, the division by-location can be handily generated from the by-time division. For the

by-location division, there are 460,800 key-value pairs with the longitude and latitude of location as the key and a time series of rain rates as the corresponding value.

In this section, we will discuss two common data preparation operations: handling missing values and data sampling. To the best of my knowledge, the presence of missing values can reduce the data available to be analyzed, cause a significant bias in the results, and eventually influence the reliability of its results. There exist missing values in the TRMM data, thereby requiring one to determine the extent and type of missingness, and to choose a course of action accordingly.

3.4.1 Missing Values

The first task is to study the patterns of missing data before conducting data analysis. Considering by-time division and by-location division as a two-dimensional view point of data would give us a more comprehensive understanding of the TRMM data. Also, it is reasonalble to analyze the missing pattern in these two dimensions: by time and by location.

For the by-location division, we investigate the missingness from two aspects: missing ratios and missing runs. The missing ratio at a location is defined as the number of missing observations plus one divided by the total number of observations in the time-series of length 49,184 while the missing runs are the lengths of consecutive missing observations in the time-series.

Figure 3.3 graphically displays the log2 of the missing ratios across 460,800 locations. The ratios in the original scale are in the range $[2.033 \times 10^{-5}, 0.025]$ while their log scale (base 2) is in the range [-15.59, -5.321]. Generally speaking, the missing ratios are greatly influenced by the satellite's path.

Overall the missing ratios are small. However, it is possible that the max length of missing runs can be very large, which might cause problems when we conduct time-series analysis. For example, there will be around 490 missing observations if the missing ratio is 0.01. These missing observations might scatter sparsely in the time-

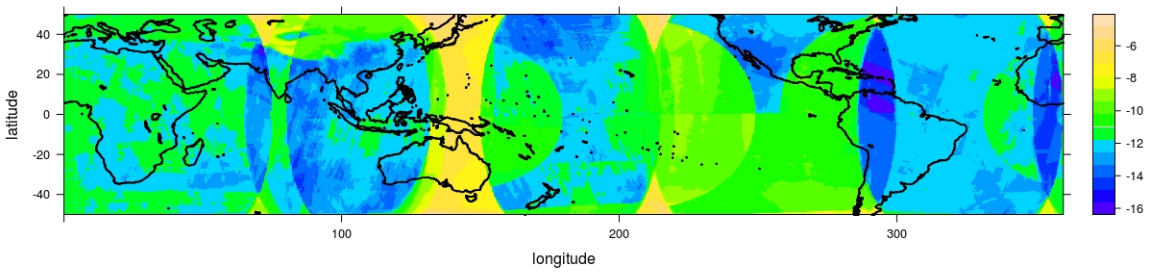


Fig. 3.3.: Levelplot of log2 of the missing ratios across all locations. The log2 of the missing ratios are represented by colors. The more blue, the smaller the missing ratio.

series of length 49,184 such as $[0.1, NA, 0.1, 0.3, NA, \cdots]$ or they are clustered together in a short time such as $[0.5, NA, NA, NA, \cdots, NA, 0.5, \cdots, 0.7, 1, NA, NA, NA, \cdots]$. To see the pattern of these missing runs graphically, we make the plot the max of the length of missing runs against longitude and latitude in Figure 3.4, where the max of the length of missing runs is defined as one plus the longest length of NA sub-series of the original time-series in each location. The large missing runs happen in the high latitude 36N-40N and longitude 137E-142E. And the largest missing run is 64, which means the longest consecutive unobserved days is 8 as there are eight observations per day.

Figure 3.3 and Figure 3.4 demonstrate the missing pattern in space. On the other hand, the missing behavior over time is illustrated in Figure 3.5 and 3.6. For the by-time division, there are $1440 \times 320 = 460800$ observations in each 3-hr timestamp, which starts at 1998-07-01 00:00 UTC. Every one point in Figure 3.5 indicates the log of the number of missing values plus one (log base 2) at the corresponding timestamp. It is clear that there are a large number of missing values between 21550 and 22550 timestamp, due to satellites upgrade between December 2005 and March 2006. In terms of the missing ratio, Figure 3.6 display the quantile plot of log2 of missing ratios for by-time division data. There are more than 99% of the time in which the missing

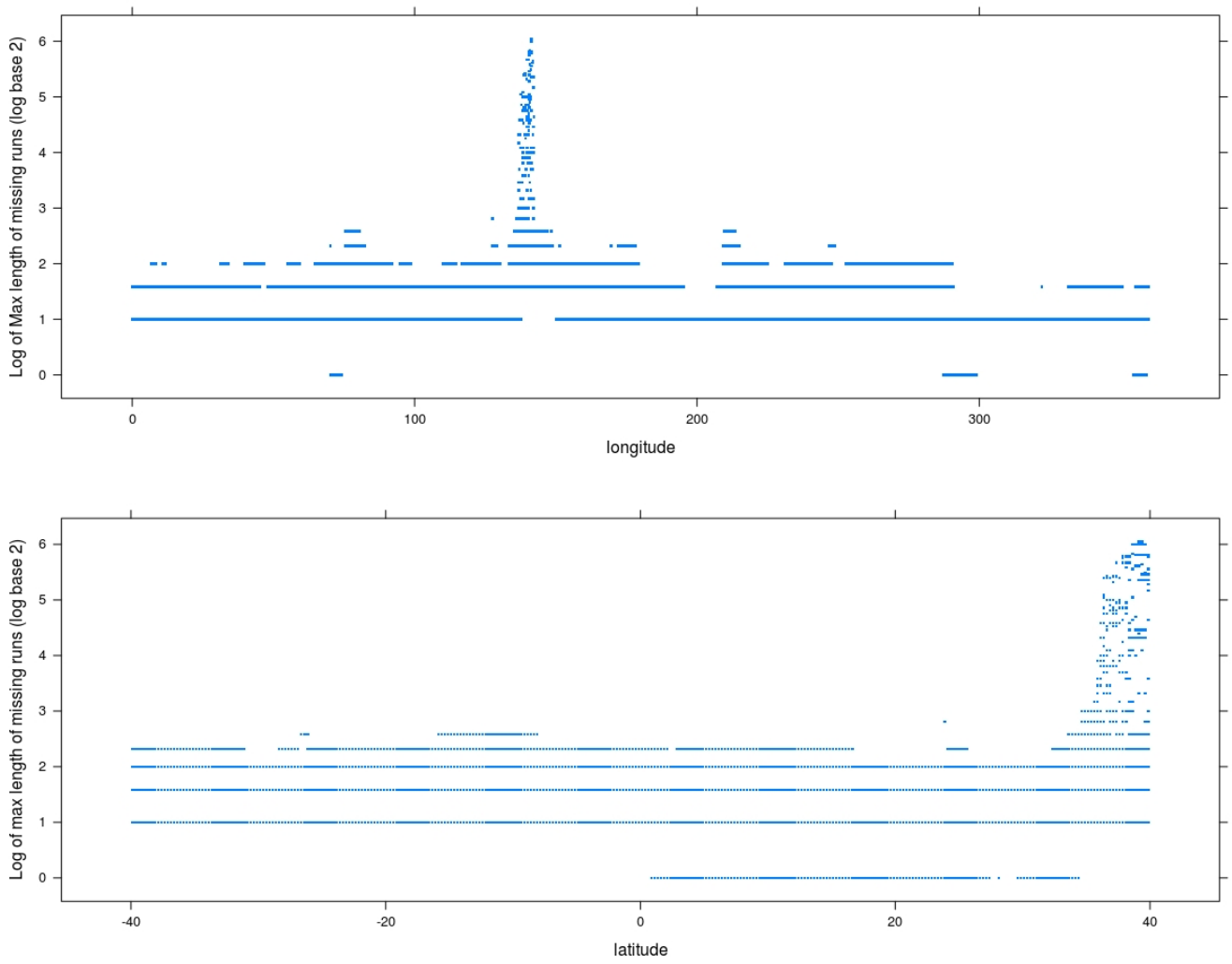


Fig. 3.4.: Scatter plot of log of the max length of missing runs against longitude (latitude)

ratio is less than 2.2%. And 2.5%, 50%, and 97.5% quantiles of missing ratios are 0, 0, and 0.7%, respectively.

Based on the analysis of missing pattern in both spatial and temporal dimensions, we can make a conclusion that missing ratios are quite small in most cases. Therefore, we can simply throw out those cases in the final model fitting stage, with a minor influence on the reliability of analysis results.

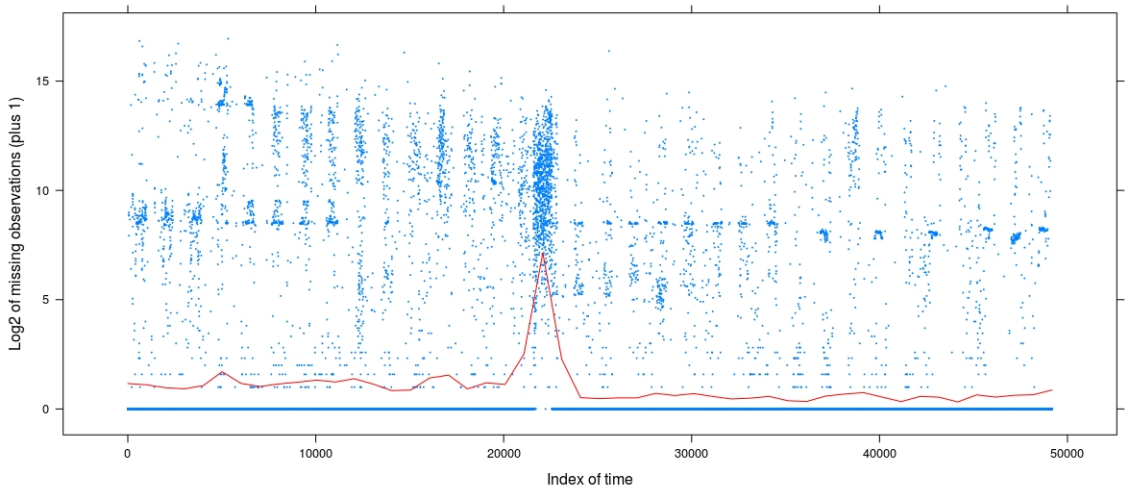


Fig. 3.5.: Plot of of log2 of the number of missing observations (+1) over time. A loess smooth curve with span 0.05 and degree 1 is displayed in red line.

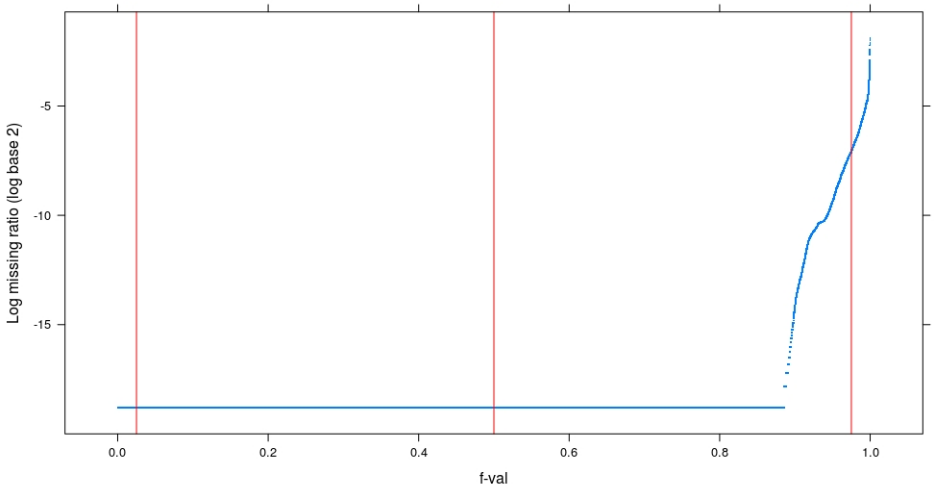


Fig. 3.6.: Quantile plot of log2 of the ratio of missing observations to total number observations for each timestamp. 2.5%, 50% and 97.5% quantiles are indicated by three red vertical lines, respectively.

3.4.2 Sampling

Nothing serves comprehensive analysis better than data visualization. This principle has been widely accepted and used for decades [61]. Visualization can be helpful in exploratory data analysis, model building, diagnosis. For a large and complex dataset, this requires making a large number of displays many of which can have a large number of pages and many panels per page. It will be overwhelmed if we make every diagnostic plot for models on big data. In order to conduct deep analysis in the model building procedure integrated with data visualization, it is necessary to obtain a representative sample of all locations. In past decades, a large number of sampling methods have been proposed such as simple random sampling (SRS), stratified sampling, cluster sampling and systematic sampling. As the TRMM data are spatio-temporal data, SRS and cluster sampling would lead to significant loss of original information. The downsample method, a specific case of stratified sampling and systematic sampling, is a reasonable and efficient sample method on the TRMM data.

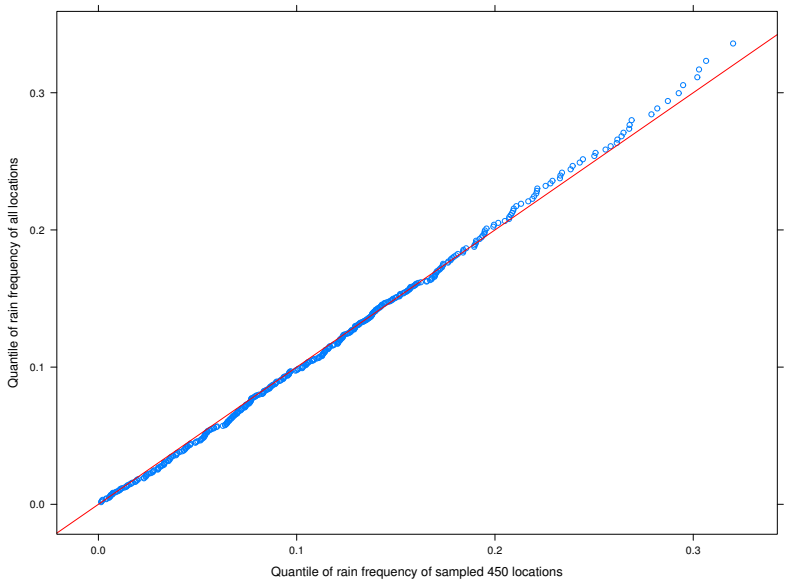


Fig. 3.7.: Quantile plot of rain frequency of sampled 450 locations against the one for all locations. The reference line $y = x$ is graphed by the red line

We sample 450 locations which consist of all combination pairs of 45 equally spaced longitudes and 10 equally spaced latitudes from 460,800 locations, resulting in $8° \times 8°$ latitude-longitude resolution. To check whether the sample is representative, we test whether the distribution of rain frequency of the whole population (all locations) is the same to the one for the subsample. Figure 3.7 graphs quantile of rain frequency of sampled 450 locations against the one for all 460,800 locations. The fact that the blue points are scattered along the straight line indicates that these two distributions are quite similar. Therefore, these 450 locations are good representative locations in terms of the rain frequency. Figure 3.8 shows the log2 of missing ratios at the sampled 450 locations.

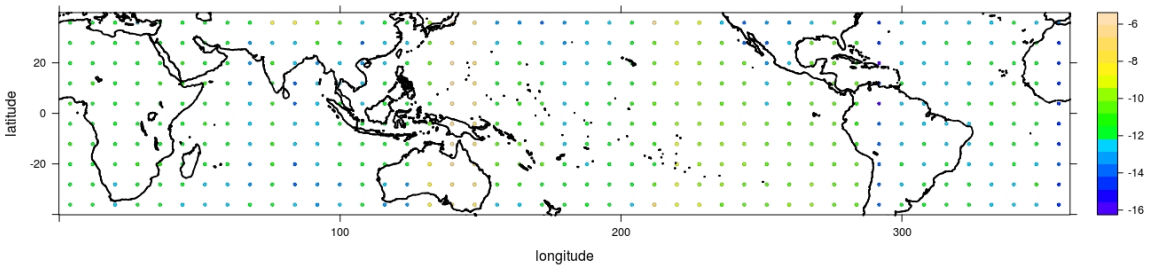


Fig. 3.8.: Levelplot of log2 of missing ratios on the sampled locations

In the following sections, all candidate models will be first applied to data of 450 locations and potential good candidate models will be chosen to be applied to all data after visual diagnostics and data validation.

3.5 Exploratory Data Analysis

Exploratory data analysis (EDA) is a critical first step in analyzing the data. EDA is for seeing what the data can tell us beyond the formal modeling or hypothesis testing task. More specifically, it can help us detect and describe patterns, trends, and relations in data with motivation from certain purposes of investigation. EDA makes

intensive use of data visualization, the basic objective of which is to provide an efficient graphical display for summarizing and reasoning about quantitative information.

As discussed before, the TRMM data have two dimensions: by time and by location. Ideally, we would like to investigate the evolution of spatial patterns in time, and distribution of temporal behaviors over space of the TRMM data, respectively. However, it will be overwhelmed by a huge amount of plots if we make plots for the data at 3-hr time scale for around 17 years, with $0.25° \times 0.25°$ latitude-longitude resolution. Therefore, we display spatio-temporal patterns for an aggregated version of the TRMM data. And seasonal behaviors are investigated in more detail by using STL+ model.

