Stochastic Precipitation Generation for the Potomac River Basin Using Hidden Markov Models

CyberTraining: Big Data + High-Performance Computing + Atmospheric Sciences

Gerson C. Kroiz[1], Jonathan N. Basalyga[1], Uchendu Uchendu[2],
RAs: Reetam Majumder[1], Carlos A. Barajas[1], Mentor: Matthias K. Gobbert[1],
Clients: Kel Markert[3], Amita Mehta[4], Nagaraj K. Neerchal[1,5]

[1]Department of Mathematics and Statistics, UMBC
[2]Department of Information Systems, UMBC
[3]The University of Alabama in Huntsville
[4]Joint Center for Earth Systems Technology, UMBC
[5]Chinmaya Vishwavidyapeeth, Kerala, India

Technical Report HPCF–2020–11, hpcf.umbc.edu > Publications

Abstract

A daily precipitation generator based on hidden Markov models (HMM) using satellite precipitation estimates is studied for the Potomac river basin in Eastern USA over the wet season months of July to September. GPM-IMERG data between 2001–2018 is used for the study, which at a $0.1^\circ \times 0.1^\circ$ spatial resolution results in 387 grid points across the basin. A 4-state model has been considered for the state process, and the semi-continuous emission distribution for precipitation at each location is modeled using a mixture comprising a delta function at 0 and two Gamma distributions. The underestimation of the observed spatial correlations between the grid points based on this model is noted, and the HMM is extended using Gaussian copulas to generate spatially correlated precipitation amounts. Performance of this model is examined in terms of dry and wet day stretches, spatial correlations between grid points, and extreme precipitation events. The HMM with Gaussian copulas (HMM-GC) is shown to outperform the classical HMM formulation for precipitation generation when using remote sensing data in the Potomac river basin.

Key words. Spatiotemporal modeling, Hidden Markov models, Precipitation generation, Stochastic simulations

1 Introduction

The modeling and forecasting of precipitation play a significant role in determining water allocation and resource management for regions that are dependent on precipitation as a primary water source. Simulations from these models can be used as input for agricultural, hydrological and ecological modeling. It is important to model both the seasonal and interannual variations of precipitation, and one method of achieving this is through precipitation generators. Using a dynamical or statistical model created from historical precipitation data, these generators can produce time series of synthetic data representative of the general

rainfall patterns of the region. Stochastic weather generators aim to replicate key statistical properties of the historical data, including sequences of dry and wet days, pairwise spatial correlations and extreme weather events. These models are used at different spatial and temporal resolutions, and our interest lies in modeling multi-site daily precipitation.

We use a hidden Markov model (HMM) to simulate daily precipitation data for the Potomac river basin. Located by the East Coast of the USA, the basin is the primary source of water for the region and receives large portions of its water supply from rainfall. HMMs have been widely used for simulating daily rainfall based on observed weather station data, and they factor the joint distribution of daily rainfall to depend on a small set of underlying discrete rainfall states [9, 18]. The state process is assumed to be a Markov chain, and conditional on the state, the emission process (also known as the observation process) is modeled at each location using a mixture of a delta function at zero and Exponential or Gamma distributions for positive precipitation. The Integrated Multi-satellitE Retrievals for Global Precipitation Measurement (GPM-IMERG) dataset [8] is used to develop the model for the months of July to September using data from 2001–2018.

For the Potomac river basin, we fit a 4-state Markov process to estimate the weather states, and use a mixture of two Gamma distributions to model the precipitation amounts at each grid point. We note that while the model is able to replicate the temporal patterns, it underestimates spatial correlation of precipitation between grid points. We extend the model by adding Gaussian copulas to generate correlated precipitation amounts, and verify that the resulting hidden Markov model with Gaussian copulas (HMM-GC) provides an improvement in the spatial correlations of the synthetic data over the existing HMM formulation.

The rest of this report is organized as follows. Section 2 describes the Potomac river basin and the IMERG dataset in detail. Section 3 goes into the theoretical formulation of the HMM and initial hyper-parameter tuning to fit the model, and discusses the need to explicitly specify a spatial correlation structure. The addition of copulas to improve spatial correlation and the resulting HMM-GC is discussed in Section 4. Estimation of the HMM-GC and the properties of the estimated hidden states are further described in Section 5, along with the hardware and software used for computations. Section 6 discusses simulation results from the HMM-GC, and compares model performance against historical data and the synthetic data from the HMM without copulas. The report ends with conclusions and future work on the HMM implementation as well as deep learning approaches in Section 7.

2 Study Area and Precipitation Dataset

We use daily precipitation data for the Potomac river basin located on the East Coast of the USA across West Virginia, Virginia, Pennsylvania, Maryland, and the District of Columbia. Figure 2.1 from NASA's "Blue Marble"[1] provides a sketch of the basin's extent. The watershed is one of the main sources of water for the area; in addition to increasing demand on water, climate variability plays an important role in the region's water supply. In particular precipitation, the main source of water in the Potomac basin, varies inter-

[1]Sourced using the `Basemap` package in `Python` from https://visibleearth.nasa.gov/

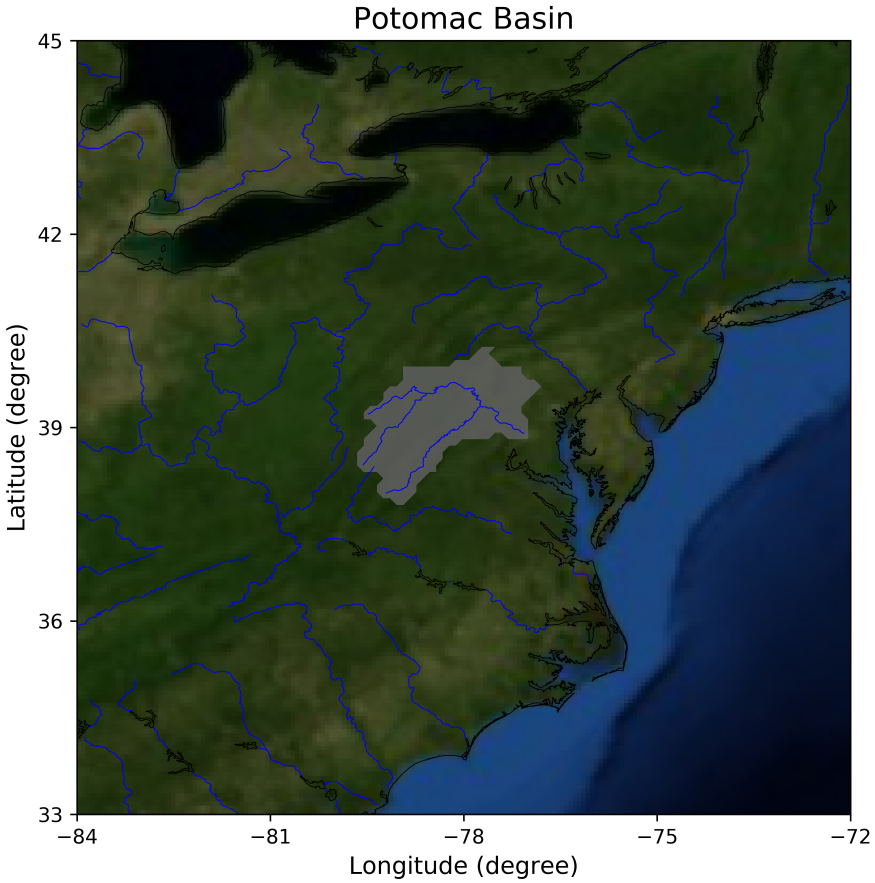

Figure 2.1: Extent of the Potomac river basin on the East Coast of the USA indicated by the gray shape; rivers are represented by blue lines. The Potomac originates at two separate sources in West Virginia and Virginia, and drains into the Chesapeake Bay which connects to the Atlantic ocean on the Eastern side of the map.