3.5.1 Spatio-temporal Patterns for Aggregated Data

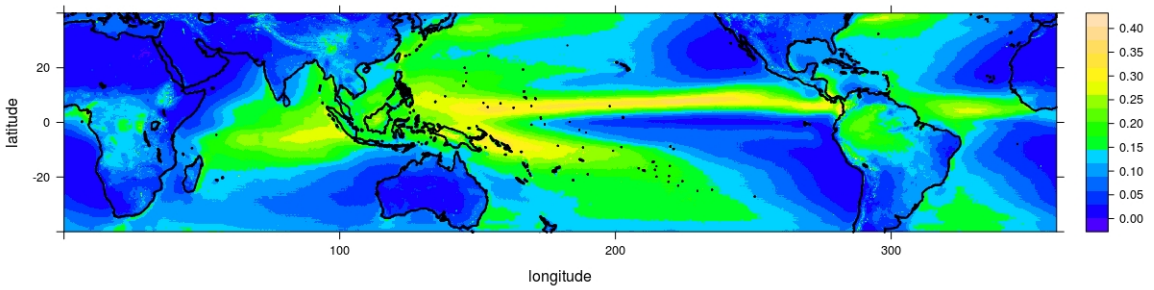


Fig. 3.9.: Levelplot of non-zero rainfall probability over time at each location

For the by-location division, the data is a rain rate time-series of length 49,184 at each location. The first aggregation method is to compute the probability of non-zero rainfall for each of 460,800 locations, with missing values removed. Figure 3.9 demonstrates the spatial patterns of non-zero rainfall probabilities. It is clear that

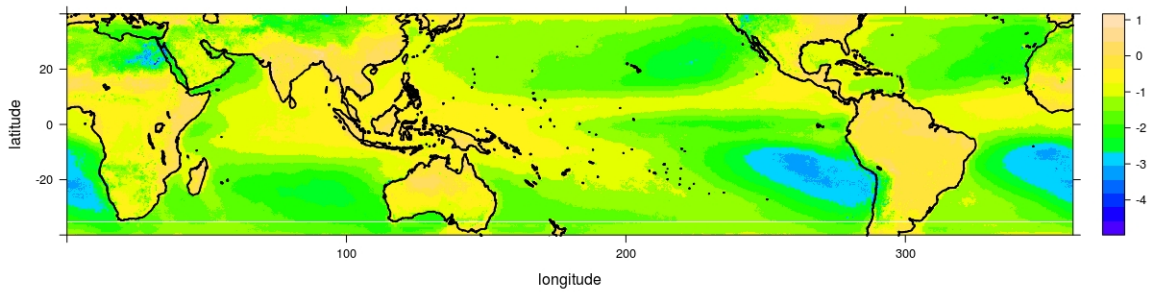


Fig. 3.10.: Levelplot of the mean of the log of positive rainfall at each location

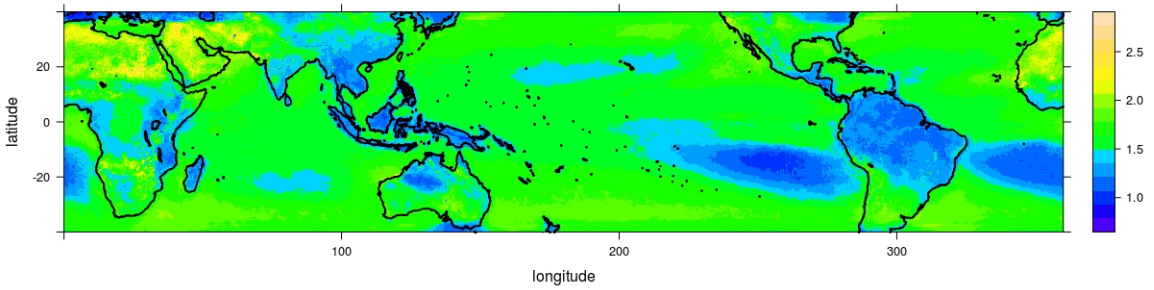


Fig. 3.11.: Levelplot of standard deviation of log of positive rainfall at each location

equatorial regions have a much higher frequency of rainfall than off-equatorial area. In general, the frequency of rainfall over west oceans is significantly larger than the one on the continents. Besides, tropical South America and tropical Africa are rainy regions on the continents.

Apart from the frequency of precipitation occurrence, rainfall intensity is, definitely, of great interest. The extremely variable nature of rain makes it difficult to compute time averages and higher moments of the rainfall amounts directly from the observational data. Experience shows that the probability density functions (PDFs) for positive rain rates are highly asymmetrical and skewed toward larger rain rates. Therefore, a Gaussian PDF is not appropriate in this case. There are many PDFs that are bounded on the left by zero and positively skewed. Among these distributions, the gamma distribution and lognormal are widely used to model rain rates. Cho et al. [62] conducts a comparison of Gamma and lognormal distributions for characterizing satellite rain rates from the TRMM 3A26 data. This comparison indicates that the Gamma fits outperform the lognormal fits in wet regions, whereas the lognormal fits are better than the Gamma fits for dry regions. Due to that most of continents

are dry regions, the lognormal distribution is used to characterize positive rain rates for the visualization purpose.

Figure 3.10 displays geographical patterns of the mean of log-transformed positive rain rates. It indicates that rainfall intensity is relatively high in equatorial regions, South Africa, Australian, South America, east of North America, and south of Asia. East of South Atlantic Ocean and South Pacific Ocean, and North Africa have low rainfall intensity. To see the variability of log-transformed positive rain rates, we can refer to Figure 3.11 for more detail. Collectively, Figure 3.9, 3.10 and 3.11 provide the basic statistical characteristics of the TRMM data in space. For example, southeast of Pacific Ocean has relatively low rainfall frequency, low rainfall intensity if it rains, and low variability of rainfall based on 3.9, 3.10 and 3.11, respectively.

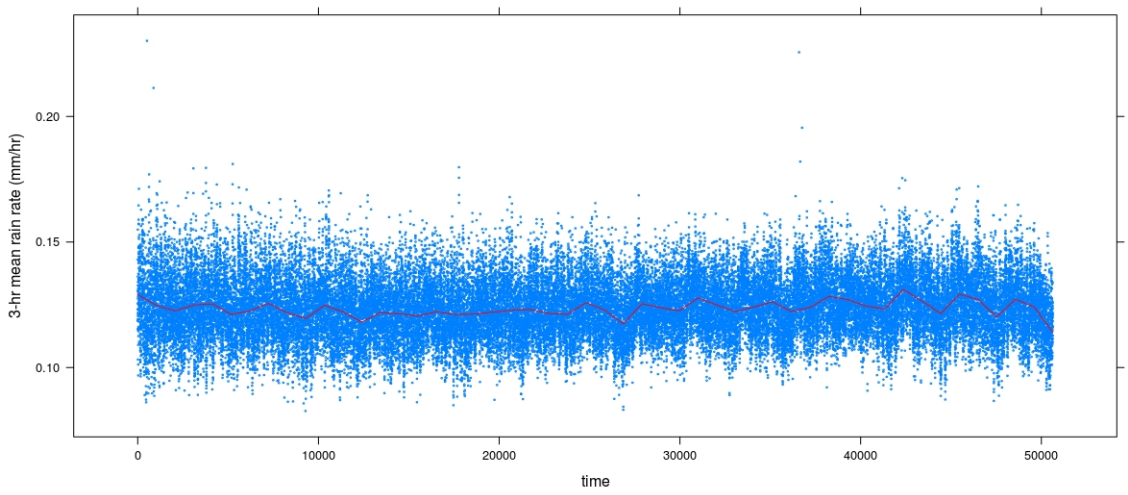


Fig. 3.12.: Time series plot of 3-hr mean rain rate

To explore temporal behaviors of the TRMM data, we aggregate 3-hr rain rates over space. For each timestamp, 3-hr mean rain rate is calculated by averaging 3-hr rain rates over 460,800 locations. Collectively, all 49,184 mean rain rates form a time series at a time scale of 3-hr shown in Figure 3.12. Obviously, there exists seasonal pattern in the mean rain rates. Furthermore, a monthly mean rain rate time series is obtained by averaging 3-hr mean rain rates over each month. Finally, we can easily

generate a time series of yearly rain rates from the monthly mean rain rate time series through averaging monthly rain rates over each year. For yearly rain rates, we only consider years in which all monthly rain rates are available as the average over a partial year can result in a biased estimate.

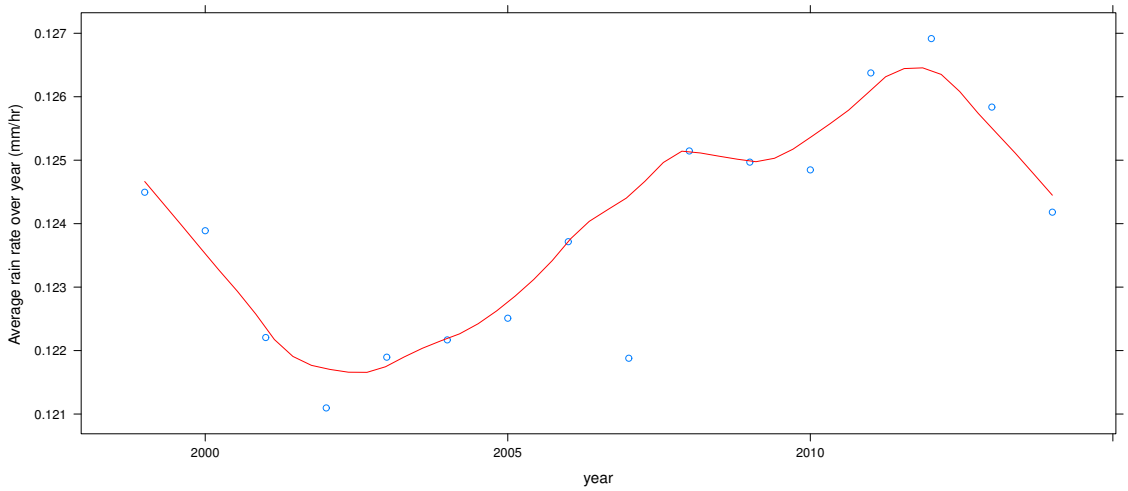


Fig. 3.13.: Time series plot of yearly rain rate

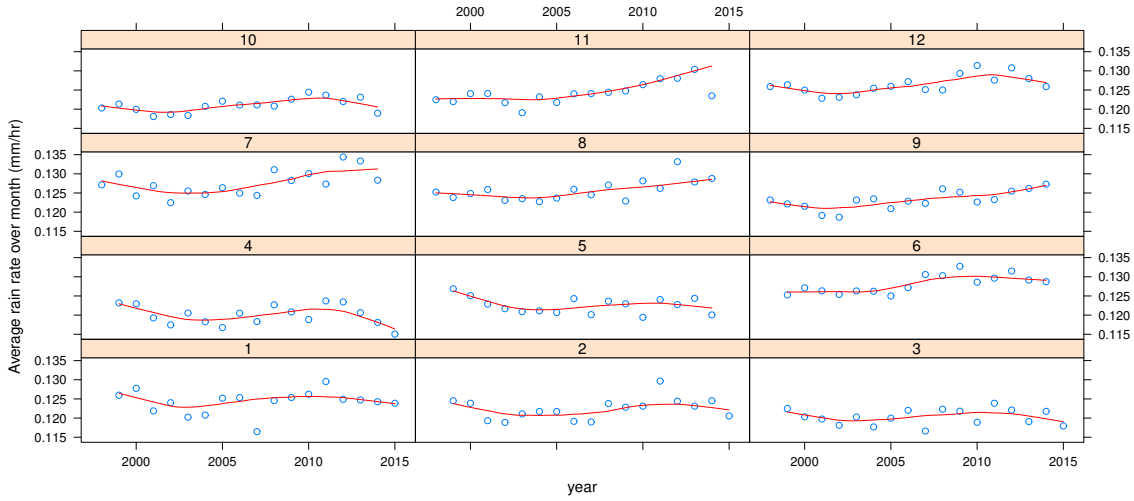


Fig. 3.14.: Plot of monthly rain rate against year conditional on month

Figure 3.13 plots yearly rain rates from 1999 to 2014, superimposed by the red loess curve with degree=1, span=1/3. Based this aggregation method, yearly rain rates vary across years in a shape similar to a sin trigonometric curve. In order to explore seasonal patterns, we plot monthly rain rate against year conditional on month in Figure 3.14. For each month panel, blue points are the monthly mean rain rates at the corresponding year while the red curve is the loess curve fitted with degree=1, span=0.5 on blue points. Each monthly sub-series goes down first and goes up later as year increases, which is consistent with the yearly rainfall pattern in Figure 3.13. Due to the same scale in all panels in Figure 3.14, it is significant that there is an increase trend from April to June and from October to December.

Collectively, there are some seasonal patterns in aggregated data. We expect a variety of seasonal behaviors for different locations. As the Earth travels around the Sun, the area of sunlight in each hemisphere changes. At a solstice, the area of sunlight is at a maximum in one hemisphere and a minimum in the other hemisphere. In the next subsection, seasonal patterns will be discussed in more detail for representative sampled locations.

3.5.2 Seasonal Behavior

Let $R_{s,t}$ be the observed precipitation rate on the t-th period at site s. The index t represents the index of 3-hr interval if it is the 3-hourly data, the index of the month if it is the monthly data. The next step of exploratory data analysis is to investigate yearly seasonal behaviors of the data. we will model monthly data using Seasonal Trend Decomposition using Loess (STL), to explore potential seasonality and long-term trend across all locations.

As discussed in chapter one, STL is a filtering procedure for decomposing a seasonal time series into three components: trend, seasonal, and remainder. Suppose the data, the trend component, the seasonal component, and the remainder component

are denoted by Y_i, T_i, S_i and R_i, respectively, for i = 1 to N. Here N is the total number of observations. Then

$$Y_i = T_i + S_i + R_i.$$

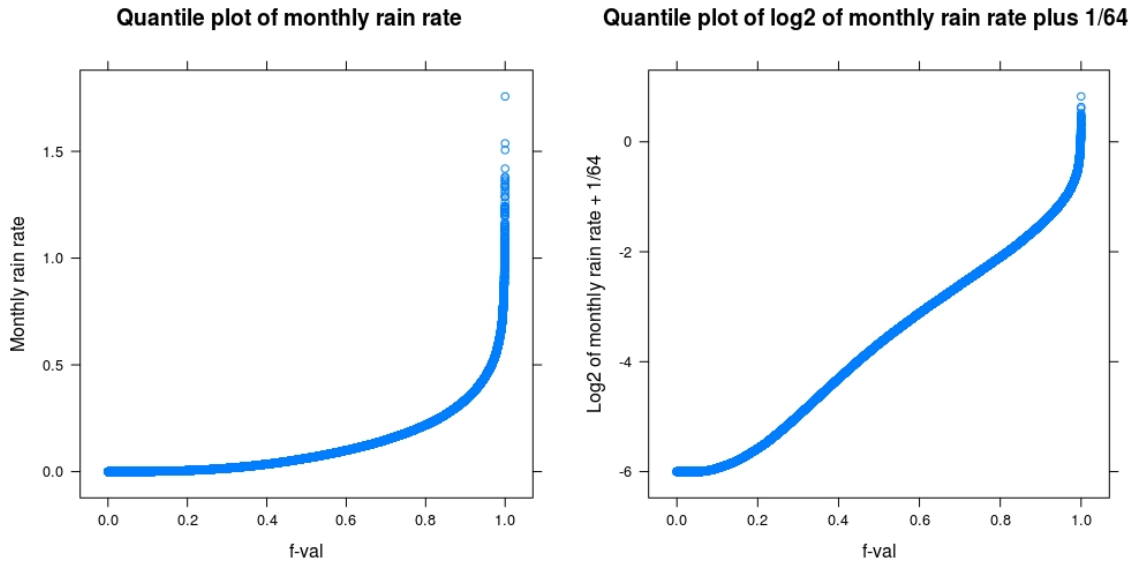


Fig. 3.15.: Quantile plot of monthly rain rate and it's log transformation

Before applying a STL model on monthly rain rate data, it is essential to know what the data distribution looks like. We define the monthly rain rate as the average of 3-hour rain rates over each month at given year. The monthly rain rates across all 460,800 locations are in the range [0, 4.23]. The zero rain rate indicates that it does not rain for the whole month in the corresponding locations, most likely in the desert regions. To look at the distribution of the monthly rain rate, we made a uniform quantile plot of monthly rain rates for all locations shown in the left panel of Figure 3.15. This plot indicates that the monthly rain rates are highly right-skewed. In contrast, the log transformation is applied to the monthly data plus a positive constant and its corresponding uniform quantile plot is displayed in the right panel of Figure 3.15. This plot implies that the distribution of the log-transformed monthly rain rates becomes quite close to the uniform distribution, except it has heavier tails.

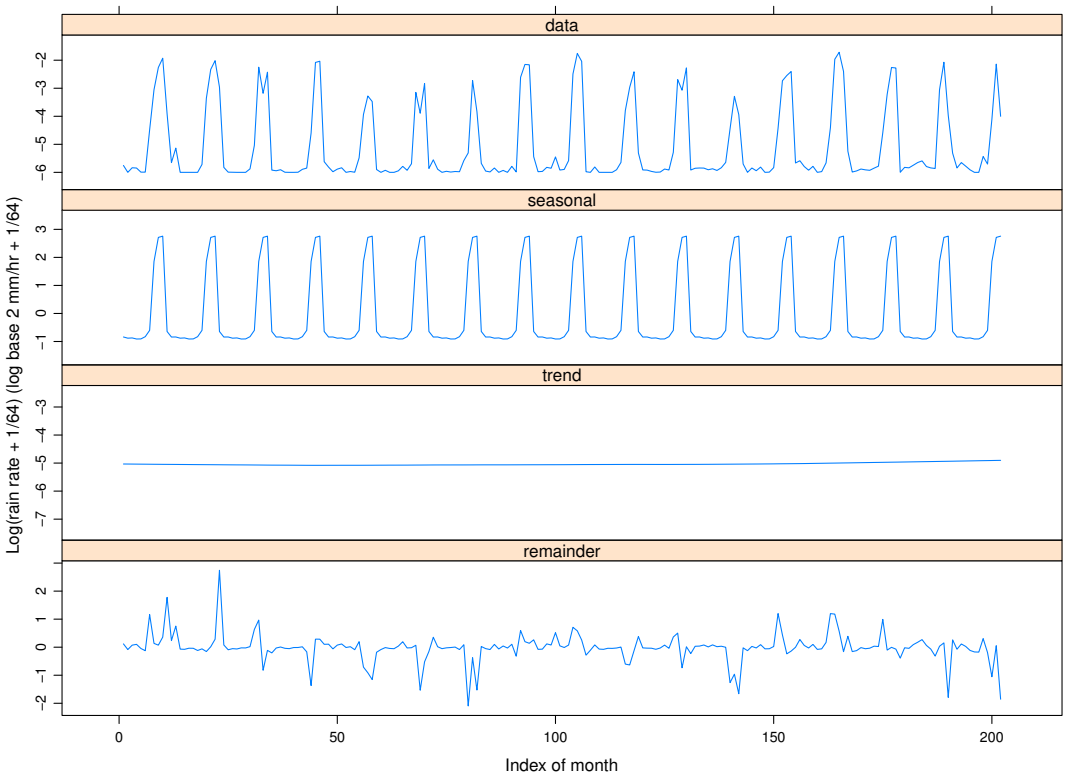


Fig. 3.16.: Decomposition plot of log-transformed monthly rain rate at location (4.125°S, 92.125°W)

Now, we apply a STL+ model on the log-transformed monthly rainfall data for 450 sampled locations with $t_{window} = 84, t_{degree} = 1, s_{window} = periodic, inner = 10, outer = 10$. For each location, there are 202 monthly rain rates from July 1998 to April 2015. The time series is split into 12 cycle-subseries, each of which is defined to be the subseries at each time point of the seasonal cycle. For example, all the observations of January will be the first subseries. This STL+ fit has a special case. $s_{window} = periodic$ makes the seasonal component strictly periodic, that is, each seasonal subseries is constant through time. Figure 3.16 demonstrates the decomposition plot of log-transformed monthly rain rate at location (4.125°S, 92.125°W) for this STL+ model.

The top panel shows the log transformation of monthly rain rate plus 1/64 (response) against the index of the month. The second panel graphs the corresponding

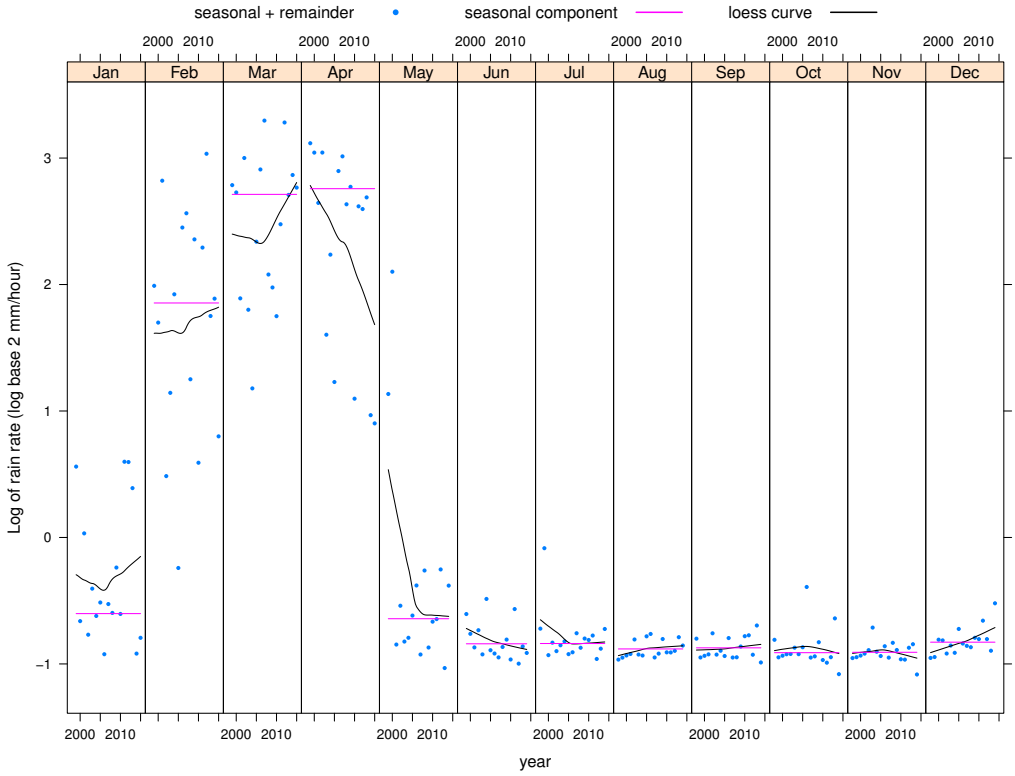


Fig. 3.17.: Seasonal diagnostic plot for log-transformed monthly rain rate at location (4.125°S, 92.125°W)

decomposed seasonal component: variation in the data at or near the seasonal frequency, which is one cycle per year in the monthly data. The third panel plots a trend component: the low frequency variation in the data. While the remainder component shown in the bottom panel is the remaining variation beyond that explained in the seasonal and trend component. Scale for each series has the same number of units per centimeter, which enables the variability of each series to be compared. Comparing these four time-series, we can make a conclusion that most of the variability of the data can be explained by the seasonal component while the trend component can barely explain the variability of the data. Is this an appropriate STL model to decompose this log-transformed monthly data? What can we do next if not? To answer these question, we can resort to diagnostic plots.

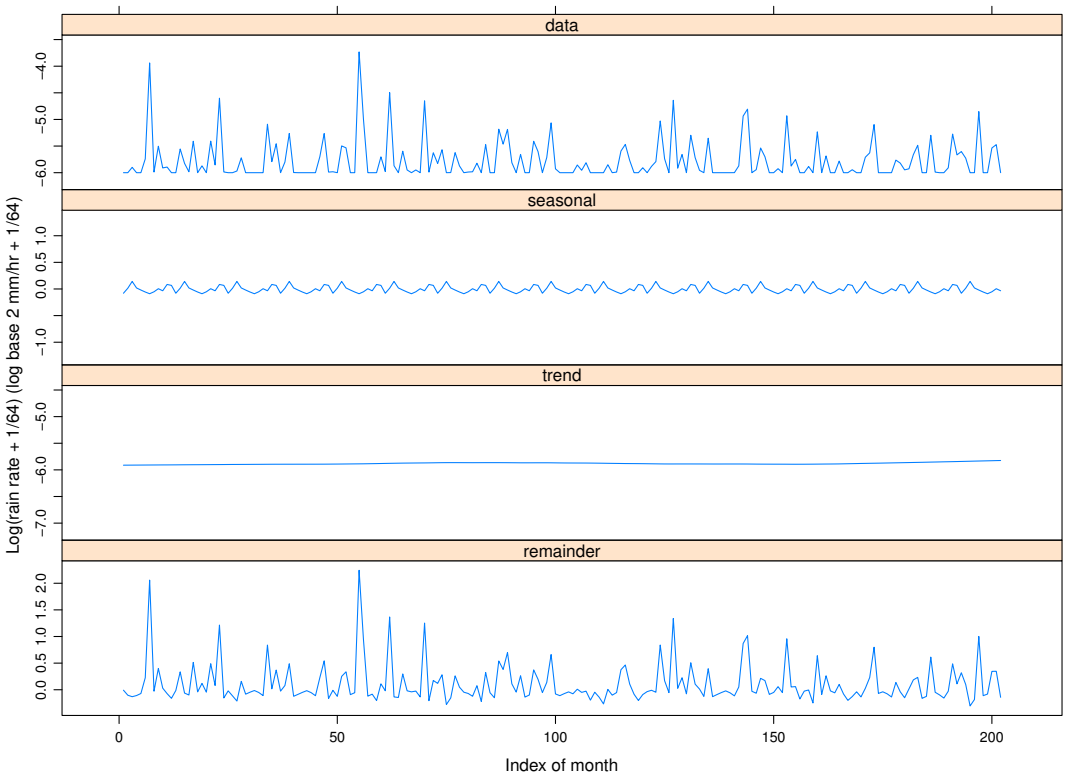


Fig. 3.18.: Decomposition plot of log-transformed monthly rain rate at location (27.875°N, 3.875°E)

Figure 3.17 is the seasonal diagnostic plot for log-transformed monthly data at location (4.125°S, 92.125°W). Each cycle-subseries is graphed separately against year. First, the January values are plotted, then the February values are graphed, and so forth. The midmean of the values is portrayed by the red horizontal line, namely seasonal component. The seasonal component plus the remainder component, the data with the trend component removed, is plotted against year, displayed by the blue dots. The black smooth curve fitted by loess is superposed on them. The loess smoothing line here can be helpful to judge the lack of fit for the seasonal component. The plot shows that the seasonal component is able to capture the trend of the data. Clearly, there is not any lack of fit problem left in the remainder for June-December sub-series since loess smoothing line is all around horizontal red line. We still can not make a conclusion for January-April sub-series even through red lines for January-

April sub-series are greatly different from the black loess curves in that the variation of these sub-series are quite large and there are not enough monthly data to help us make a judgment.

Link to figure

Fig. 3.19.: Time series plot of data, seasonal, trend and remainder for 450 locations

Based on Figure 3.16 and 3.17, we see the strong yearly seasonality in the monthly rainfall at location (4.125°S, 92.125°W). However, the characteristics of seasonality vary dramatically among different locations even with the same smoothing parameters. Therefore, we can not extend the conclusion to all locations. Figure 3.18 displays the STL decomposition on the data at the location (27.875°N, 3.875°E). By comparing data, seasonal, trend, and remainder time-series, neither seasonal component nor trend component can greatly explain the variation in the data. More figures for the STL decomposition on the representative sampled locations can be found in the link at Figure 3.19.

what is the overall performance of STL+ model on log-transformed monthly data across all 460,800 locations? The seasonal amplitude, which is defined as the difference between the maximum and minimum in the seasonal series, can be used for the measurement of the variation of the data explained by the seasonal component. Similarly, the trend magnitude, which is defined as the difference between the maximum and minimum in the trend series, is the measurement of the variation of the data explained by the trend component. For each location, a STL+ model with the same tuning parameters is fitted to log-transformed monthly rain rates. Then the seasonal amplitude and trend magnitude are computed. Collectively, there are 460,800 amplitudes and magnitudes in total for all locations. Figure 3.20 demonstrates the quantile plot of seasonal amplitude and trend magnitude for all locations and shows that the

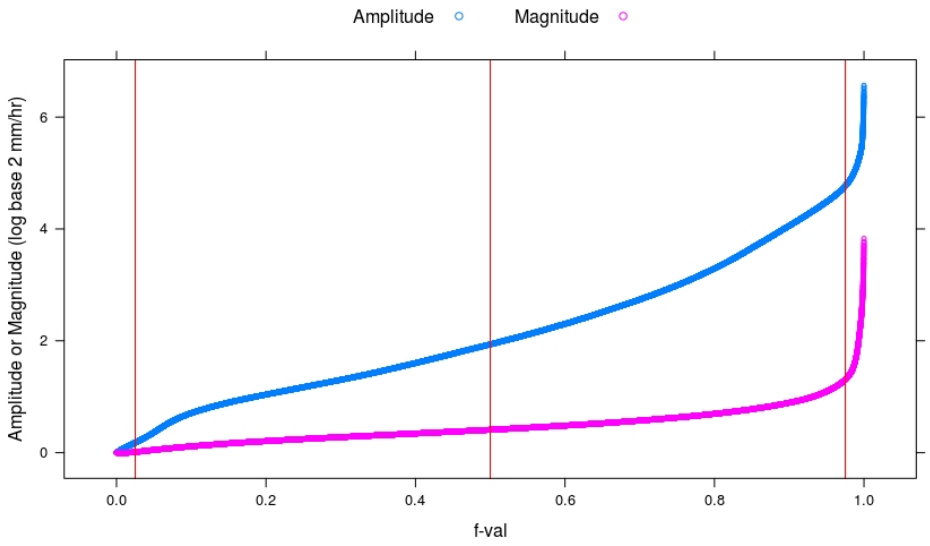


Fig. 3.20.: Quantile plot of seasonal amplitude and trend magnitude for all locations

seasonal amplitude is greatly larger than the trend magnitude, which means a larger portion of variation in the data can be explained by the seasonal component overall.

3.5.3 Spatial Correlation

Assume we apply a STL+ model on the log-transformed monthly rainfall data. A portion of variation in the data can be explained by the seasonal component overall. Generally speaking, a relatively large portion of variation in the data remains in the remainder. It is worthwhile to investigate whether the variation left in the remainder can be explained through some spatial features. For any location in the TRMM data, there are four closest locations in left, right, upper and lower direction, respectively. We call these four locations as spatial neighbors of a center location, displayed in Figure 3.21.

The distances between the center location and its neighborhood locations are quite close to each other. A reasonable and realistic distance in spatial dimension is the Great-circle distance, which is the shortest distance between two points on the surface of a sphere. More specifically, let ϕ_1, λ_1 and ϕ_2, λ_2 be the geographical latitude and

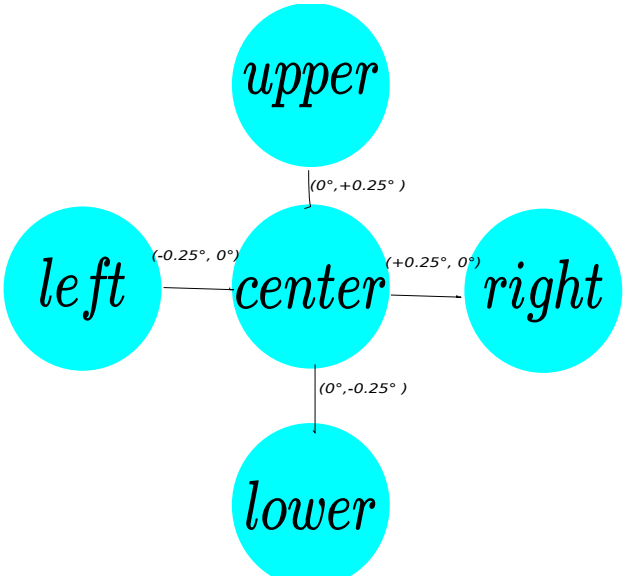


Fig. 3.21.: Spatial neighbors of a center location

longitude of two points a and b, and $\triangle_\phi, \triangle_\lambda$ their absolute differences. Then $\triangle_\delta$, the central angle between them, is given by:

$$\triangle_\delta = 2 \arcsin \sqrt{\sin^2(\triangle_\phi/2) + \cos(\phi_1)\cos(\phi_2)\sin^2(\triangle_\lambda/2)}$$

Then the distance between these two points, i.e. the arc length, for a sphere of radius r is

$$d(a,b) = 2\pi r \frac{\triangle_\delta}{360}$$

As the difference in latitude and longitude is either (0°, 0.25°) or (0.25°, 0°) for the neighbor locations, $\triangle_\delta = 2\arcsin(\cos(\phi)\sin(0.125))$ or 0.25. In the TRMM data, the max ϕ is 40, so the min value of $\triangle_\delta$ is around 0.19. The distance between the center location and its neighborhoods is in the range between 13.2 and 17.4 miles.

In applied statistics, a partial residual plot is widely used to show the relationship between a given independent variable and the response variable, given that other independent variables are also in the model. Similarly, we can make a remainder plot to investigate the relationship of the log-transformed monthly rainfall between the

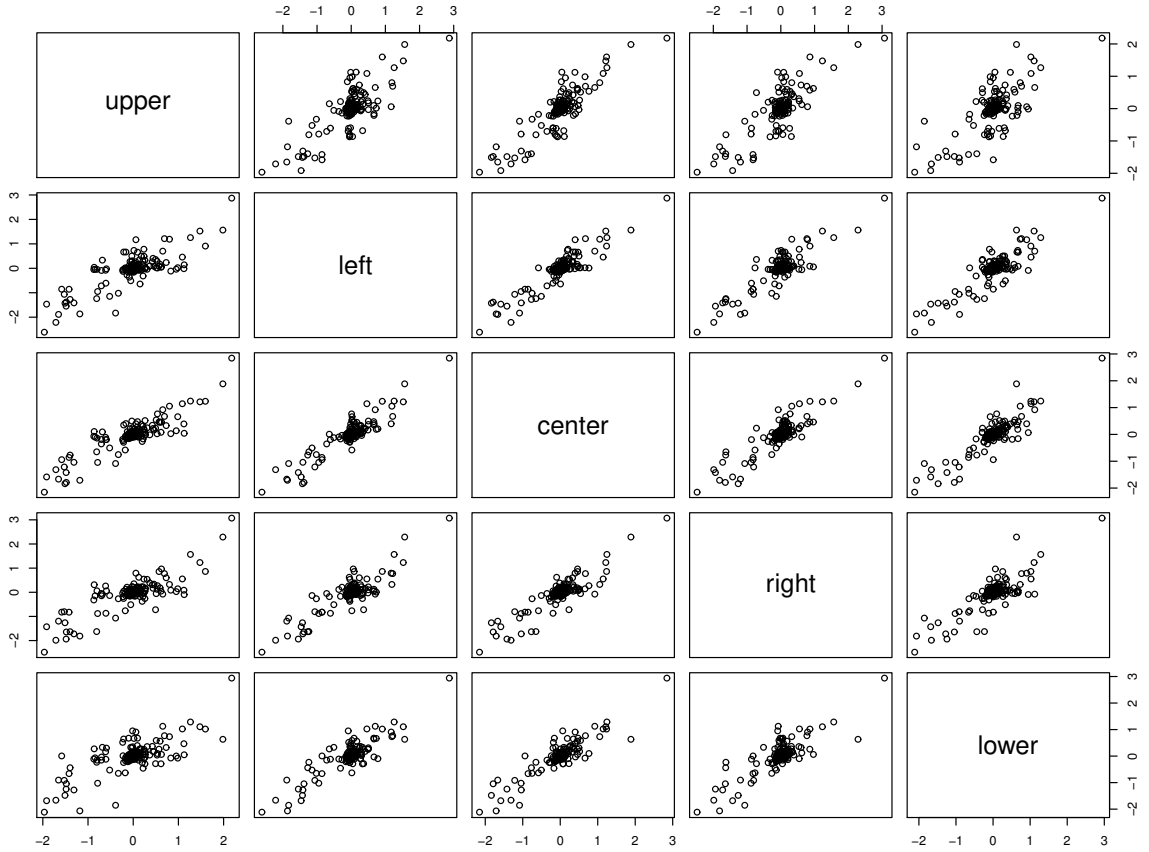


Fig. 3.22.: Scatter matrix of remainders decomposed from STL+ model on log-transformed monthly rain rates at center location (4.125°S, 92.125°W)) and its neighborhoods

center location and its neighborhoods, given that the seasonal variables are included in the model.

Figure 3.22 is a scatter plot of the remainder decomposed from STL+ model on log-transformed monthly rain rates at the center location (4.125°S, 92.125°W)) against ones at its neighborhood locations. A significant correlation between remainders indicates that spatial features should be included in the model to further explain the variation of data.

3.6 Explanatory Modeling

Rainfall exhibits extensive variability on a wide range of spatial and temporal scales, and the data correlation in space and time is unknown. In this section, we will build explanatory models for 3-hr rainfall occurrence with the joint use of spatial and temporal features based on EDA. The top priority in terms of model performance in explanatory modeling is assessing explanatory power, which measures the strength of relationship indicated by a model function.

In terms of explanatory power, one of the most popular methods is McFadden R^2 for Logistic regression. Logistic regression is estimated by maximizing the likelihood function. Let L_0 be the value of the likelihood function for a model with no predictors, and let L_M be the likelihood of the model being estimated. McFaddens R^2 is defined as

$$R^2_{McF} = 1 - \log(L_M)/\log(L_0)$$

where $\log(\cdot)$ is the natural logarithm.

To understand whether this definition makes sense, suppose first that the covariates in our current model give no explanatory information about the outcome. For individual binary data, the likelihood contribution of each observation is between 0 and 1 (a probability), and so the log likelihood contribution is negative. If the model has no explanatory ability, the likelihood value for the current model will not be much greater than the likelihood of the null model even though it is always larger. Therefore the ratio of the log-likelihood of the current model to one for the null model will be close to 1, and McFaddens R^2 will be close to zero, as we would expect.