annually and makes it challenging to plan for water allocation within the basin. Therefore, understanding seasonal to inter-annual variations in water availability within the basin due to climate variability is important for planning water resources management [14].

The months of April to September constitute the rainy season for the basin, and we have chosen the 92 day period beginning July 1 for the years 2001–2018 for our study. Daily data is used since features of interest like dry and wet spells as well as extreme precipitation events can only be observed in daily data, even though a lot of statistics are eventually presented in monthly and seasonal scales. In comparison, monthly data is often smoothed and are also smaller datasets. July to September overlap with the peak rainy season for the Potomac basin and is part of the same seasonal cycle. Satellite precipitation estimates (SPE) from the GPM-IMERG Level 3 daily dataset [8] is used for the study. The data is available at a $0.1^\circ \times 0.1^\circ$ resolution, which translates to approximately a 10 km $\times$ 10 km grid with 387 grid points spread across the basin. Our observed data, therefore, constitutes a 387 dimensional multivariate time series of length 1656 days.

3 The Hidden Markov Model for Daily Synthetic Precipitation Generation

3.1 Model Formulation

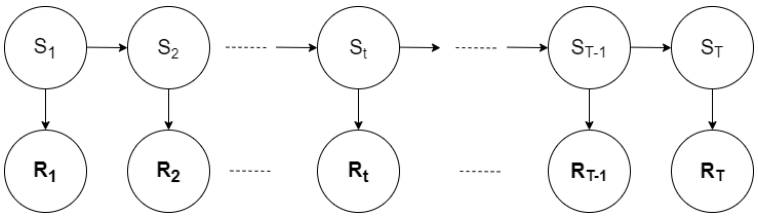

Figure 3.1: Graphical representation of a hidden Markov model

Our hidden Markov model for precipitation generation is based on the work of *Robertson et al.* [18] and *Hughes and Guttorp* [9]. Let $\mathbf{R}_{1:T} = \{\mathbf{R}_1, \ldots, \mathbf{R}_t, \ldots, \mathbf{R}_T\}$, where $\mathbf{R}_t^{'} = (R_t^1, \ldots, R_t^M)$ be the $M \times T$ matrix of precipitation **amounts** for a network of M grid points over T days. Let $S_{1:T} = \{S_1, \ldots, S_t, \ldots, S_T\}$ be the set of hidden (unobserved) weather states, where $S_t \in \{1, \ldots, J\}$. At each location,

$$p[R_t^m = r | S_t = j] = \begin{cases} p_{jm0} & \text{if } r = 0 \\ \sum_{c=1}^{C} p_{jmc} f(r|\alpha_{jmc}, \beta_{jmc}) & \text{if } r > 0 \end{cases} \tag{3.1}$$

with $p_{jmc} \geq 0$ and $\sum_{c=0}^{C} p_{jmc} = 1$ for all $m = 1, \ldots, M$ and $j = 1, \ldots, J$; $f(\cdot|\alpha, \beta)$ is the density function of a Gamma distribution with parameters $\alpha > 0, \beta > 0$. The states arise from a stationary, first-order Markov process, i.e.

$$p(S_1, \ldots, S_T) = p(S_1) \prod_{t=2}^{T} p(S_t | S_{t-1}) \tag{3.2}$$

where $p(S_t|S_{t-1})$ are given by a $J \times J$ stochastic matrix of state transition probabilities $\Pi = ((\pi_{ij}))$, $1 \leq i \leq J$, $1 \leq j \leq J$, and $p(S_1)$ is the initial distribution. Daily rainfall $\mathbf{R}_t$ depends only on the state S_t on day t, i.e.

$$p(\boldsymbol{R}_{1:T} | S_{1:T}) = \prod_{t=1}^{T} p(\boldsymbol{R}_t | S_t), \text{ and} \tag{3.3}$$

$$p(\boldsymbol{R}_{1:T}, S_{1:T}) = \left\{ p(S_1) \prod_{t=2}^{T} p(S_t | S_{t-1}) \right\} \left\{ \prod_{T=1}^{T} p(\boldsymbol{R}_t | S_t) \right\} \tag{3.4}$$

Spatial dependence is captured implicitly by the Markov chain $\{S_t\}$, and the M location components of $\mathbf{R}_t$ are independent of each other given S_t, i.e.

$$p(\boldsymbol{R}_t | S_t) = \prod_{m=1}^{M} p(R_t^m | S_t) \tag{3.5}$$

The model parameters are estimated using the Baum-Welch algorithm [16], a variant of the Expectation Maximization (EM) algorithm. Estimation of the most likely sequence of states using the Viterbi algorithm is discussed in Section 5, which can then be used for simulating data.

3.2 Optimum parameter configuration

There are three aspects to the parameter selection for our study. The first is the number of hidden states that we want in our model. Hidden Markov models have usually used 4-6 states in previous studies, and it has been noted that large number of states are good for simulation whereas a smaller number allows easier physical interpretation of the states [20]. Secondly, the amounts process has conventionally been modeled using a mixture of exponential distributions, though Gamma distributions are also used [1, 15]. Finally, the models usually use a mixture of Exponentials or a single Gamma distribution, but we have considered a mixture of Gamma distributions as well. Model goodness of fit is reported in terms of the Bayesian Information Criterion (BIC) scores of the models, with a lower BIC score indicating a better model fit. However, it has been demonstrated before [2] that the theoretical grounds for the most common likelihood based techniques like BIC fail to hold in the order selection context, and should not be used as the only criterion for model selection.

Table 3.1 lists the BIC scores for different number of states, number of mixture components where the first component is always a delta function at zero, and the mixture distribution for positive precipitation (Gamma and Exponential). We noticed that the Gamma distribution models had consistently better BIC scores (lower is better) compared to their Exponential distribution counterparts. While a 6-state model with 2 components (i.e. a single Gamma distribution) has the lowest BIC, we chose to go with the 4-state, 3 component model with Gamma amounts. This allows for easier interpretations for the states. Using two instead of a single Gamma distribution is also made in the hopes of capturing the tail of the distribution, corresponding to extreme weather events, better.

Table 3.1: BIC Scores for hidden Markov models with 2–6 states, Gamma and Exponential distributions, and with 2 and 3 mixture components

Number of	Gamma distribution		Exponential distribution	
hidden states (J)	C = 2	C = 3	C = 2	C = 3
2	2.194e+06	2.193e+06	2.388e+06	2.221e+06
3	2.058e+06	2.056e+06	2.188e+06	2.073e+06
4	1.991e+06	**1.994e+06**	2.102e+06	2.006e+06
5	1.964e+06	1.969e+06	2.055e+06	1.973e+06
6	1.931e+06	1.943e+06	2.013e+06	1.946e+06

3.3 From weather stations to SPEs

Robertson et al. [17] considered a network of 10 weather stations in NE Brazil and daily data from the February–April wet season between 1975–2002 for their study, and reported the mean of observed Pearson correlation coefficients between stations as 0.248. If we use the same model for the 2001–2018 IMERG data from July to September for the Potomac river basin, the mean and maximum of observed Pearson correlation between grid points are 0.642 and 0.986 respectively. Further, Table 3.2 shows the average absolute difference in pairwise correlations between the IMERG data and the HMM based on Section 3.1, where the first 16 years of data was used for training, and the last 2 years used for the out-of-sample estimates. We note that the original model formulation underestimates the observed correlations by 0.253 for a 4-state model. We can attribute this to the higher spatial resolution of our data.

Most studies that we have come across [6, 9, 18] are based on observed data for a small number of weather stations located irregularly over a large area for their analysis. The largest study we could find was *Holsclaw et al.* [7] with 52 stations. HMMs adequately capture spatial correlations in these situations through the shared daily state, but that is not necessarily the case with satellite precipitation estimates. For example, IMERG data for the Potomac basin comprises 387 grid points at $0.1^\circ \times 0.1^\circ$ resolution. Thus there is a need to explicitly capture the spatial correlation in the data.

An intuitive way to achieve this would involve extending to a multivariate state process where different states at different locations are generated based upon a correlation structure. Alternatively, we could generate correlated precipitation amounts once the daily state has been specified. For a larger area of study, we can justify different weather regimes at different locations, and therefore consider a multivariate hidden state process. However, for the Potomac basin, it is reasonable for all locations to have a common daily state, and therefore we concentrate on generating correlated precipitation amounts by means of a Gaussian copula.

Table 3.2: In-sample and out-of-sample average absolute difference in spatial correlations between pairs of grid points for historical and synthetic data

Number of hidden states (J)	In-sample average absolute difference in spatial correlation	Out-of-sample average absolute difference in spatial correlation
2	0.476	0.378
3	0.348	0.261
4	**0.253**	0.195
5	0.255	0.194
6	0.214	0.171

4 A Hidden Markov Model with Gaussian Copulas (HMM–GC) for Spatial Correlation

For a grid of M locations, there are M(M-1)/2 pairwise location combinations. Further, for each state, the locations have different distributions for precipitation amounts. Therefore, for state j, it is possible to construct an M-variate copula $(Z_j^1, \ldots, Z_j^M)$ and generate daily correlated precipitation amounts $(R_j^1, \ldots, R_j^M)$ using the correlation structure of $\boldsymbol{Z}_j$. Following the work of *Mhanna and Bauwens* [15], we use the estimated Spearman rank correlation $\hat{\rho}_j(k,l)$ of the observed data for state j between locations k and l to capture the corresponding Pearson correlation $\hat{\zeta}_j(k,l)$ using the following relationship for a bivariate Normal distribution [13]:

$$\zeta(k,l) = 2\sin\left[\pi\frac{\rho(k,l)}{6}\right] \tag{4.1}$$

This works since $\rho(k,l)$ is invariant under monotone transforms, and we use a CDF and an inverse CDF transform to transform the observed correlations into copula correlation.

Based on this idea, precipitation over the basin is generated using the following algorithm.

Algorithm 1: Algorithm for multi-site daily precipitation generation

for *For day t from 1:T* **do**
- Generate a random number u^* from a Uniform(0,1) distribution; using the algorithm proposed in *Serfozo (2009)* [19], generate the state $j \in \{1, \ldots, J\}$;
- Generate the vector $\boldsymbol{z_j} = (z_j^1, \ldots, z_j^M)$ from the corresponding multivariate Normal distribution $\boldsymbol{Z}_j$ with mean $\underline{\mathbf{0}}$ and covariance matrix $\boldsymbol{\Sigma}_j$;
- Transform the vector element-wise to $\boldsymbol{u}_j = (u_j^1, \ldots, u_j^M) = \Phi(\boldsymbol{z}_j)$, where $\Phi()$ is the CDF of $\boldsymbol{Z}_j$;
- **for** *For location m in 1:M* **do**
  - Generate a random number u_m from a Uniform(0,1) distribution;
  - Compare u_m against the mixing probabilities $(p_{jm1}, p_{jm2}, p_{jm3})$ as per 3.1 for location m to ascertain whether there will be no rainfall, rainfall from the first Gamma component, or rainfall from the second Gamma component;
  - **if** $u_m < p_{jm1}$ **then**
    - set rainfall for location m as zero;
  - **end**
  - **else**
    - Generate rainfall r_j^m as $r_j^m = \Gamma^{-1}(u_j^m)$ for $m = 1, \ldots M$, where $\Gamma^{-1}()$ is the inverse CDF for the Gamma distribution;
  - **end**
- **end**

end

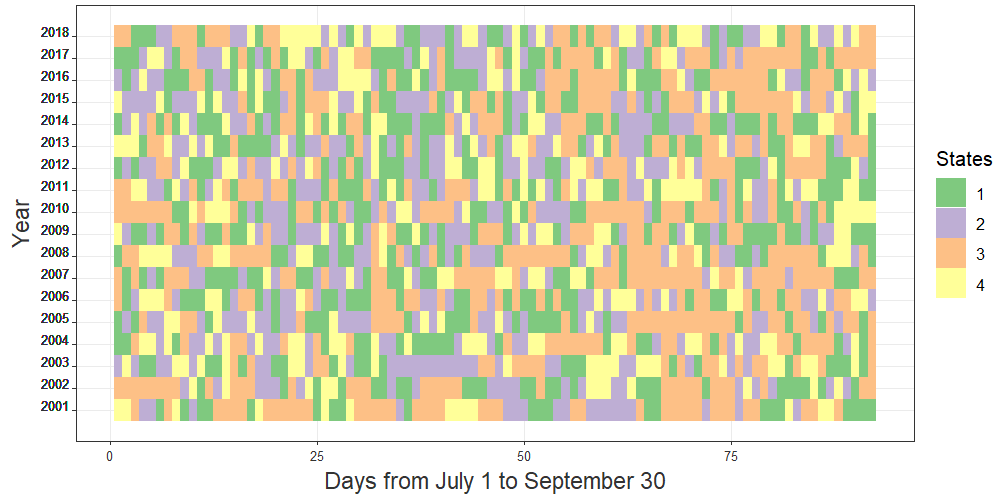

Figure 5.1: Most likely sequence of states for the historical IMERG data from 2001–2018 estimated by the Viterbi algorithm based on a 4 state HMM with 2 Gamma components.