Next, suppose our current model explains virtually all of the variation in the outcome Y. How would this happen? As the logistic regression model's purpose is to give a prediction for $P(Y = 1)$ for each observation, we would need $P(Y = 1) \approx 1$ for those observations who did have $Y = 1$, and $P(Y = 1) \approx 0$ for those observations who had $Y = 0$. If this is the case, the probability of seeing $Y = 1$ is almost 1 when $P(Y = 1) \approx 1$, and similarly, the probability of seeing $Y = 0$ is almost 1 when

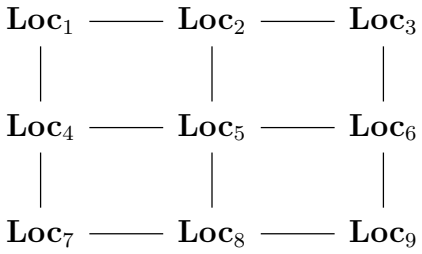
$P(Y = 1) \approx 0$. This means that the likelihood value for each observation is close to 1. As the log of 1 is 0, the log-likelihood value will be close to 0. Then McFaddens R^2 will be close to 1.

3.6.1 Spatio-temporal Logistic Model

Suppose $R_{s,t}$ is the observed precipitation rate on the t-th period at the location s. Here, the index t stands for the index of 3-hr interval. Then we define the precipitation occurrence $Y_{s,t}$ on the t-th period at location s as follows:

$$Y_{s,t} = \begin{cases} 0 & \text{if } R_{s,t} = 0 \\ 1 & \text{if } R_{s,t} > 0. \end{cases}$$

Based on the exploratory data analysis in the previous section, spatial features are helpful in variation explanation of monthly rainfall. Intuitively, spatial features are expected to be good explanatory variables for 3-hr rainfall data as well. To make notations simple, we define the neighborhood relationship as follows:



Where $\mathbf{Loc}_5$ is the center location in which the rainfall status (rain or no-rain) is considered as the response variable in our models while other locations are the neighborhood of the center location. The rainfall status at neighborhood locations are constructed as explanatory variables. We assume that whether it rains or not at the center location is correlated with whether it rains or not at its neighborhoods. Intuitively, the closer to the center location the neighborhood is, the higher the correlation is. Therefore, we classify these 8 neighborhood locations into two layers. The first layer includes $\mathbf{Loc}_2$, $\mathbf{Loc}_4$, $\mathbf{Loc}_6$, and $\mathbf{Loc}_8$. The second layer consists of $\mathbf{Loc}_1$, $\mathbf{Loc}_3$, $\mathbf{Loc}_7$, and $\mathbf{Loc}_9$.

Combining spatial correlation with seasonal behaviors found in exploratory data analysis, we propose the following spatial-temporal (ST) logistic model:

- ST Model 1: $logit(p(Y_{s,t} = 1)) = month\beta_m^s + year\beta_y^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + Loc_1\beta_1^s + \cdots + Loc_4\beta_4^s + Loc_6\beta_6^s + \cdots + Loc_9\beta_9^s$

where the $\mathbf{Loc}_i$ indicates the rainfall status (1: rain; 0: no-rain) on the t-th period at the i-th neighborhood location, $i \neq 5$. And the $\beta_i^s, i \neq 5$ is the corresponding neighborhood variable coefficient. In the model, month, hour, and lag_k are the categorial variables. Due to yearly seasonal patterns discovered in EDA, month variable is included in the model. In total, there are 12 levels for the month factor and β_m^s is the month coefficient vector. Hour factor is included due to the diurnal rainfall cycle in some regions [63] such as Indochina peninsula. There are 8 levels of the hour factor as there are 8 observations per day for 3-hourly data. Here β_h^s is the coefficient vector for hour variable. lag_k indicates whether it rains at time t-k at the center location ($\mathbf{Loc}_5$). In majority of sampled locations, the rain rates have an autocorrelation function that has a geometric decay as the lag increases and have a partial autocorrelation function which has a significant cutoff at 2. This suggests that it is appropriate to add lag_1 and lag_2 in the model. Finally, the year is a numeric variable, which can be used to explain the trend. $\beta_y^s, \beta_{l1}^s, \beta_{l2}^s$ are the coefficient parameters for year, lag_1 and lag_2, respectively.

In the ST model 1, the categorical values of month would cause a gap in two consecutive days, for example, May 31 and June 1, which are supposed to have similar yearly seasonal behavior. This motivates us to include day-of-year seasonality in the model in order to maintain the continuity of probability of precipitation occurrence over days. Therefore, we model precipitation occurrence by logistic regression on a series of harmonics to include seasonality in addition to the neighborhood precipitation occurrence. Specifically, within each season of a given year, the precipitation occurrence $Y_{(s,t)}$ is fitted using logistic regression accounting for the location-dependency and the day-of-year seasonality. The updated model is

- ST Model 2: $logit(p(Y_{s,t} = 1)) = year\beta_y^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + \sum_{i=1}^{13}\{\beta_{si}^s \sin(2\pi\frac{day\times 26}{365}\times\frac{i}{26}) + \beta_{ci}^s \cos(2\pi\frac{day\times 26}{365}\times\frac{i}{26})\} + Loc_1\beta_1^s + \cdots + Loc_4\beta_4^s + Loc_6\beta_6^s + \cdots + Loc_9\beta_9^s$

Where day is the index of day of year, which is in the range between 1 and 365. And β_{si}^s and β_{ci}^s are corresponding coefficient parameters for Sine and Consine series.

3.6.2 Model Selection

In explanatory modeling, the candidate models are compared according to the explanatory power. Stepwise regression procedure iteratively tries to remove predictor variables from the model in an attempt to delete variables that do not significantly add to the fit. Stepwise-type methods might appear suitable for achieving high explanatory power.

Table 3.2.: Model selection summary

Variable	year	hour	lag_1	lag_2	season	neighborhoods
Proportion of locations	37.3%	40.4%	40.0%	27.6%	37.0%	100%

We apply stepwise model selection procedure to data at all 460,800 locations in parallel; and choose a final model based on AIC for each location. Table 3.2 is the summary of model selection results from the full model – ST model 2. Here the "season" is defined by a set of $\sin(2\pi\frac{day\times 26}{365}\times\frac{i}{26}), \cos(2\pi\frac{day\times 26}{365}\times\frac{i}{26}), i = 1, \cdots, 13$. And neighborhoods consist of a set of neighborhood locations $\{Loc_i, i \neq 5\}$. Proportion of locations is the ratio of the number of locations in which the variable is included in the final selected model, to the total number of locations. For example, if anyone of seasonal features is included in the final selected model at one location, we will count 1; otherwise, we count 0. Then we obtain the number of locations for "season" by summing these counts over locations, and the proportion is this number divided by 460,800. It is interesting to note that the stepwise model selection schema keeps spa-

tial features in the final model for all locations. However, there are only 37% locations which keep season features in the final model. One explanation is that the rainfall status at neighborhood locations, in fact, already include yearly season information to some extent, because rain rates at the center location and neighborhood locations are collected at the same time.

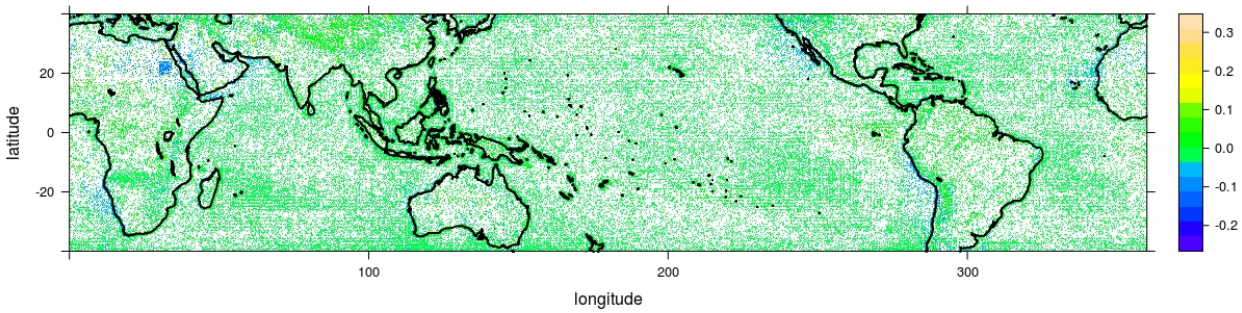


Fig. 3.23.: Levelplot of the coefficient of year in the final selected model using stepwise model selection procedure on all locations

There are 37.3% of locations including "year" in the final selected model. Figure 3.23 displays heatmap of the coefficient of year at locations where year is selected in the final model. As shown in this Figure, the coefficient is negative along west coast of the United States, Chile and Peru, Egypt, Sudan, Western Sahara and Namibia. This implies that the expected change in log odds between rain and no-rain in these regions is the value of coefficient of year as it increases one year. In other words, odds of rainfall becomes smaller and smaller as time goes in these regions.

To see explanatory power, we make a levelplot in Figure 3.24 to display the spatial patterns of the explanatory power of the final selected models for all locations. The value in each pixel represents McFaddens R^2 of the selected model fitted on 3-hr rain rates at the corresponding latitude-longitude location. 75% of final models selected by the stepwise model selection procedure can achieve McFaddens R^2 at least 0.7. The

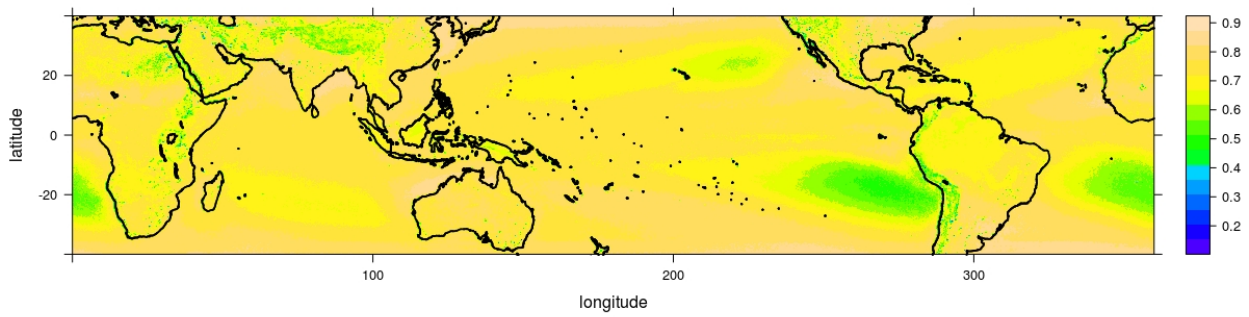


Fig. 3.24.: Levelplot of McFaddens R^2 for the final selected model using stepwise model selection procedure on all locations

representation of both temporal and spatial variables in the selected models, are found to explain a substantial amount of variance in these 75% of locations. The regions where the model selection routine ends up with a model of low explanatory power align with those locations where the rain intensity is relatively small by comparing with Figure 3.10.

3.6.3 Model Diagnostics

In order for our analysis to be valid, the selected model has to satisfy the assumptions of logistic regression. When the assumptions of logistic regression analysis are not met, we may have problems, such as biased coefficient estimates or very large standard errors for the logistic regression coefficients, and these problems may lead to invalid statistical inferences. Therefore, we need to check that our model fits sufficiently well and check for influential observations that have an impact on the estimates of the coefficients before we can use our model to make any statistical inference. In

this section, we are going to focus on conducting model diagnostics for the selected model.

Diagnostic methods can be graphical or numerical. We generally prefer graphical methods because they tend to be more versatile and informative. It is virtually impossible to verify that a given model is exactly correct. As George Box said: "all models are wrong, but some are useful". The purpose of the diagnostics is more to check whether the model is not grossly wrong.

Diagnostic methods can be divided into two types [64]. Some methods are designed to detect single cases or small groups of cases that do not fit the pattern of the rest of the data. Outlier detection is an example of this. Other methods are designed to check the assumptions of the model. These methods can be subdivided into those that check the structural form of the model, such as the choice and transformation of the predictors, and those that check the stochastic part of the model, such as the nature of the variance about the mean response. Here, we focus on methods for checking the assumptions of the model.

When we build a logistic regression model, we assume that the logit of the outcome variable is a linear combination of the independent variables. This involves two aspects, as we are dealing with the two sides of our logistic regression equation. First, consider the link function of the outcome variable on the left-hand side of the equation. We assume that the logit function (in logistic regression) is the correct function to use. Secondly, on the right-hand side of the equation, we assume that we have included all the relevant variables, that we have not included any variables that should not be in the model, and the logit function is a linear combination of the predictors. It could happen that the logit function as the link function is not the correct choice or the relationship between the logit of the outcome variable and the independent variables are not linear. In either case, we have a specification error. The misspecification of the link function is usually not too severe compared with using other alternative link function choices such as probit (based on the normal distribu-

tion). In practice, we are more concerned with whether our model has all the relevant predictors and if the linear combination of them is sufficient.

Residual analysis for logistic regression is more difficult than for linear regression models because the response Y_i take on only the value 0 and 1. Consequently, the i-th ordinary residual will assume one of two values:

$$e_i = \begin{cases} 1 - \hat{\pi}_i & \text{if } Y_i = 1 \\ -\hat{\pi}_i & \text{if } Y_i = 0 \end{cases}$$

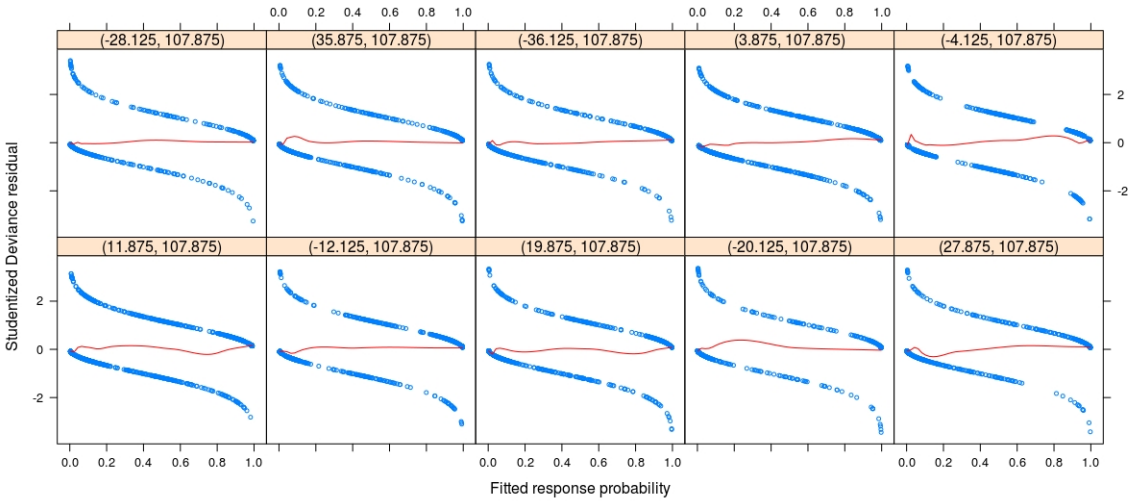


Fig. 3.25.: xyplot of studentized Deviance residual against fitted probability

The ordinary residuals will not be normally distributed and, indeed, their distribution under the assumption that the fitted model is correct is unknown. Plots of ordinary residuals against fitted values or predictor variables will generally be uninformative [65]. If the logistic regression model is correct, then $E(Y_i) = \pi_i$ and it follows asymptotically that:

$$E(Y_i - \pi_i) = Ee_i = 0$$

This suggests that if the model is correct, a lowess smooth of the plot of the residuals against the estimated probability $\hat{\pi}_i$ or against the linear predictor should result

approximately in a horizontal line with zero intercepts. Any significant departure from this line suggests that the model may be inadequate.

Figure 3.25 displays studentized Deviance residual plot for 10 out of 450 sampled locations. In fact, plots for other sampled locations are quite similar to Figure 3.25. The blue dots on each panel show studentized Deviance residual against the fitted probability for the corresponding location (latidude, longitude), superposed by the lowess smooth curve in red line. The fact that the lowess smooth approximates a line having zero slope and intercept suggests that there is apparently no significant model inadequacy.

Another way to check whether the model fits the data is to directly compare the fitted probabilities and observed values. If the probability of seeing $Y = 1$ is almost 1 when $Y = 1$, and similarly the probability of seeing $Y = 1$ is almost 0 when $Y = 0$, then the model fits the data.

Figure 3.26 is a xyplot of observed values and fitted probabilities against the index of 3-hr data in the year 1999 at the location $(4.125°S, 92.125°W)$. Observed values are indicated by colors: no rain in black and rain in red. The length of the bar at the index of time t corresponds to the fitted probability of final selected model at time t. Figure 3.26 shows that the rainfall probability is quite close to 1 in most cases when it rains, and the rainfall probability is close to 0 when there is no-rain. This pattern is observed in other representative sampled locations as well.

Finally, checking for multicollinearity is a standard operation in assessing model fit. This practice is relevant in explanatory modeling, where multicollinearity can lead to inflated standard errors, which interferes with inference. Generalized variance-inflation factor (VIF) for each explanatory variables in the final selected model can be obtained by using function vif in the car package. We can conclude that multi-collinearity is not an issue for the final selected model based on the quantile plot of max vif of explanatory variables in the selected model for 450 sampled locations in Figure 3.27.