5 Generating Synthetic Precipitation using HMM–GC

5.1 Estimation of parameters

The most probable daily sequence of the 4 states over the 18 year period is estimated using the Viterbi algorithm. The estimates correspond to the maximum a posteriori (MAP) estimates of the states from the data.

Table 5.1 lists the daily and monthly precipitation statistics corresponding to the 4 states over all grid points within the Potomac river basin and the percentage of days that each state appears during the season. We see that state 4 corresponds to heavy precipitation as well as to extreme precipitation events, while state 3 is the driest state. Based on Figure 5.1, state 3 also repeats the most often, allowing the simulation of dry day stretches. States 1 and 2 model intermediate rainfall patterns. Further, the incidence of state 4 reduces from July to September as the rainy season winds down, while the incidence of state 3 correspondingly goes up.

This is further validated by Table 5.2 where we see that the highest transition probability

Table 5.1: Precipitation statistics for the 4 HMM states over all 387 locations within the Potomac river basin

State	Daily mean precipitation (mm)	Daily maximum precipitation (mm)	% days in total	% days in Jul	% days in Aug	% days in Sep
1	0.72	9.82	26.69	27.60	27.60	24.82
2	3.38	11.82	24.52	27.60	27.60	18.15
3	0.03	1.12	30.98	25.45	27.24	40.56
4	14.53	65.6	17.84	19.36	17.56	16.48

Table 5.2: Estimated transition probability matrix for the four-state HMM

	To state			
	1	2	3	4
From state				
1	0.326	0.237	0.285	0.152
2	0.250	0.314	0.188	0.248
3	0.259	0.160	0.524	0.057
4	0.211	0.317	0.143	0.328

in the table is for state 3 to transition back onto itself. Finally, we also note that states 3 and 4 have very low transition probabilities between them, and we usually see states 1 and 2 appear as we transition from heavy rainfall stretches to dry stretches and vice versa.

The Baum-Welch algorithm provides estimates of the mixture probabilties $(p_{jm1}, p_{jm2}, p_{jm3})$, as well the Gamma parameters $(\alpha_{jm1}, \beta_{jm1}), (\alpha_{jm2}, \beta_{jm2})$ at each location for each state. The MAP state sequence is estimated using the Viterbi algorithm. The following procedure is then used to construct the Gaussian copula for each state.

Algorithm 2: Algorithm to calculate Gaussian copula for each state

for *states j in 1:4* **do**
  Subset the days corresponding to state j;
  Calculate the M(M-1)/2 values of $\rho_j(k, l)$ as discussed in section 4;
  Calculate each binormal correlations for the copula using equation 4.1;
  Plug the value in as the *(k,l)* element of the Normal correlation matrix;
  Set diagonal elements of the correlation matrix to be 1;
  Ensure that the resulting matrix is positive definite;
end

Note that the correlation matrix of the copula is also the covariance matrix since we set all diagonal elements to 1. Positive definiteness is ensured by diagonalizing the matrix and replacing all negative eigenvalues with a small positive number, and recalculating the matrix to ensure positive-definiteness is achieved. A more detailed discussion with further references can be found in *Mhanna and Bauwens* [15].

5.2 Hardware and Software

The hardware used in the computational studies is part of the UMBC High Performance Computing Facility (`hpcf.umbc.edu`). The study used CPU nodes with two 18-core Intel Xeon Gold 6140 Skylake CPUs (2.3 GHz clock speed, 24.75 MB L3 cache, 6 memory channels) and 384 GB memory. The nodes are connected by a network of four 36-port EDR (Enhanced Data Rate) InfiniBand switches with 100 Gb/s bandwidth and 90 ns latency.

The precipitation data has been preprocessed using Python scripts developed by Kel Markert for the NASA SERVIR training on hydrologic modeling using VIC. The code is available at `https://github.com/KMarkert/servir-vic-training` and was run using `Python`

2.7.x on taki. The software used for the majority of the hidden Markov model computations is the MVNHMM toolbox [11] developed by Sergey Kirshner and Padhraic Smyth and available at http://www.sergeykirshner.com/software/mvnhmm. The toolbox was developed for Linux and was installed and used on taki. Ancillary scripts written in Python 3.6.x and R 3.6.x were used extensively in the analysis. More details are presented in Section 5.3. All the code written for the project has been uploaded to GitHub at https://github.com/big-data-lab-umbc/cybertraining/tree/master/year-3-projects/team-1.

5.3 Implementation of the code

Within the MVNHMM toolbox, the main functions used in this study were the Baum-Welch algorithm for estimation, the Viterbi algorithm for the generation of most likely estimated states, and the simulation algorithm. Calling any of these actions requires a specialized parameter file. We wrote scripts with Python 3.6.4 for automating the generation of parameter files and running the models with relative ease using batch jobs, and also used the mpi4py library to parallelize the tasks of grid-search for optimum parameters and model estimation and simulation for different locations in parallel. Parallelization allows for the creation and execution of multiple parameter files simultaneously, leading to overall time reductions when running a study consisting of parameter variations. The number of desired parameter files can be requested via the batch scripts used to run the code.

The bulk of the statistical analysis and data generation based on the Gaussian copula was carried out in R 3.6.3. The MVNHMM toolbox was still used to run the Baum-Welch and Viterbi algorithms, since it is the only available software which can fit a mixture emission distribution. A script was written in Python to extract information from the generated parameter files, which was then imported into R for the remaining part of the study. This was necessary since the toolbox does not have a mechanism to simulate correlated emissions or states. All plots were generated using the ggplot2 library in R.

6 Results from the Comparison of Synthetic Data from HMM and HMM-GC with Historical Data

6.1 Spatial correlation in the synthetic data

Figure 6.1 shows box plots of the daily average precipitation amount for the HMM, HMM-GC, and the IMERG data. The low median and interquartile range of HMM and HMM-GC compared to IMERG suggest that both models struggle with capturing spatial correlation to different degrees. We see that the classical HMM for precipitation tends to severely underestimate the correlations between precipitation amounts. There are also negative values which are artifacts of trying to estimate zero correlation in the simulated data. The HMM-GC does a significantly better job of estimating the spatial correlations. As the amount of rainfall decreases from July to September, however, the estimated correlations fall. This can be attributed to the fact that our copula is constructed to capture the correlation between

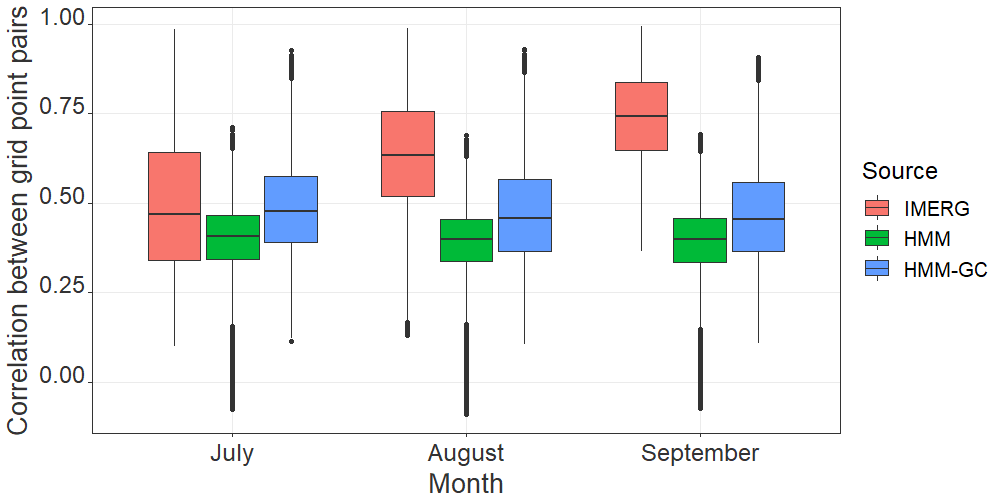

Figure 6.1: Pairwise spatial correlation between grid points for historical IMERG compared with synthetic data from HMM and HMM-GC models based on 18 years of data

positive rainfall amounts, and therefore does not fare as well when trying to capture the correlation of no-precipitation events across the basin.

Figure 6.2 shows the total basin rainfall for the wet season averaged over 18 years of data at the 387 grid points. A visual inspection suggests that the HMM-GC does a better job of simulating spatial patterns within the basin. We believe that larger training sets and longer simulation chains would result in smoother gradients in both plots. We found another issue with extreme values in the regular HMM; 5 of the simulated values were greater than 500 mm, with the largest being over 1500 mm. They have been left out of the plot in the interest of legibility, and are denoted by 5 white grid points within the plot. These values are far higher than the historical data, and probably caused due to the underestimated correlation for the HMM. The HMM-GC is not affected by this problem.

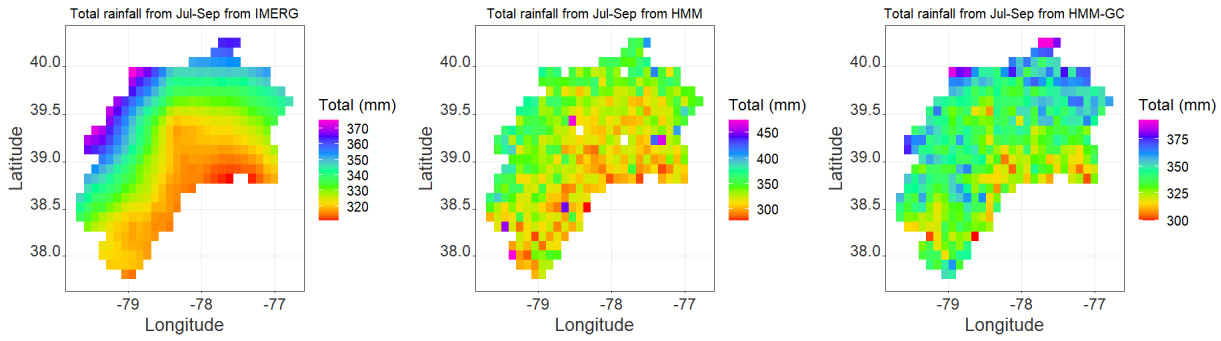

Figure 6.2: Spatial patterns in the total rainfall over the basin from July to September averaged over 18 years of data

6.2 Temporal distribution of the synthetic data

Figures 6.3 and 6.4 display the daily total precipitation amounts over the basin from July to September of 2018. In both plots, the green line shows the same IMERG data. These are compared to the red lines corresponding to the HMM in Figure 6.3 and the HMM-GC in Figure 6.4. The IMERG data in both figures simulates low precipitation events around 5000 mm and extreme precipitation events of up to 15000 mm. When comparing the HMM and HMM-GC data, we see that the HMM-GC can replicate the high precipitation events with daily rainfall values above 10000 mm, much better than the 5000 mm maximum simulated in the HMM data. The failure to simulate extreme precipitation events in the classical HMM formulation may potentially be due to the underestimation of spatial correlations.

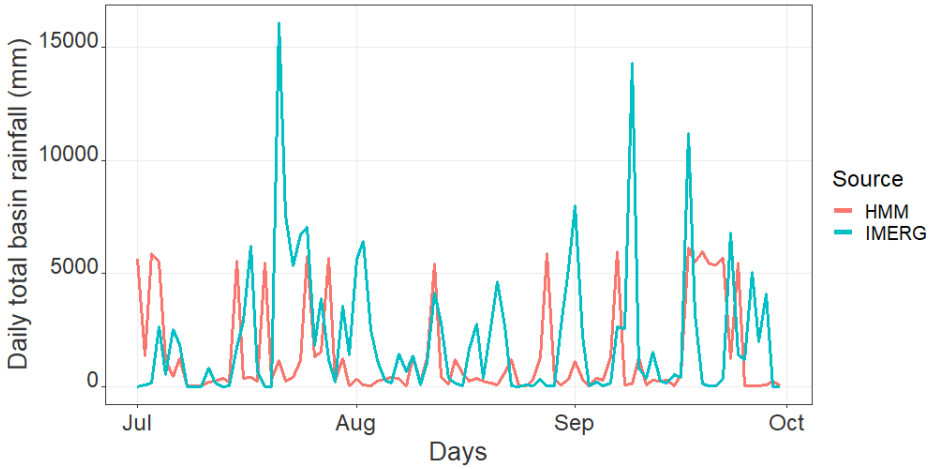

Figure 6.3: Time series of total daily rainfall over the basin in July to September 2018, compared against a single realization from the HMM