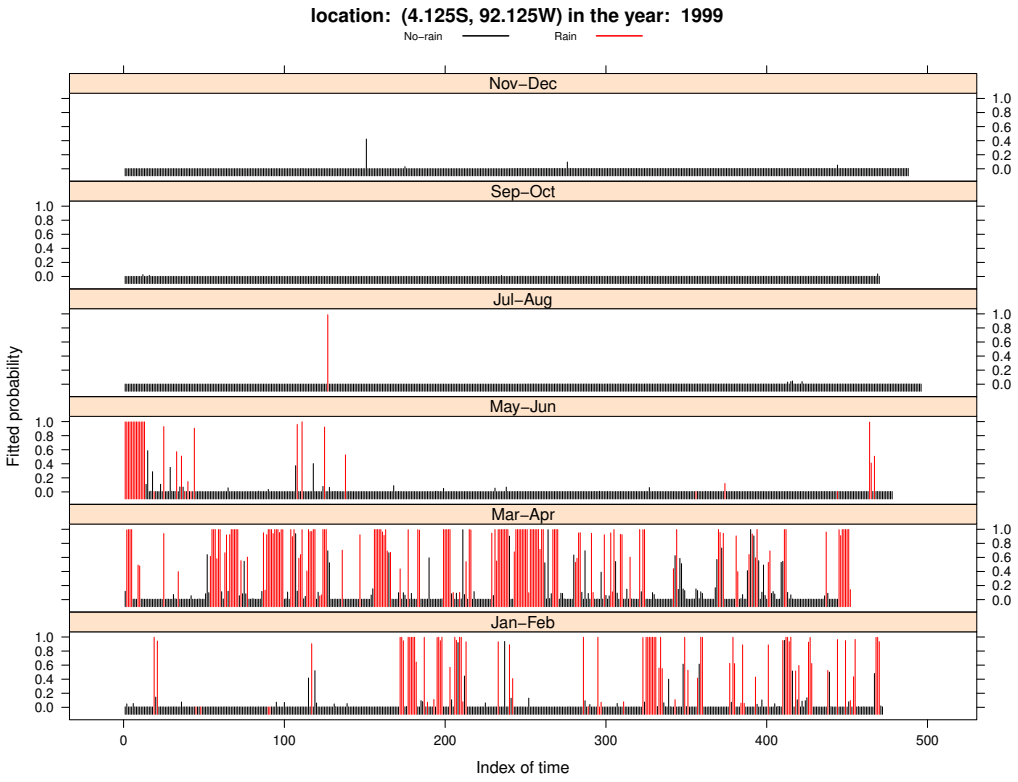


Fig. 3.26.: xyplot of response and fitted probability against the time

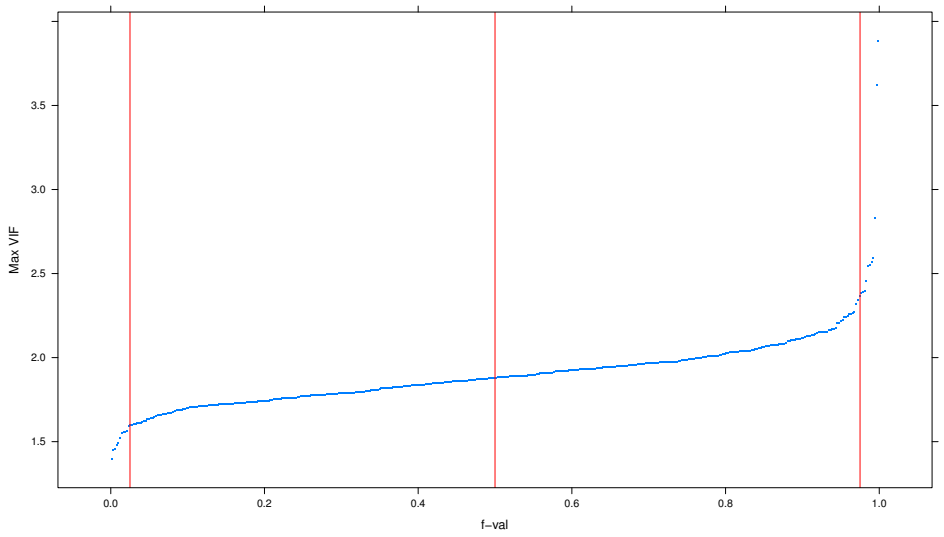


Fig. 3.27.: Quantile plot of max VIF of explanatory variables for 450 locations

3.6.4 Model Inference

Based on the model diagnostics, the final selected logistic model is an appropriate model for the data. Any complete data analysis requires that analysts are able to make statistical inference as well. Especially, the estimation of probability of rainfall and it's confidence interval are of high interest. Consider a Bayesian analysis with a uniform prior, the posterior distribution of parameter β is proportional to likelihood function, namely

$$p(\beta|X,y) \propto \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i}$$

where $logit(p_i) = \log(\frac{p_i}{1-p_i}) = X\beta$. The following strategy is useful for simulating a draw from the posterior predictive probability distribution of data, given draws from the posterior distribution of the parameters.

1. Draw the parameter vector β from its posterior distribution, $p(\beta|X,y)$, given the observed data (y,X).

2. Obtain a draw of predictive probability using $\pi = logit(X\beta)$ given the drawn β

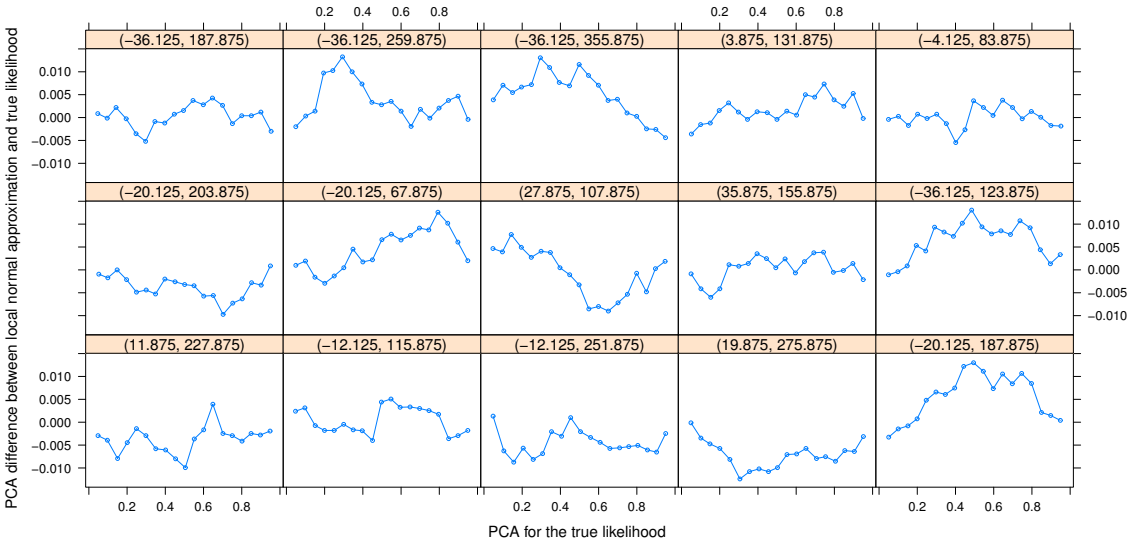


Fig. 3.28.: Normal approximation diagnostics using CPA

If the posterior distribution $p(\beta|X, y)$ is unimodal and roughly symmetric, it can be convenient to approximate it by a normal distribution; that is, the logarithm of the posterior density is approximated by a quadratic function of β [42].

$$p(\beta|X, y) \sim N(\hat{\beta}, [I(\hat{\beta})]^{-1})$$

where $I(\hat{\beta})$ is the observed fisher information, and

$$I(\beta) = -\frac{\partial^2 log(p(\beta|X, y))}{\partial \beta^2}.$$

Recall that we introduce CPA to measure the similarity between approximate multivariate likelihood function and the true multivariate likelihood function in chapter 2. Similarly, we can compare the posterior distributions of the hyper-parameters β and its normal approximate density using CPA as well. Figure 3.28 displays contour probability differences between approximate densities and the true posterior density under series of regions bounded by ellipsoids for 15 out of 450 representative locations. The larger the contour probability difference is, the further the approximate density departs away from the true posterior density. Collectively, all contour probability differences, no matter which ellipsoid region, no matter which location, are in the range between -0.03 and 0.038. Furthermore, most of contour probability differences are within 0.02. This implies that it is valid to approximate the posterior distribution of β using the normal distribution.

Here, we illustrate that the posterior predictive probability distribution using draws from the normal approximate distribution of parameters is quite close to the one using draws from the posterior distribution of the parameters in Figure 3.29. In each panel, one scatter point is the corresponding quantile of the predictive probability distribution for one observation using 1000 draws from normal approximate parameter distribution, against the one using the posterior distribution of parameters conditional on locations. The approximate predictive distribution performs quite well in terms of approximating quantiles of the posterior predictive distribution in that the scatter points lie along the straight line $y = x$. This result is promising as the

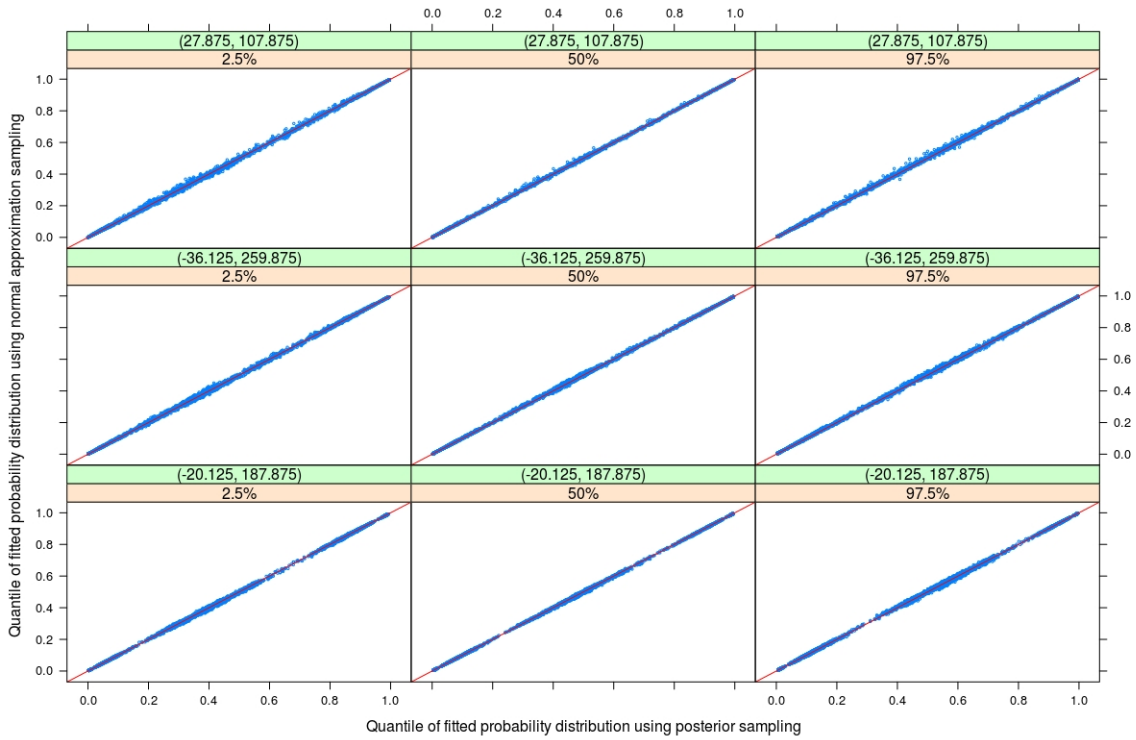


Fig. 3.29.: Comparison of approximate predictive probability distribution and the true one

predictive probability distribution for new data can easily obtained by using draws from the multivariate normal distribution with known mean and variance. To have an overall perspective of the 95% confidence interval of the fitted probability of rainfall occurrence, we compute the difference between the upper (lower) bound of 95% confidence interval and the median for each observation, instead of showing error bar of each fitted probability in the time series of length 49184.

Figure 3.30 only shows the result for 9 locations. In each panel, the blue points are for the difference between the upper bound of 95% confidence interval and the median for each observation, against the median probability; while the pink ones are for the difference between the median of 95% confidence interval and the lower bound. Scatter points in all panels appear in a parabola shape, indicating that the variance of low (high) probability of rainfall is relatively small in contrast with the one for

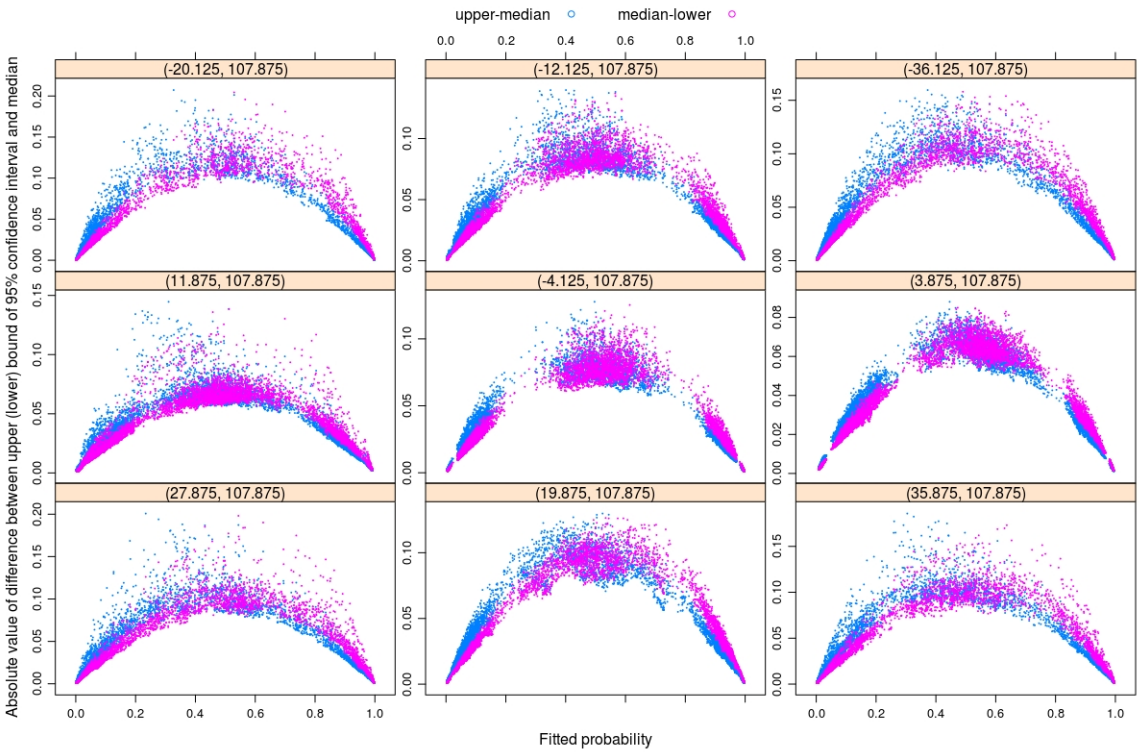


Fig. 3.30.: 95% confidence interval for fitted probability

the probability around 0.5. In other words, we have a high confidence in predicting no-rain when the fitted probability is small, and forecasting rain when the fitted probability is high. On the other hand, we are more uncertain whether it will rain or not when the fitted probability is around 0.5. This finding makes sense, empirically.

3.7 Predictive Modeling

Above two ST models are useful for explaining the variation of precipitation occurrence at the center location, but not applicable for predicting future rainfall occurrence because the neighborhood rain rates at the time of prediction is unknown. Before we start developing advanced models to predict 3-hr rainfall occurrence, lets try a simple, common-sense approach. It will serve as a sanity check, and will establish a baseline that we will have to beat in order to demonstrate the usefulness of

more-advanced models. For example, the dataset contains 80% observations of no-rain and 20% observations of rain, then a common-sense approach to the classification task is to always predict no-rain when we make a prediction in the further. Such a classifier is 80% accurate overall, and any learning-based approach should therefore beat this 80% score in order to demonstrate usefulness.

The first step is to find significant features to develop appropriate models to characterize the probability of precipitation occurrence at time t for each location. In rainfall prediction community, the first or second order Markov chain has been widely applied in the simulation of daily rainfall variability across multiple weather stations. With the consideration of different Markov chain orders and seasonal variability, we model the logit transformation of the probability of precipitation occurrence at location s at time t as a linear function of several lags of time series, the indicator of the month and hour, and year as follows.

- Model k: $logit(p(Y_{s,t} = 1)) = month\beta_m^s + year\beta_y^s + hour\beta_h^s + \sum_{j=1}^{k} \beta_{lj}^s lag_j$, $k = 1, \cdots, 8$

where $lag_j = Y_{s,t-j}$.

First of all, we fit each of 8 models ($k = 1, \cdots, 8$) to training data at sampled 450 locations and compute predicted proabilities on the test data. Given a specific model and a location, the corresponding ROC curve can be graphed and the AUC can be computed. Therefore, we can see the performance of the models in discriminating between rain and no-rain using the overall distribution of AUC for sampled 450 locations. In Figure 3.31, each line corresponds to the uniform quantile plot of AUC for one model. For example, the blue curve is the quantile plot of AUC for the model with lag one, the pink one is for the model with lag one and lag two, and so forth. In principle, the higher the AUC is, the better the model is. Based on Figure 3.31, the model 1 underperform the rest of considered models, which indicates that the first order Markov chain is not appropriate for 3-hr precipitation occurrences. Therefore, model 2 is selected as the pure temporal model, which will be considered

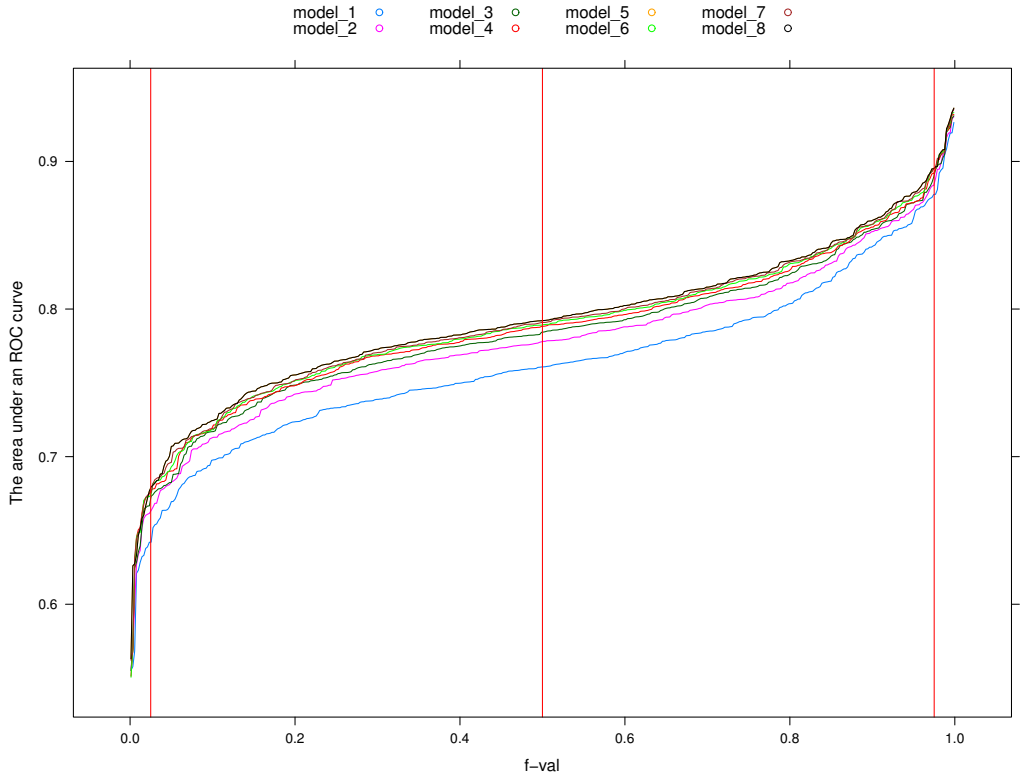


Fig. 3.31.: Uniform quantile plot of AUC for 8 models across 450 sampled locations. The red vertical lines are 0.025, 0.5, 0.975 quantiles.

as the benchmark model later, as it is the simplest model that has similar predictive power with other 6 models.

As shown in Figure 3.31, the proportion of locations where AUC is larger than 0.8 is less than 0.5, even for the best model. Definitely, we want to find a better model to predict 3-hr rainfall occurrence. One potential approach is to make full use of spatial information in the data which leads us to develop a spatial-temporal logistic model for 3-hr precipitation occurrences in the next section.