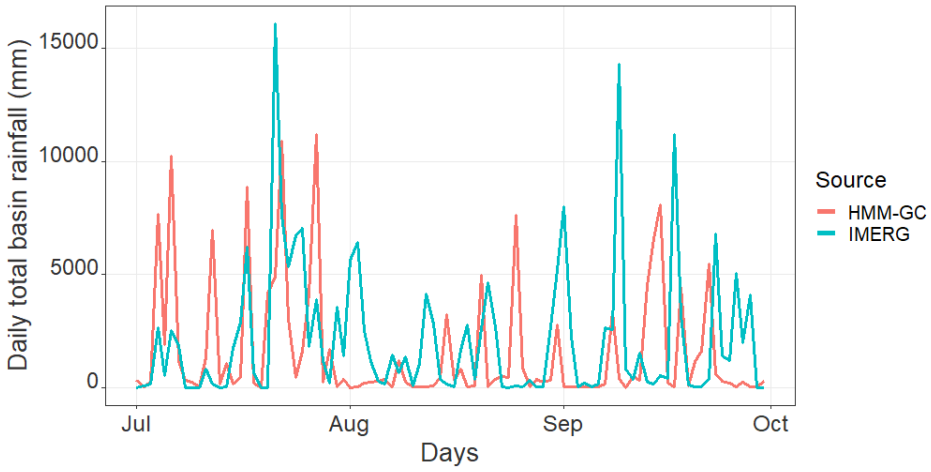

Figure 6.4: Time series of total daily rainfall over the basin in July to September 2018, compared against a single realization from the HMM-GC model

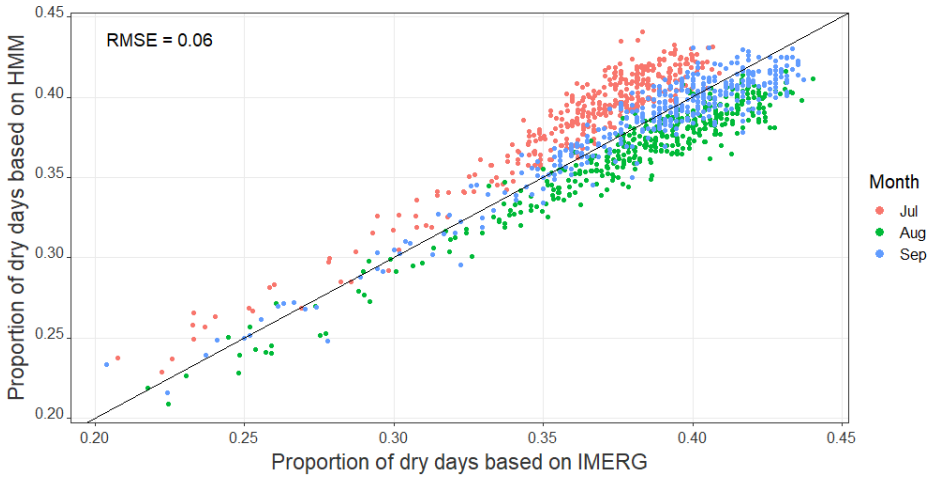

Figure 6.5: Scatterplot of the proportion of dry days per month at each grid point based on historical IMERG data (2001–2018) compared with 100 years of synthetic HMM-GC data

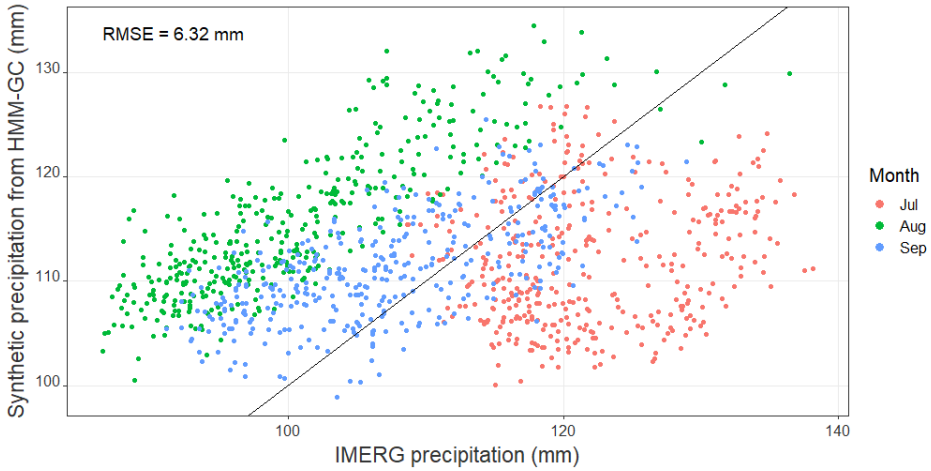

Figure 6.6: Scatterplot of the mean precipitation per month at each grid point based on historical IMERG data (2001–2018) compared with 100 years of synthetic HMM-GC data

While the HMM-GC performs much better at capturing a wider range of daily precipitations, both methods perform generally well in simulating low precipitation events. This conclusion is determined by the similarities in the troughs and peaks of both graphs.

Figures 6.5 and 6.6 represent a comparison of the simulated monthly proportion of dry days and the simulated monthly mean at each location in the basin with the corresponding historical IMERG data. The synthetic data is based on 100 years of simulations from the HMM-GC. A linear relationship between the historical data and the synthetic data can be seen in the plots. The RMSE for the monthly proportions is 0.06, and the RMSE for the monthly means is 6.32 mm, and signifies that the HMM-GC can simulate this three month period well. However, we also note some systemic patterns in the synthetic data, where for some months the model parameters are underestimated and for other months they are

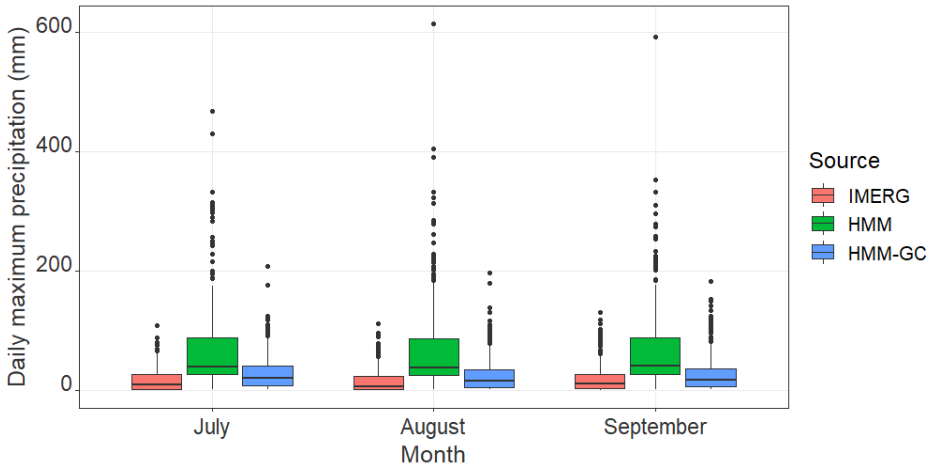

Figure 6.7: Distribution of the maximum daily basin precipitation for historical data from 2001–2018 compared against 18 years of HMM and HMM-GC simulated data

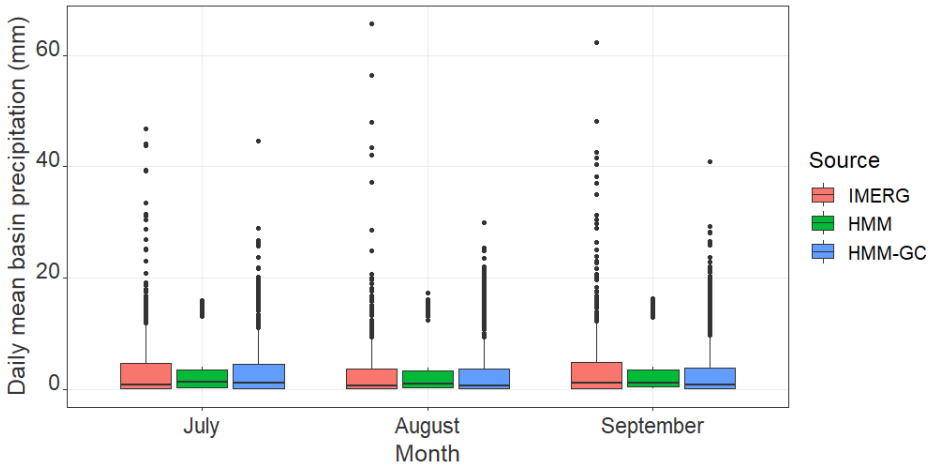

Figure 6.8: Distribution of the average daily basin precipitation for historical data from 2001–2018 compared against 18 years of HMM and HMM-GC simulated data

overestimated. This suggests that the HMM-GC fails to capture at least some features of the data which vary from month to month.

6.3 Extreme precipitation events in the synthetic data

Figures 6.7 and 6.8 plot the distributions of daily maximum and mean basin precipitation for IMERG, HMM and HMM-GC data. We notice in Figure 6.7 that the classical HMM tends to overestimate the daily maximum precipitation, as shown through the long upper tail. However, the short upper tail for the HMM in Figure 6.8 shows that the HMM also underestimates the daily mean precipitation. This can be attributed to the lack of spatial correlation, where some locations simulated very high values, but since they were being

generated independently of the other locations, there was no way to simulate basin-wide consistent behaviour. This has largely been mitigated by the HMM-GC approach; the observation is based on the relative similarities between the HMM-GC and IMERG plots in both Figures 6.7 and 6.8. The daily mean is however still slightly underestimated, suggesting the presence of local extreme weather events within the basin influenced by factors not currently captured by our states.

7 Conclusions and Future Work

7.1 Conclusions

In this paper, we applied a hidden Markov model to remote sensing data from GPM-IMERG over the Potomac river basin and examined how the synthetic data produced by the model compares to historical data described in Section 2. Initial studies based on the original formulation (Section 3) suggested the model falls short in adequately capturing the spatial correlations of the gridded historical data. To address this issue, we extended the original HMM by adding Gaussian copulas to the emissions process, and the corresponding simulation and estimation process for this HMM-GC formulation is discussed in Sections 4 and 5.

For spatiotemporal models of multi-site daily precipitation like the HMMs in this paper, there are three major data features we want to capture and replicate in synthetic data: spatial correlation between locations in the region, rainfall amounts and occurrences for long periods of time, and the extreme weather events of the region. In Section 6, we discussed model performance for the HMM and HMM-GC based on these metrics.

Figures 6.1 and 6.2 demonstrate that the addition of the Gaussian copulas in the HMM (HMM-GC) results in more realistic spatial correlations compared to the HMM, though the HMM-GC still underestimates it. Figures 6.3 and 6.4 show that both the HMM and HMM-GC adequately model low precipitation events, which is captured largely through the hidden states common to both models, discussed in section 5. A key difference lies in how each model can replicate extreme precipitation events, where the HMM-GC does a superior job as the high spatial correlations allow simultaneous basin-wide simulation of extreme weather events better than the HMM. This is further displayed in Figure 6.8, where the HMM-GC has more variability and a longer upper tail than the HMM, even though Figure 6.7 shows that the HMM actually overestimates the maxima.

These results suggest that the HMM-GC improves the default HMM's ability to capture spatial correlations between locations in the region, the rainfall amounts and occurrences for long periods of time, and the extreme weather events of the region. However, there is still room for improvement in several aspects of the HMM-GC. Figures 6.5 and 6.6 demonstrated that despite low RMSE values when simulating the July-September time period as a whole, the direction of estimation errors depends on the month. This signifies information in the historical data we have failed to capture. Furthermore, significant improvements to the spatial correlation can still be made as shown in Figures 6.1 and 6.2. The results suggest that improvements in the spatial correlation could influence model performance as a whole.

7.2 Future work

Our immediate next step is documenting our code as well as publishing it on GitHub[2] so that any researcher working in this field can take advantage of it. While the code is currently in a mix of Python and R, we plan to convert the entire code to Python because it has a much wider user base. We realized the need for this while working on the project, since other than the MNVHMM toolbox there are not a lot of software resources for precipitation modeling using HMMs. We consider our work to be an extension to the existing toolkit that allows more flexibility in modeling as well as takes advantage of `MPI` if there is a need to scale.

There are also aspects of the current study that can be explored in more detail. The HMM-GC can itself be fine tuned further; since one of our goals was to compare with the current HMM, we preferred a selection of parameters close to what is suggested in existing work. We believe that model performance can be further improved by exploring the parameterization suggested by the BIC scores. The copula approach becomes untenable if we use larger basins for our study, where the parameters might not be estimable due to the size of the dataset, since IMERG data only dates back to 2000 for now. For larger areas, the underlying idea of there being a shared state also becomes difficult to justify; we believe we are already seeing signs of this in our study, where some parameters are not estimated well enough for some months since different parts of the basin have slightly different weather patterns. Expanding out from the shared state paradigm is thus an avenue we want to explore. There is also a need to do a similar study for the drier periods of the year and check the model performance and draw inference on times of the year this sort of a model is most suited for, and the duration of time it can simulate within its framework.

Finally, there is scope for comparing this model with the Wilks approach [15, 21], which is another method where copulas are used to explicitly specify spatial correlations. While the underlying model formulation is quite different, we believe both models can benefit from a comparison study of their strengths and weaknesses. A preliminary study comparing of the two methods for the same IMERG data can be found in [12]. Deep Learning approaches should also be explored for the problem of synthetic precipitation generation, and we close out with an overview of future research directions based on deep learning literature.

One way to leverage deep learning techniques is by using a Neural network to simulate wet/dry sequences instead of simulating rainfall directly as shown in [3]. This works by using a non-parametric approach which is implemented by a multi-layer perceptron (MLP) neural network architecture, wherein the model fits two curves which are the cumulative distribution functions (CDF) of dry and wet sequences. This neural network consists of an input layer containing one input neuron and output neuron. However, the neurons in the hidden layer are optimized by the model. In the training phase, an MLP is optimized for each month at each station. During this phase, the MLP is made to smoothly fit the CDF curve by interpolating the curves and approximating the distribution. By using a Bayesian approach, the MLP is being penalized with a bigger architecture to avoid over-fitting. In order to simulate wet/dry sequences at each station, the model generates a random number between 0 and 1 and inserts into the input layer, with the output being the length of the wet/dry

[2] https://github.com/big-data-lab-umbc/cybertraining/tree/master/year-3-projects/team-1

spell. Generating rainfall is simulated the same way wet/dry sequences are instead only using the wet spell. After the random number is generated and entered into the neural network, the model produces an output which is the amount of daily rainfall. This formulation has an inherent shortcoming in that it has to model months separately and is thus not very good at simulating long chains even though they are modeling wet/dry stretches. A larger problem, however, is that it considers locations to be unconditionally independent of each other, which we have seen to be inadequate for gridded remote sensing data. This calls for a deep learning approach which can accommodate a spatial structure as well.