3.7.1 Two-stage Model

The unavailability of neighborhood predictors at the prediction time poses a big challenge for us to build powerful predictive models. A model with only temporal

predictors can be considered as the benchmark model, which is model 1 as follows. On the other hand, the ST model 1 with the assumption that the neighborhood information is available, is considered as our golden standard model. Can we propose other models that have a better predictive power than the benchmark model and achieve as close as possible to the predictive power of the golden standard model? Intuitively, it is a good idea to replace the observed spatial predictors with the predicted ones in the golden standard model. Therefore, we propose two-stage models as follows.

- Model 1: $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s$
- Model 2: $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + Loc_1\beta_1^s + \cdots + Loc_4\beta_4^s + Loc_6\beta_6^s + \cdots + Loc_9\beta_9^s$
- Model 3 (two-stage): $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + \hat{Loc}_1\beta_1^s + \cdots + \hat{Loc}_4\beta_4^s + \hat{Loc}_6\beta_6^s + \cdots + \hat{Loc}_9\beta_9^s$
- Model 4 (two-stage): $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + \tilde{Loc}_1\beta_1^s + \cdots + \tilde{Loc}_4\beta_4^s + \tilde{Loc}_6\beta_6^s + \cdots + \tilde{Loc}_9\beta_9^s$

Where $\hat{Loc}_i$ and $\tilde{Loc}_i$ are the fitted status of rainfall (0 or 1) and fitted probability of rainfall at t-th time period on i-th neighborhood location using Model 1, respectively.

Figure 3.32 shows the predictive power of four candidate models on the test data. At each location, we fit model 1 and model 2 to a set of the training data which consist of 3-hr rain rates from 1998 to 2013. Then we apply learning models to test data which include observations from 2014 to 2015 and compute the area under the corresponding ROC curves, respectively. For two-stage models, spatial predictors are obtained by computing predicted rainfall status or predicted rainfall probability using model 1 on the corresponding neighborhood location. Here the predicted rainfall status is determined by choosing the optimal threshold which maximizes the prediction accuracy. Collectively, there are 450 AUCs for each of four models over 450 representatives sampled locations. Uniform quantile plot of 450 AUCs for these four models is displayed in figure 3.32. Obviously, model 2 has a distinguished predictive

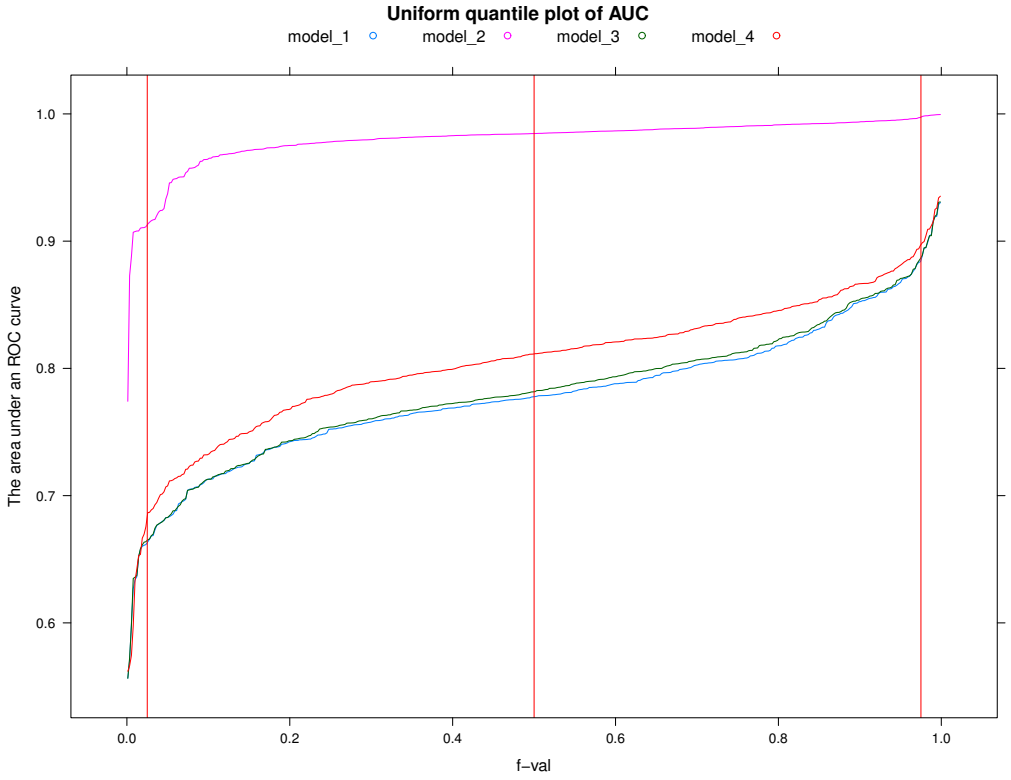


Fig. 3.32.: Uniform quantile plot of AUC for 4 models on test data across 450 sampled locations. The red vertical lines are 0.025, 0.5, 0.975 quantiles.

power since more than 97.5% of AUCs is larger than 0.9. Model 1 and two-stage model 3 perform quite similar in terms of the predictive power. Using fitted rainfall probability in the two-stage model 4 can improve the predictive power, comparing to the pure temporal model 1.

3.7.2 Markov Random Field Model

To build a more powerful predictive model, we propose a two-stage model (model 4), which is to replace the unknown rainfall status of the neighborhood with the fitted probability of rainfall. Another approach is the autologistic model which is a Markov random field model for spatial binary data [66]. One advantage of autologistic models is that they can model some interactions in a more direct and interpretable

fashion, capturing some of the dynamics of a process [67]. The spatial-temporal autologistic regression model captures the relationship between a binary response and potential explanatory variables, and adjusts for both spatial dependence and temporal dependence simultaneously by a space-time Markov random field [68].

Let Z be the random field of interest, where $Z_{s,t} \in \{0, 1\}$ represents the observation at the lattice site s and time point t with $s = 1, \cdots, n$ and $t = 1, \cdots, T$, the full conditional distributions for the traditional auto-logistic model are given by

$$logit(P(Z_{s,t} = 1)) = X_{s,t}\beta_s + \sum_{j \neq s} \alpha_{sj} Z_{j,t},$$

where $X_{s,t}$ is the temporal predictors at time t at site s, β_s are the regression parameters, and $\alpha_s = \{\alpha_{sj}, j \neq s\}$ are dependence parameters such that $\alpha_{sj} \neq 0$ iff Z_s and Z_j are neighbors.

In the TRMM data, it is about 3-hr rain rate observations at different locations at a series of time. They are stored as key-value pairs in HDFS with the time as key and a matrix of rain rates as the value. The spatial-temporal autologistic regression model building procedure is shown as follows:

- Step I: Swapping to a by-location division.

  The observations are divided into one key-value pair per location. The key is a pair of longitude and latitude index, and the value is observations across the time in the corresponding location. Each row represents an observation at a given time. More specifically, each observation consists of the status of rainfall occurrence at the center location and its corresponding 4 nearest neighbors, month, year, hour and its lags .

- Step II: Fit logistic regression in parallel.

  For each location, a logistic regression model is applied to the training observations. And then the coefficient parameters $\hat{\alpha}_s$ and $\hat{\beta}_s$ can be learned. Actually, $\hat{\gamma}_s = X_{s,test}\hat{\beta}_s$ can be computed for the test observations to save the data size to be shuffled in MapReduce job.

- Step III: Swapping to a by-time division.

  The observations are divided into one key-value pair per time. The key is the index of the test time and the value are $\hat{\beta}_s$ or $\hat{\gamma}_s$, and $\hat{\alpha}_s$ for the corresponding locations.

- Step IV: Gibbs sampling in parallel.

  Simulate $Z_{s,t}$ from the auto-logistic model:

$$Z_{s,t} \sim Binomial(1, p), p = \frac{\exp^{X_{s,t}\hat{\beta}_s + \sum_{j\sim s}\hat{\alpha_{sj}} Z^{\star}_{j,t}}}{1 + \exp^{X_{s,t}\hat{\beta}_s + \sum_{j\sim s}\hat{\alpha_{sj}} Z^{\star}_{j,t}}}$$

  where $j \sim s$ indicates j is the neighbor of s and $X_{s,t}$ is a temporal vector including year, month, hours and lags at site s. $Z^{\star}_{j,t}$ initializes at the observed rainfall occurrence. We can compute the ratio of $Z_{s,t} = 1$ from the sample given the location, which is an estimate of the probability of rainfall occurrence at time t.

- Step V: Swapping to by-location division.

  This step is similar to Step I. The key is still a pair of longitude and latitude index. Each observation of the value is estimated the probability of precipitation occurrence at the corresponding time at the given location.

To see the performance of the Markov random field model, we apply the whole procedure on TRMM train dataset and test dataset. For simplicity, we only use 4 nearest neighborhood locations due to similar predictive power with 8 neighborhood locations. The corresponding golden standard model and alternative good predictive model are proposed as follows for comparison.

- Model 1: $logit(p(Y_{s,t} = 1)) = year\beta^s_y + month\beta^s_m + hour\beta^s_h + lag_1\beta^s_{l1} + lag_2\beta^s_{l2} + Loc_2\beta^s_2 + Loc_4\beta^s_4 + Loc_6\beta^s_6 + Loc_8\beta^s_8$

- Model 2: $logit(p(Y_{s,t} = 1)) = year\beta^s_y + month\beta^s_m + hour\beta^s_h + lag_1\beta^s_{l1} + lag_2\beta^s_{l2} + Loc^{Gibbs}_2\beta^s_2 + Loc^{Gibbs}_4\beta^s_4 + Loc^{Gibbs}_6\beta^s_6 + Loc^{Gibbs}_8\beta^s_8$

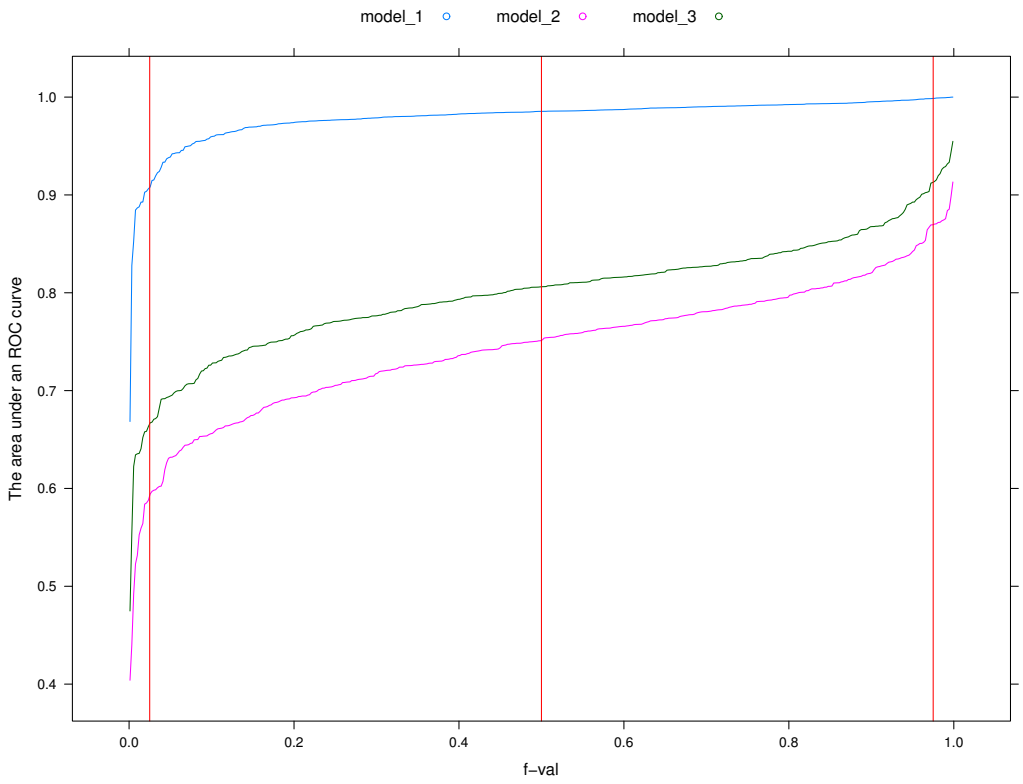


Fig. 3.33.: Uniform quantile plot of AUC for 3 models across 450 sampled locations. The red vertical lines are 0.025, 0.5, 0.975 quantiles.

- Model 3: $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + Loc_2^{t-1}\beta_2^s + Loc_4^{t-1}\beta_4^s + Loc_6^{t-1}\beta_6^s + Loc_8^{t-1}\beta_8^s$

Where Loc_2^{t-1} means the rainfall status of the previous time at location i. The fitted probability in Model 2 is estimated by using Gibbs sampling in spatial-temporal autologistic regression model building procedure

Figure 3.33 demonstrates that the Markov random field method does not improve the predictive power of the logistic regression. Even the logistic regression (model 3) which includes the rainfall status of the previous time at neighborhood locations outperforms the Markov random field.

3.7.3 Summary

Model 4 in the two-stage model subsection and Model 3 in the Markov random field model subsection are promising. To assess the overall performance on all locations, we apply the two-stage model, neighbor recurrent models (4 neighbors and 8 neighbors), benchmark model and golden model on 460,800 locations.

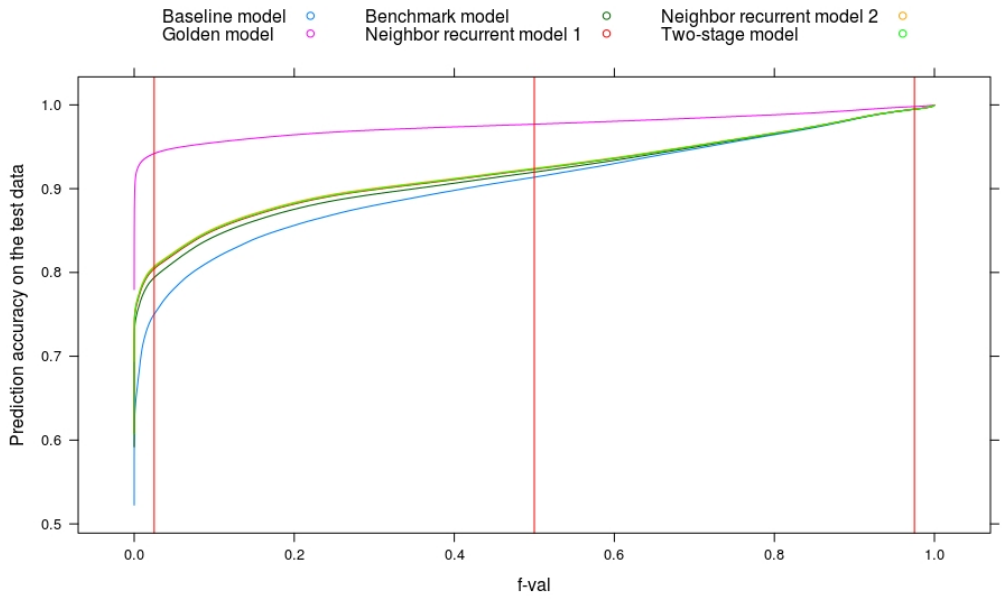


Fig. 3.34.: Uniform quantile plot of prediction accuracy on the test data for baseline model, benchmark model, golden model, two-stage model, neighbor recurrent model 1 and neighbor recurrent model 2

- Baseline model: $Y_{s,t} = 0$
- Benchmark model: $logit(p(Y_{s,t} = 1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s$
- Golden model: $logit(p(Y_{s,t} = 1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + Loc_1\beta_1^s + \cdots + Loc_4\beta_4^s + Loc_6\beta_6^s + \cdots + Loc_9\beta_9^s$

- Two-stage model): $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + \hat{Loc}_1\beta_1^s + \cdots + \hat{Loc}_4\beta_4^s + \hat{Loc}_6\beta_6^s + \cdots + \hat{Loc}_9\beta_9^s$
- Neighbor recurrent model 1: $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + Loc_2^{t-1}\beta_2^s + Loc_4^{t-1}\beta_4^s + Loc_6^{t-1}\beta_6^s + Loc_8^{t-1}\beta_6^s$
- Neighbor recurrent model 2: $logit(p(Y_{s,t}=1)) = year\beta_y^s + month\beta_m^s + hour\beta_h^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + Loc_1^{t-1}\beta_1^s + \cdots + Loc_4^{t-1}\beta_4^s + Loc_6^{t-1}\beta_6^s + \cdots + Loc_9^{t-1}\beta_9^s$

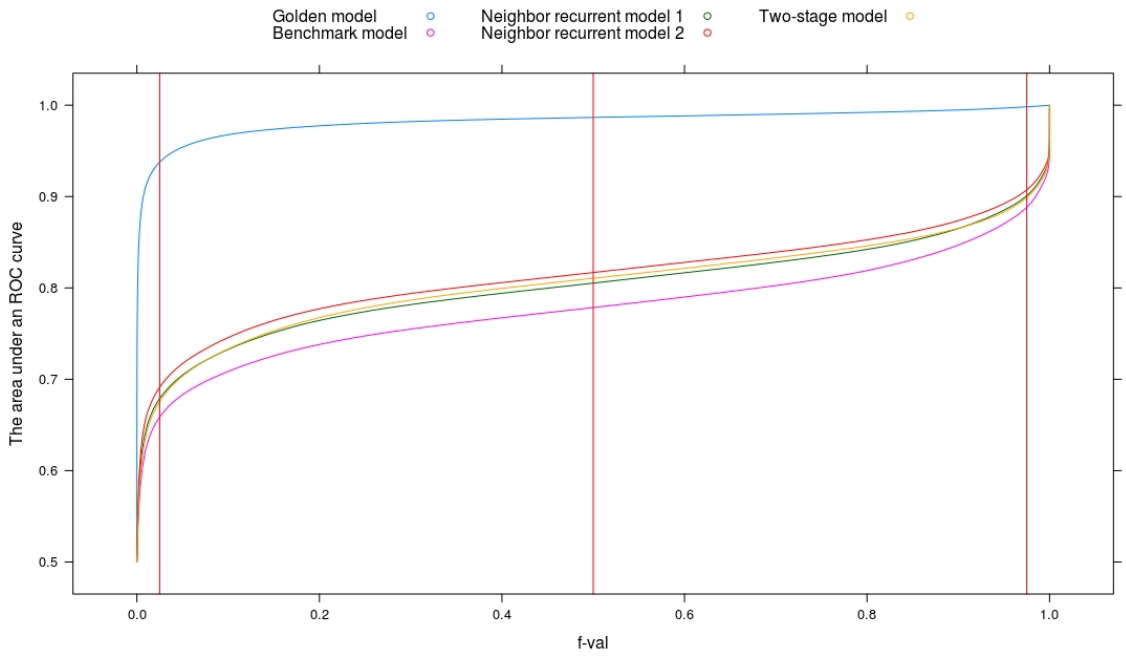


Fig. 3.35.: Uniform quantile plot of AUC on the test data for benchmark model, golden model, two-stage model, neighbor recurrent model 1 and neighbor recurrent model 2

In terms of prediction accuracy, Figure 3.34 demonstrates the distribution of classification accuracy of all 6 predictive models on the test data over all locations. 1) All advanced models outperform the baseline model; 2) The golden model has a significant better predictiion accuracy than those models without knowing the rainfall

status of neighborhoods; 3) Neighbor recurrent models and two-stage model have a quite similar prediction performance, but perform slightly better than the benchmark model.

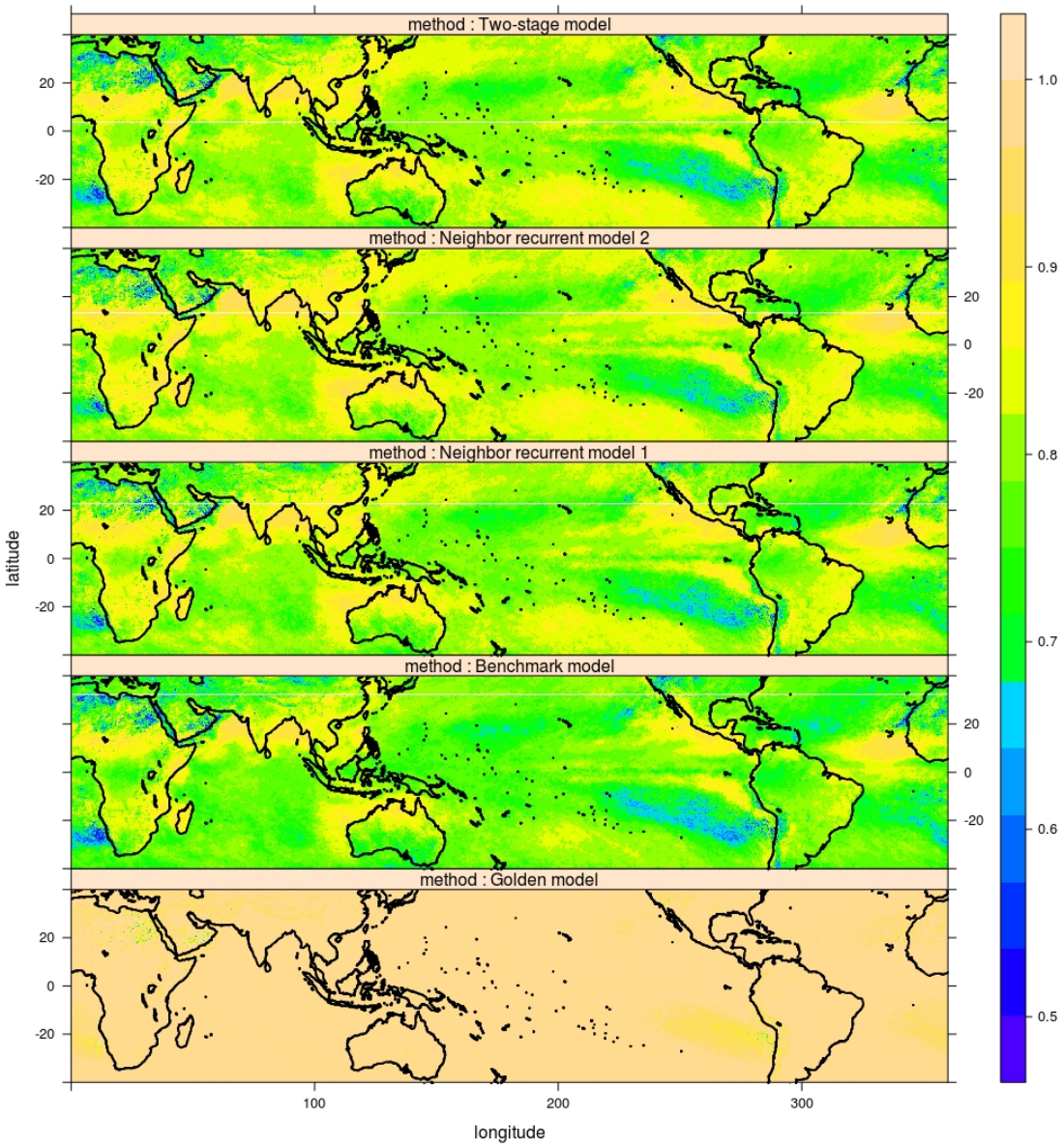


Fig. 3.36.: Levelplot of AUC on the test data for benchmark model, golden model, two-stage model, neighbor recurrent model 1 and neighbor recurrent model 2

The predictive power of these five candidate models except the baseline model over all locations, are shown in Figure 3.35 and Figure 3.36. We graph the quantile plot of the AUCs for each of five models on the test data at 460,800 locations in Figure 3.35. To see the performance geographically, we make levelplot of AUCs in Figure 3.36 for benchmark model, golden model, two-stage model, neighbor recurrent model 1 and neighbor recurrent model 2, respectively. Comparing with the benchmark model indicated in pink curve, advanced models such as the two-stage model and neighbor recurrent models have a higher predictive power. Overall, the neighbor recurrent model 2 is the best predictive model among above proposed models.

Furthermore, more complex models with higher order of predictors and interaction between predictors are fitted to 3-hr data, resulting in no improvements in predictive power.

3.8 Extreme Weather

Satellite-derived rainfall can be a critical tool for identifying hazards from flood events. One extension of spatio-temporal models developed in the previous section is to apply on heavy rainfall data. In this section, the goal is to build explanatory models for daily heavy rainfall occurrence based on the joint use of spatial and temporal features. We define heavy rain as follows:

$$Y_{s,t} = \begin{cases} 0 & \text{if } R_{s,t} \leq c_s \\ 1 & \text{if } R_{s,t} > c_s. \end{cases}$$

Where c_s is the threshold for location s.

Intuitively, the threshold varies from location to location, due to a large variation of rain rates and rain frequency across the earth. Here, the 95-quantile of rain rates at each location for each summer and winter season is chosen to be the threshold for the corresponding location and season. We use the concept of monsoon years" [69] starting with summer as May through October, followed by winter as November

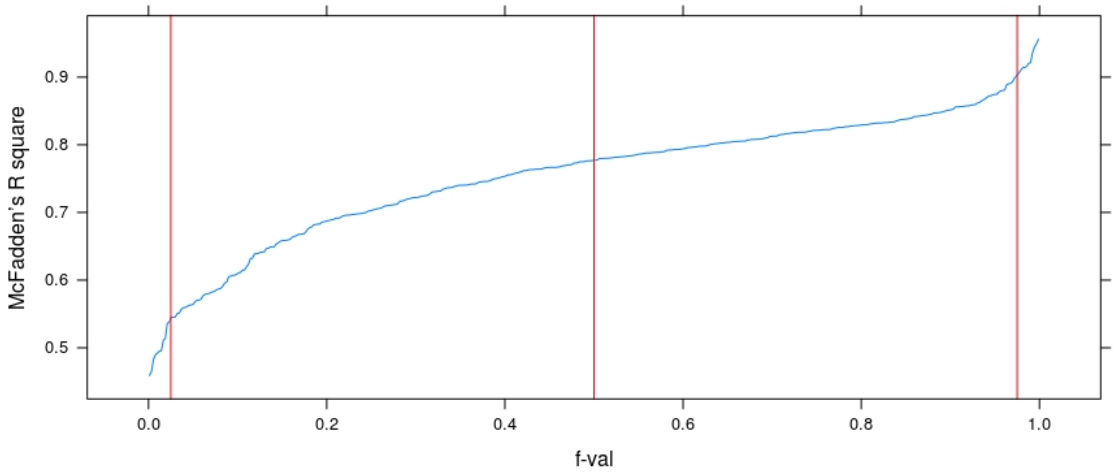


Fig. 3.37.: Uniform quantile plot of McFaddens R^2 on 450 sampled locations. The red vertical lines are 0.025, 0.5, 0.975 quantiles.

through the next April. Then we build spatio-temporal logistic models for each location with temporal features and spatial features as follows.

- Model: $logit(p(Y_{s,t} = 1)) = year\beta_y^s + lag_1\beta_{l1}^s + lag_2\beta_{l2}^s + season\beta_s^s + Loc_1\beta_1^s + \cdots + Loc_4\beta_4^s + Loc_6\beta_6^s + \cdots + Loc_9\beta_9^s$

Where season is the indicator of summer with value 1 at summer and 0 at winter. Loc_i indicates whether it is a heavy rain at the i-th neighborhood location near the center location s. We fit this model to 450 sampled locations and graph quantile plot of McFaddens R^2 in Figure 3.37. The plot implies that this spatial-temporal model has a great explanatory power overall.

3.9 Conclusion

In summary, we have described the procedure of data processing of the TRMM data, marginally analyzed the patterns of missingness over time and across space, followed by a brief introduction of sampling methods to obtain representative locations on which we can conduct comprehensive data analysis. Next, we studied the

spatial patterns of precipitation frequency, rainfall intensity and its variability, and the seasonal behaviors of monthly and yearly mean rain rates. Extensively utilizing DeltaRho computational environment, STL+ model were fitted to log-transformed monthly rain rates on all 460,800 locations, demonstrating that a significantly larger portion of variation in the data can be explained by the seasonal component than the trend component. Further spatial correlation analysis of the remainder components provided a strong evidence that the spatial features have an additional explanatory power of data, given that the seasonal variables are included in the model. Furthermore, we have proposed and validated spatio-temporal logistic models, which are automatically selected by using the stepwise AIC method, to explain the variation of the 3-hr precipitation occurrence for all 460,800 locations. The final selected models achieved a great explanatory power measured by McFadden's R^2 on more than 75% locations. Finally, we developed more advanced predictive models to forecast the probability of 3-hr precipitation occurrence on all locations: two-stage logistic regression model, spatial-temporal autologistic regression model, and neighbor recurrent logistic regression model. Overall, two-stage model and neighbor recurrent model displayed significantly higher predictive power, quantified by the AUC, than the spatial-temporal autologistic regression model and benchmark model. The regions where two-stage model and neighbor recurrent model did not show great predictive power has a property of low rainfall intensity.

REFERENCES

REFERENCES

[1] Saptarshi Guha, Ryan Hafen, Jeremiah Rounds, Jin Xia, Jianfu Li, Bowei Xi, and William S. Cleveland. Large complex data: divide and recombine (d&r) with rhipe. *Stat*, 1(1):53–67, 2012.

[2] Ryan Hafen, Luke Gosink, Jason McDermott, Karin Rodland, Kerstin Kleese-Van Dam, and William S Cleveland. Trelliscope: A system for detailed visualization in the deep analysis of large complex data. In *Large-Scale Data Analysis and Visualization (LDAV), 2013 IEEE Symposium on*, pages 105–112. IEEE, 2013.

[3] Deep analysis of large complex data.

[4] R Core Team et al. R: A language and environment for statistical computing. 2013.

[5] Apache hadoop.

[6] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[7] Saptarshi Guha. Computing environment for the statistical analysis of large and complex data. 2010.

[8] Reinaldo B Arellano-Valle and Adelchi Azzalini. The centred parametrization for the multivariate skew-normal distribution. *Journal of Multivariate Analysis*, 99(7):1362–1382, 2008.

[9] A. Azzalini and A. Dalla Valle. The multivariate skew-normal distribution. *Biometrika*, 83(4):715–726, 1996.

[10] Dani Gamerman. Sampling from the posterior distribution in generalized linear mixed models. *Statistics and Computing*, 7(1):57–68, 1997.

[11] Christian P Robert, George Casella, and George Casella. *Introducing monte carlo methods with r*, volume 18. Springer, 2010.

[12] Andrew Martin, Kevin Quinn, and Jong Hee Park. Mcmcpack: Markov chain monte carlo in r. *Journal of Statistical Software*, 42(1):1–21, 2011.

[13] Don van Ravenzwaaij, Pete Cassey, and Scott D Brown. A simple introduction to markov chain monte–carlo sampling. *Psychonomic bulletin & review*, pages 1–12, 2016.

[14] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[15] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[16] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In *Readings in Computer Vision*, pages 564–584. Elsevier, 1987.

[17] Radford M Neal. Probabilistic inference using markov chain monte carlo methods. 1993.

[18] Nicholas G. Polson, James G. Scott, and Jesse Windle. Bayesian inference for logistic models using plyagamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013.

[19] Sanvesh Srivastava, Volkan Cevher, Quoc Dinh, and David Dunson. Wasp: Scalable bayes via barycenters of subset posteriors. In *Artificial Intelligence and Statistics*, pages 912–920, 2015.

[20] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 681–688, 2011.

[21] Sungjin Ahn, Anoop Korattikara, and Max Welling. Bayesian posterior sampling via stochastic gradient fisher scoring. *arXiv preprint arXiv:1206.6380*, 2012.

[22] Tamara Broderick, Nicholas Boyd, Andre Wibisono, Ashia C Wilson, and Michael I Jordan. Streaming variational bayes. In *Advances in Neural Information Processing Systems*, pages 1727–1735, 2013.

[23] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[24] Steven L. Scott, Alexander W. Blocker, and Fernando V. Bonassi. Bayes and big data: The consensus monte carlo algorithm. In *Bayes 250*, 2013.

[25] Willie Neiswanger, Chong Wang, and Eric Xing. Asymptotically exact, embarrassingly parallel mcmc. *arXiv preprint arXiv:1311.4780*, 2013.

[26] Chong Wang, John Paisley, and David Blei. Online variational inference for the hierarchical dirichlet process. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 752–760, 2011.

[27] Alekh Agarwal and John C Duchi. Distributed delayed stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 873–881, 2011.

[28] Changye Wu and Christian P Robert. Average of recentered parallel mcmc for big data. *arXiv preprint arXiv:1706.04780*, 2017.

[29] Robert B Cleveland, William S Cleveland, and Irma Terpenning. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6(1):3, 1990.

[30] William S Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74(368):829–836, 1979.

[31] Ryan P Hafen. *Local regression models: Advancements, applications, and new methods*. Purdue University, 2010.

[32] Ariel Kleiner, Ameet Talwalkar, Purnamrita Sarkar, and Michael I. Jordan. A scalable bootstrap for massive data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(4):795–816, 2014.

[33] Ping Ma and Xiaoxiao Sun. Leveraging for big data regression. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(1):70–76, 2015.

[34] Faming Liang, Yichen Cheng, Qifan Song, Jincheol Park, and Ping Yang. A resampling-based stochastic approximation method for analysis of large geostatistical data. *Journal of the American Statistical Association*, 108(501):325–339, 2013.

[35] D. B. Rubin A. P. Dempster, N. M. Laird. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[36] Apache spark.

[37] Chieh-Yen Lin, Cheng-Hao Tsai, Ching-Pei Lee, and Chih-Jen Lin. Large-scale logistic regression and linear support vector machines using spark. In *Big Data (Big Data), 2014 IEEE International Conference on*, pages 519–528. IEEE, 2014.

[38] Gautier Philip. *Divide and Recombine for Large Complex Data: The Subset Likelihood Modeling Approach to Recombination*. PhD thesis, Purdue University, 2015.

[39] Frank J Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.

[40] James D Loudin and Hannu E Miettinen. A multivariate method for comparing n-dimensional distributions. In *Proceedings of the Conference on Statistical Problems in Particle Physics, Astrophysics and Cosmology (PHYSTAT)*, pages 207–210, 2003.

[41] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[42] Stern H.S. Gelman A., Carlin J.B. and et al. *Bayesian Data Analysis*. Chapman and Hall/CRC, 2013.

[43] Prasun K Kundu and Ravi K Siddani. A new class of probability distributions for describing the spatial statistics of area-averaged rainfall. *Journal of Geophysical Research: Atmospheres*, 112(D18), 2007.

[44] Michael D Hudlow and Vernon L Patterson. Gate radar rainfall atlas. 1979.

[45] Ying Sun, Michael L Stein, et al. A stochastic space-time model for intermittent precipitation occurrences. *The Annals of Applied Statistics*, 9(4):2110–2132, 2015.

[46] Prasun K Kundu and Ravi K Siddani. Scale dependence of spatiotemporal intermittence of rain. *Water Resources Research*, 47(8), 2011.

[47] Thomas L Bell. A space-time stochastic model of rainfall for satellite remote-sensing studies. *Journal of Geophysical Research: Atmospheres*, 92(D8):9631–9643, 1987.

[48] Xiaogu Zheng and Richard W Katz. Simulation of spatial dependence in daily rainfall using multisite generators. *Water Resources Research*, 44(9), 2008.

[49] James P Hughes, Peter Guttorp, and Stephen P Charles. A non-homogeneous hidden markov model for precipitation occurrence. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(1):15–30, 1999.

[50] S Renuga Devi, P Arulmozhivarman, C Venkatesh, and Pranay Agarwal. Performance comparison of artificial neural network models for daily rainfall prediction. *International Journal of Automation and Computing*, 13(5):417–427, 2016.

[51] Sigit Hardwinarto, Marlon Aipassa, et al. Rainfall monthly prediction based on artificial neural network: A case study in tenggarong station, east kalimantan-indonesia. *Procedia Computer Science*, 59:142–151, 2015.

[52] George J Huffman, David T Bolvin, Eric J Nelkin, David B Wolff, Robert F Adler, Guojun Gu, Yang Hong, Kenneth P Bowman, and Erich F Stocker. The trmm multisatellite precipitation analysis (tmpa): Quasi-global, multiyear, combined-sensor precipitation estimates at fine scales. *Journal of hydrometeorology*, 8(1):38–55, 2007.

[53] George J Huffman and David T Bolvin. Trmm and other data precipitation data set documentation. *NASA, Greenbelt, USA*, 28, 2013.

[54] Galit Shmueli et al. To explain or to predict? *Statistical science*, 25(3):289–310, 2010.

[55] Danilo Bzdok and BT Thomas Yeo. Inference in the age of big data: Future perspectives on neuroscience. *NeuroImage*, 2017.

[56] Mohammad R Arbabshirani, Sergey Plis, Jing Sui, and Vince D Calhoun. Single subject prediction of brain disorders in neuroimaging: promises and pitfalls. *NeuroImage*, 145:137–165, 2017.

[57] Paul D Allison. Measures of fit for logistic regression. In *Proceedings of the SAS Global Forum 2014 Conference*, 2014.

[58] Martina Mittlböck, Michael Schemper, et al. Explained variation for logistic regression. *Statistics in medicine*, 15(19):1987–1997, 1996.