As discussed in [4], Recurrent Neural Networks (RNNs) are neural networks specifically designed to deal with time series data. Given time series data, an RNN can make accurate predictions and classification using the technical ordering of the provided data. However, traditional RNNs are not designed to look at spatially relevant details like a convolution neural network might [10]. In some applications the spatial properties are entirely removed in order for the LSTM to be used. Recent research into convolutional RNNs has created a neural network type which is capable of using both temporal and spatial properties to make accurate predictions [10]. Such a neural network would be able to use and learn all of the complex features present in our data. This would give us the basis for a network which can handle our data but we would still need additional techniques to produce a network capable of robust data generation. We could take the proposed ConvRNN and use a similar adversarial model proposed in [5] to create generative adversarial networks (GANs). Both the generator and the discriminator would be variations comprised of ConvRNNs. The trained generator would create new time series data given a noise vector whereas the trained discriminator would be capable of determining whether provided time series data is natural occurring or artificially generated. All of these concepts together allow us to create a generator which is capable of producing robust spatially relevant time series data.

Acknowledgments

This work is supported by the grant CyberTraining: DSE: Cross-Training of Researchers in Computing, Applied Mathematics and Atmospheric Sciences using Advanced Cyberinfrastructure Resources from the National Science Foundation (grant no. OAC–1730250). Co-authors Gerson Kroiz, Jonathan Basalyga, Uchendu Uchendu were supported through an REU Supplement to this grant. Co-author Gerson Kroiz was also supported through an Undergraduate Research Award (URA) from UMBC. The hardware in the UMBC High Performance Computing Facility (HPCF) is supported by the U.S. National Science Foundation through the MRI program (grant nos. CNS–0821258, CNS–1228778, and OAC–1726023) and the SCREMS program (grant no. DMS–0821311), with additional substantial support from the University of Maryland, Baltimore County (UMBC). See hpcf.umbc.edu for more information on HPCF and the projects using its resources. Co-author Reetam Majumder was supported by JCET and as HPCF RA. Co-author Carlos Barajas also acknowledges support as HPCF RA.

References

[1] E. Bellone, J. Hughes, and P. Guttorp. A hidden Markov model for downscaling synoptic atmospheric patterns to precipitation amounts. *Clim. Res.*, 15(1):1–12, 2000.

[2] Enrica Bellone. Nonhomogeneous hidden Markov models for downscaling synoptic atmospheric patterns to precipitation amounts. Ph.D. Thesis, Department of Statistics, University of Washington, 2000.

[3] Jean-Philippe Boulanger, Fernando Martinez, Olga Penalba, and Enrique Segura. Neural network based daily precipitation generator (NNGEN-P). *Clim. Dynam.*, 28:307–324, 2007.

[4] J. T. Connor, R. D. Martin, and L. E. Atlas. Recurrent neural networks and robust time series prediction. *IEEE Transactions on Neural Networks*, 5(2):240–254, 1994.

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[6] A. M. Greene, A. W. Robertson, and S. Kirshner. Analysis of Indian monsoon daily rainfall on subseasonal to multidecadal time-scales using a hidden Markov model. *Q. J. Roy. Meteor. Soc.*, 134(633):875–887.

[7] Tracy Holsclaw, Arthur M. Greene, Andrew W. Robertson, and Padhraic Smyth. A Bayesian hidden Markov model of daily precipitation over South and East Asia. *J. Hydrometeorol.*, 17(1):3–25, 2016.

[8] G. J. Huffman, E. F. Stocker, D. T. Bolvin, E. J. Nelkin, and Jackson Tan. GPM IMERG final precipitation L3 1 day 0.1 degree × 0.1 degree V06, 2019. Edited by Andrey Savtchenko, Greenbelt, MD, Goddard Earth Sciences Data and Information Services Center (GES DISC), https://disc.gsfc.nasa.gov/datasets/GPM_3IMERGDF_06/summary, accessed on June 25, 2020.

[9] J. P. Hughes and P. Guttorp. Incorporating spatial dependence and atmospheric data in a model of precipitation. *J. Appl. Meteorol.*, 33:1503–1515, 1994.

[10] Gil Keren and Björn Schuller. Convolutional RNN: an enhanced model for extracting features from sequential data, 2016.

[11] Sergey Kirshner. Modeling of multivariate time series using hidden Markov models. Ph.D. Thesis, University of California, Irvine, 2005.

[12] Gerson C. Kroiz. *A Comparison of Stochastic Precipitation Generation Models for the Potomac River Basin*. Senior Thesis, Department of Mathematics and Statistics, University of Maryland, Baltimore County, 2020.

[13] William H. Kruskal. Ordinal measures of association. *J. Am. Stat. Assoc.*, 53(284):814–861, 1958.

[14] Reetam Majumder, Redwan Walid, Jianyu Zheng, Carlos Barajas, Pei Guo, Chamara Rajapakshe, Aryya Gangopadhyay, Matthias K. Gobbert, Jianwu Wang, Zhibo Zhang, Kel Markert, Amita Mehta, and Nagaraj K. Neerchal. Assessing water budget sensitivity to precipitation forcing errors in Potomac river basin using the VIC hydrologic model. Technical Report HPCF–2019–11, UMBC High Performance Computing Facility, University of Maryland, Baltimore County, 2019.

[15] M. Mhanna and W. Bauwens. A stochastic space-time model for the generation of daily rainfall in the Gaza Strip. *Int. J. Climatol.*, 32:1098–1112, 2012.

[16] Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications. 1989.

[17] A. W. Robertson, S. Kirshner, and P. Smyth. Downscaling of daily rainfall occurrence over Northeast Brazil using a hidden Markov model. *J. Climate*, 17:4407–4424, 2004.

[18] A. W. Robertson, S. Kirshner, P. Smyth, S. P. Charles, and B. C. Bates. Subseasonal-to-interdecadal variability of the Australian monsoon over North Queensland. *Q. J. Roy. Meteor. Soc.*, 132:519–542, 2006.

[19] Richard Serfozo. *Basics of Applied Stochastic Processes, Springer 2009.* 01 2009.

[20] Augustin Touron. Modeling rainfall using a seasonal hidden Markov model, 2017.

[21] D. S. Wilks. Multisite generalization of a daily stochastic precipitation generation model. *J. Hydrol.*, 210(1–4):178–191, 1998.