[59] Scott Menard. Coefficients of determination for multiple logistic regression analysis. *The American Statistician*, 54(1):17–24, 2000.

[60] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[61] F.J. Anscombe. Graphs in statistical analysis. *American Statistician*, 27:17–21, 1973.

[62] Hye-Kyung Cho, Kenneth P Bowman, and Gerald R North. A comparison of gamma and lognormal distributions for characterizing satellite rain rates from the tropical rainfall measuring mission. *Journal of Applied Meteorology*, 43(11):1586–1597, 2004.

[63] HG Takahashi, H Fujinami, T Yasunari, and J Matsumoto. Diurnal rainfall pattern observed by tropical rainfall measuring mission precipitation radar (trmm-pr) around the indochina peninsula. *Journal of Geophysical Research: Atmospheres*, 115(D7), 2010.

[64] Julian J Faraway. Extending the linear model with r: Generalized linear. *Mixed effects and nonparametric regression models*, 1, 2006.

[65] Michael H Kutner, Chris Nachtsheim, John Neter, and William Li. *Applied linear statistical models*. McGraw-Hill Irwin, 2005.

[66] John Hughes, Murali Haran, and Petruţa C Caragea. Autologistic models for binary data on a lattice. *Environmetrics*, 22(7):857–871, 2011.

[67] Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 192–236, 1974.

[68] Jun Zhu, Hsin-Cheng Huang, and Jungpin Wu. Modeling spatial-temporal binary data using markov random fields. *Journal of Agricultural, Biological, and Environmental Statistics*, 10(2):212, 2005.

[69] Tetsuzo Yasunari. The monsoon yeara new concept of the climatic year in the tropics. *Bulletin of the American Meteorological Society*, 72(9):1331–1338, 1991.

APPENDICES

Table 3.: California Democratic Poll Exit

fips	total_voters	sample_voters	sample_Clinton
6001	199445	100	52
6003	241	198	94
6005	3769	150	75
6007	24202	103	33
6009	5126	104	54
6011	1275	100	45
6013	117523	122	68
6015	2388	179	81
6017	20130	166	79
6019	55285	155	92
6021	1321	177	95
6023	19470	153	46
6025	8597	196	129
6027	1749	124	53
6029	33340	112	60
6031	6623	163	98
6033	5189	127	62
6035	1516	198	91
6037	1035968	144	61
6039	8688	101	54
6041	47288	123	71
6043	2048	115	62
6045	7390	140	43
6047	12577	126	61
6049	551	200	81

continued on next page

Table 3.: *continued*

fips	total_voters	sample_voters	sample_Clinton
6051	1681	118	61
6053	30311	146	90
6055	12242	177	99
6057	14154	187	75
6059	226598	165	93
6061	30402	112	69
6063	2747	173	65
6065	123078	152	90
6067	119943	166	88
6069	3504	101	62
6071	124555	124	69
6073	253744	138	75
6075	153003	140	83
6077	42003	121	81
6079	33266	175	99
6081	77763	189	118
6083	46898	184	97
6085	181757	162	105
6087	45486	150	59
6089	12290	113	58
6091	493	183	81
6093	3962	106	39
6095	55903	177	106
6097	88257	128	70
6099	27885	117	69

continued on next page

Table 3.: *continued*

fips	total_voters	sample_voters	sample_Clinton
6101	4340	120	65
6103	3117	154	86
6105	1568	103	40
6107	14414	168	106
6109	5557	182	100
6111	85219	130	65
6113	24260	163	81
6115	3387	196	85

Package 'LM.logit'

February 18, 2018

Type Package

Title Likelihood Modeling for Logistic Regression

Version 1.0

Date 2017-09-21

Author Qi Liu

Maintainer Qi Liu <liuqi.jlu@gmail.com>

Description

In divide and recombine framework, big data are divided into subsets, each analytic method is applied to subsets, and the outputs are recombined. The likelihood-model for logistic regression is a parametric probability density function of the parameters in the logistic regression. The density parameters are estimated by fitting the density to MCMC draws from each subset data-model likelihood function, and then the fitted densities are recombined.

Imports datadr, sn, BayesLogit, MASS, mvtnorm, moments

License MIT + file LICENSE

RoxygenNote 6.0.1

R topics documented:

`LM.logit-package` *Likelihood Modeling for Logistic Regression*

Description

In divide and recombine framework, big data are divided into subsets, each analytic method is applied to subsets, and the outputs are recombined. The likelihood-model for logistic regression is a parametric probability density function of the parameters in the logistic regression. The density parameters are estimated by fitting the density to MCMC draws from each subset data-model likelihood function, and then the fitted densities are recombined.

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.
The likelihood-model of logistic regression is a parametric probability density function of the parameters in the logistic regression. The density parameters are estimated by fitting the density to MCMC draws from each subset data-model likelihood function, and then the fitted densities are recombined.

Author(s)

Qi Liu

Maintainer: Qi Liu <liuqi.jlu@gmail.com>

References

- http://deltarho.org
- Qi Liu, Anindya Bhadra, Bowei Xi, and William S. Cleveland, Likelihood modeling for big data analysis using divide and recombine methods

See Also

datadr

Examples

```
## Not run:
set.seed(100)
library(datadr)
library(sn)
library(BayesLogit)
library(MASS)
library(mvtnorm)
library(moments)
x <- matrix(rnorm(1000*5), ncol=5)
ttheta <- rep(1,5)
y <- rbinom(1000, 1, 1 / (1 + exp(- x %*% ttheta)))
df <- cbind(y,x)
df <- as.data.frame(df)
names(df) <- c("y", "x1","x2","x3","x4","x5")
df_ddf <- ddf(df)
```

```
# in memory backend
 df_div <- divide(df_ddf, by =rrDiv(500))
 rst<- drml(df_div, y~x1+x2, size =1000, burnin =50, approx_method = "SN")
 pred <- predNew.local(rst, y~x1+x2, df, 1000)

 # disc backend
 tmpdir <- "./tmp"
 DiskConn <- localDiskConn(file.path(tmpdir, "KV"), autoYes = TRUE)
 addData(DiskConn, df_div)
 DiskConn <- ddf(DiskConn)
 DiskConn <- updateAttributes(DiskConn)
 rst<- drml(DiskConn, y~x1+x2, size =1000, burnin =50, approx_method = "SN")
 DiskConn_output <- localDiskConn(file.path(tmpdir, "output1"), autoYes = TRUE)
 pred <- predNew.dr(rst, y~x1+x2, DiskConn, 1000, DiskConn_output)
 head(pred[[1]]$value)

 # hdfs backend
 library(Rhipe)
 rhinit()
 seq.file <- list()
 seq.file[[1]] <-  list(2, df[1:500,])
 seq.file[[2]] <-  list(2, df[501:1000,])
 rhwrite(seq.file, file="/tmp/test1", chunk=1, kvpairs=T, verbose=F)
 HDFSconn <- hdfsConn("/tmp/test1", autoYes = TRUE)
 HDFSconn <- ddo(HDFSconn)
 HDFSconn <- updateAttributes(HDFSconn)
 rst<- drml(HDFSconn, y~x1+x2, size =1000, burnin =50, approx_method = "SN")
 HDFSoutput <- hdfsConn("/tmp/output", autoYes = TRUE)
 pred <- predNew.dr(rst, y~x1+x2, HDFSconn, 1000, HDFSoutput)
 head(pred[[1]]$value)

## End(Not run)
```

`drml` *Model Likelihood of Logistic Regression in Divide and Recombine Framework*

Description

Model the posterior distribution of parameters in logistic regression by normal distribution or skew normal distribution, where the prior distribution is the uniform distribution.

Usage

```
drml(ddo_object, formula = formula, size, burnin,
  approx_method = approx_method)
```

Arguments

ddo_object	a ddo/ddf object (in memory,) which is obtained by dividing whole data into subsets
formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are the same to the formula in lm.
size	the number of MCMC iterations saved (target distribution is the posterior distribution of parameters in the logistic regression)
burnin	the number of MCMC iterations discarded.
approx_method	the method to approximate the posterior distribution such as normal or skew normal, the default one is normal distribution

Value

norm.mean	mean parameter of recombined fitted normal distribution
norm.var	variance (covariance) of recombined fitted normal distribution
sn.mod	mean parameter of normal approximation to recombined fitted skew normal distribution if approx_method is "SN"
sn.cov	variance (covariance) of normal approximation to recombined fitted skew normal distribution if approx_method is "SN"

Author(s)

Qi Liu

See Also

divide, recombine

Examples

```
set.seed(100)
library(datadr)
x <- matrix(rnorm(1000*5), ncol=5)
ttheta <- rep(1,5)
y <- rbinom(1000, 1, 1 / (1 + exp(- x %*% ttheta)))
df <- cbind(y,x)
df <- as.data.frame(df)
names(df) <- c("y", "x1","x2","x3","x4","x5")
df_ddf <- ddf(df)
df_div <- divide(df_ddf, by =rrDiv(500))
rst<- drml(df_div, y~x1+x2, size =1000, burnin =50, approx_method = "Norm")
```

LMsubset *Subset Likelihood Modeling for Logistic Regression*

Description

Model the posterior distribution of parameters (likelihood function) in logistic regression using the normal distribution or skew normal distribution, where the prior distribution is the uniform distribution.

Usage

```
LMsubset(formula = formula, data = data, size, burnin, conf_level,
  approx_method = approx_method)
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are the same to the formula in lm.
data	a data frame containing the variables in the model. If not found in data, the variables are taken from environment(formula).
size	the number of MCMC iterations saved (target distribution is the likelihood function of parameters in the logistic regression)
burnin	the number of MCMC iterations discarded.
conf_level	a vector which consists of levels of credible intervals
approx_method	the method to approximate the posterior distribution such as normal ("Norm") or skew normal ("SN"), the default one is normal distribution

Details

Fit logistic regression to data with formula, simulate a sample of size = size with burnin = burnin using Monte carol methods from the posterior distribution (likelihood function) of coefficients. There are two approximate methods considered in this function: Normal approximation and Skew-normal approximation.

Value

prob_compare	a dataframe with two columns: approximate probability and true proability. The probability under different credible regions defined by conf_levels is estimated by using Monte Carlo methods
norm.mean	mean parameter of fitted normal distribution
norm.var	variance (covariance) of fitted normal distribution
sn.xi	location parameter of fitted skew normal distribution if approx_method is "SN", Null otherwise

sn.omega	scale parameter of fitted skew normal distributionif approx_method is "SN", Null otherwise
sn.alpha	shape parameter of fitted skew normal distributionif approx_method is "SN", Null otherwise

Author(s)

Qi Liu

See Also

subset_approx

Examples

```
x <- matrix(rnorm(1000*5), ncol=5)
y <- rbinom(1000,1,0.5)
df <- as.data.frame(x)
names(df) <- paste("x",1:5,sep="")
df$y <- y
rst <- LMsubset(y~x1+x2, data =df, size=500, burnin =50, conf_level = seq(0.05, 0.95, 0.05), approx_method = "SN"
```

---

predNew.dr *Fitted Values at distributed Datasets Based on the Fitted Results from Likelihood Modeling*

Description

predNew.dr is a function to provide the 0.025, 0.5, 0.975 quantiles of the distribution of fitted predict probability based on the fitted density of model parameters.

Usage

```
predNew.dr(fitted_par, formula, ddo_object, size = 1000, output)
```

Arguments

fitted_par	object returned by drml function
formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are the same to the formula in lm.
ddo_object	a ddo or ddf object initiated from HDFS connection or localDisk connection
size	the number of samples drawn from the distribution of model parameters
output	a "kvConnection" object indicating where the output data should reside (see localDiskConn, hdfsConn).

Value

0.025, 0.5, 0.975 quantiles of the distribution of fitted predict probability

See Also

predNew.local, datadr

Examples

```
## Not run:
set.seed(100)
library(datadr)
x <- matrix(rnorm(1000*5), ncol=5)
ttheta <- rep(1,5)
y <- rbinom(1000, 1, 1 / (1 + exp(- x %*% ttheta)))
df <- cbind(y,x)
df <- as.data.frame(df)
names(df) <- c("y", "x1","x2","x3","x4","x5")

# local disk backend
tmpdir <- "./tmp"
 DiskConn <- localDiskConn(file.path(tmpdir, "KV"), autoYes = TRUE)
 addData(DiskConn, df_div)
 DiskConn <- ddf(DiskConn)
 DiskConn <- updateAttributes(DiskConn)
 rst<- drml(DiskConn, y~x1+x2, size =1000, burnin =50, approx_method = "SN")
 DiskConn_output <- localDiskConn(file.path(tmpdir, "output1"), autoYes = TRUE)
 pred <- predNew.dr(rst, y~x1+x2, DiskConn, 1000, DiskConn_output)
 head(pred[[1]]$value)

 # HDFS backend
library(Rhipe)
 rhinit()
 seq.file <- list()
 seq.file[[1]] <-  list(2, df[1:500,])
 seq.file[[2]] <-  list(2, df[501:1000,])
 rhwrite(seq.file, file="/tmp/test1", chunk=1, kvpairs=T, verbose=F)

 HDFSconn <- hdfsConn("/tmp/test1", autoYes = TRUE)
 HDFSconn <- ddo(HDFSconn)
 HDFSconn <- updateAttributes(HDFSconn)
 rst<- drml(HDFSconn, y~x1+x2, size =1000, burnin =50, approx_method = "SN")
 HDFSoutput <- hdfsConn("/tmp/output", autoYes = TRUE)
 pred <- predNew.dr(rst, y~x1+x2, HDFSconn, 1000, HDFSoutput)
 head(pred[[1]]$value)

## End(Not run)
```

predNew.local *Fitted Values at New Data (in Memory) Based on the Fitted Results from Likelihood Modeling*

Description

predNew.local is a function to provide the 0.025, 0.5, 0.975 quantiles of the distribution of fitted predict probability based on the fitted density of model parameters.

Usage

```
predNew.local(fitted_par, formula, data, size = 1000)
```

Arguments

fitted_par	object returned by drml function
formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are the same to the formula in lm.
data	a data frame containing the variables in the model. If not found in data, the variables are taken from environment(formula).
size	the number of samples drawn from the distribution of model parameters

Value

0.025, 0.5, 0.975 quantiles of the distribution of fitted predict probability

Examples

```
set.seed(100)
library(datadr)
x <- matrix(rnorm(1000*5), ncol=5)
ttheta <- rep(1,5)
y <- rbinom(1000, 1, 1 / (1 + exp(- x %*% ttheta)))
df <- cbind(y,x)
df <- as.data.frame(df)
names(df) <- c("y", "x1","x2","x3","x4","x5")
df_ddf <- ddf(df)
df_div <- divide(df_ddf, by =rrDiv(500))
rst<- drml(df_div, y~x1+x2, size =1000, burnin =50, approx_method = "SN")
pred <- predNew.local(rst, y~x1+x2, df, 1000)
```

subset_approx *Likelihood Modeling for Logistic Regression in subset sense*

Description

Model the posterior distribution of parameters (likelihood function) in logistic regression using the normal distribution or skew normal distribution, where the prior distribution is the uniform distribution.

Usage

```
subset_approx(x, y, size, burnin, conf_level, approx_method = approx_method)
```

Arguments

x	the model matrix
y	the response variable
size	the number of MCMC iterations saved (target distribution is the posterior distribution of parameters in the logistic regression)
burnin	the number of MCMC iterations discarded.
conf_level	a vector which consists of levels of credible intervals
approx_method	the method to approximate the posterior distribution such as normal or skew normal, the default one is normal distribution

Value

prob_compare	a dataframe with two columns: approximate probability and true proability. The probability under different credible regions defined by conf_levels is estimated by using Monte Carlo methods
norm.mean	mean parameter of fitted normal distribution
norm.var	variance (covariance) of fitted normal distribution
sn.xi	location parameter of fitted skew normal distribution if approx_method is "SN", Null otherwise
sn.omega	scale parameter of fitted skew normal distributionif approx_method is "SN", Null otherwise
sn.alpha	shape parameter of fitted skew normal distributionif approx_method is "SN", Null otherwise

Author(s)

Qi Liu

Examples

```
x <- matrix(rnorm(1000*5), ncol=5)
y <- rbinom(1000,1,0.5)
a <- subset_approx(x,y, size=5000, burnin =500, conf_level = seq(0.05, 0.95, 0.05), approx_method = "Norm")
```

VITA

VITA

Education

Purdue University, West Lafayette, IN, U.S.A. 05/2018

- **Ph.D.** candidate in Statistics, Department of Statistics GPA: 4.00/4.00
- Advisor: Dr. William S. Cleveland
- Research Interests: Likelihood Modeling within Divide and Recombine Framework, Machine Learning, Data Visualization

Chinese Academy of Sciences, Beijing, China 06/2013

- **M.S.** in Applied Mathematics, Academy of Mathematics and Systems Sciences (AMSS) GPA: 3.70/4.00
- Thesis Topic: Hybrid Symbolic-numerical Computation

Jilin University, Changchun, China 06/2010

- **B.S.** in Mathematics, School of Mathematics GPA: 3.80/4.00

Publications and Presentations

Publications

- **Qi Liu**, Anindya Bhadra and William S. Cleveland, Divide and Recombine for Large and Complex Data: Model Likelihood Functions Using MCMC, arXiv preprint arXiv:1801.05007, 2018.
- **Qi Liu**, Vinayak Rao, Matt Bowers, Wen-wen Tung and William S. Cleveland, Modeling of Tropical Rainfall Measuring Mission Big Data (to be submitted)

- Zhe Li and **Qi Liu**, A heuristic verification of the degree of the approximate GCD of two univariate polynomials *Numerical Algorithms* **(SCI)**. doi: 10.1007/s11075-013-9793-9 (2013)
- Matthew Bowers, Wen-wen Tung, **Qi Liu**, William Cleveland, Recent changes in the temporal clustering patterns of tropical rainfall inferred from TRMM data (submitted)

Presentations

- **Qi Liu.** Tropical Rainfall Measuring Mission (TRMM) Data Using Divide & Recombine Methods. *Novartis.* East Hanover, NJ 02/2018
- **Qi Liu.** Divide & Recombine for Large and Complex Data: Model Likelihood Functions using MCMC. *Novartis.* East Hanover, NJ 11/2017
- **Qi Liu.** Divide & Recombine with DeltaRho Visualization and Modeling of Tropical Rainfall Measuring Mission (TRMM) Data. *Big Data and the Earth Sciences: Grand Challenges Workshop. 2017.* San Diego, CA 06/2017
- **Qi Liu.** A heuristic verification of the degree of the approximate GCD of multiple univariate polynomials. *2013 Verification Symposium Based on Symbolic and Numerical Computation. 2013.* Beijing, China 04/2013