Deep Recurrent Learned Dynamic Downscaling

Jean-Yves Djamen

Supervisor: Johanna Nešlehová

Co-Supervisor: Alexis Hannart

2022

Abstract

Global climate models represent major climate system components of the planet in order to generate long term, sparse, accurate realizations of future climatic events across the entire globe. Downscaling is the method by which these low resolution realizations are converted into high resolution simulations of climate events which can then be used by stakeholders and policy makers.

Regional climate models dynamically downscale simulated climate by conditioning global climate models on location-specific physical processes. Although these models are robust and reliable, they are computationally expensive when compared to statistical approaches for modeling a general relationship between global climate behaviour and local climate behavior. Therefore, there is need for downscaling methods that leverage the computational efficiency of statistical models while maintaining the performance of regional climate models.

In this thesis, we build upon previously proposed deep learning methods for dynamical downscaling through estimation of a regional climate model. Our proposed model is a generative adversarial network that leverages the effects of temporal dependencies within spatio-temporal climate events.

Résumé

Les modèles climatiques mondiaux reproduisent les principaux composants du système climatique de la planète afin de générer des réalisations à long terme, éparses et précises, d'événements climatiques futurs sur l'ensemble du globe. La réduction d'échelle est la méthode par laquelle ces simulations à basse résolution sont converties en simulations à haute résolution d'événements climatiques qui peuvent ensuite être utilisées par les parties prenantes et les décideurs.

Les modèles climatiques régionaux réduisent dynamiquement l'échelle du climat simulé en conditionnant les modèles climatiques mondiaux à des processus physiques spécifiques à un lieu. Bien que ces modèles soient robustes et fiables, ils sont coûteux en termes de calcul par rapport aux approches statistiques permettant de modéliser une relation générale entre le comportement du climat mondial et le comportement du climat local. Par conséquent, il existe un besoin pour des méthodes de réduction d'échelle qui tirent parti de l'efficacité de calcul des modèles statistiques tout en maintenant la performance des modèles climatiques régionaux.

Dans cette thèse, nous nous appuyons sur les méthodes d'apprentissage profond proposées précédemment pour la réduction d'échelle dynamique par l'estimation d'un modèle climatique régional. Le modèle que nous proposons est un réseau antagoniste génératif qui exploite les effets des dépendances temporelles dans les événements climatiques spatio-temporels.

Acknowledgements

First, I would like to thank professor Nešlehová for inspiring and mentoring me during both my undergraduate and graduate studies. Second, I would like to thank Alexis Hannart for his technical contributions to this thesis. I would also like to thank IVADO for providing the funding for this work. Lastly, I would like to thank my friends and family for their support.

Contents

List of Figures

1 Introduction

There is a clear symbiotic nature to the relationship between the planet's inhabitants and its climate. As our populations develop and grow, we make use of Earth's resources, often through means that negatively impact our atmosphere. This in turn leads to drastic changes in our environment such as the endangerment of vulnerable species [25], an increase in the frequency of extreme climate events [44], and the creation new weather patterns [70]. In the quest of adapting to the changing climate, the ability to predict hourly, daily, seasonal or annual climate events provides advantages and benefits in the development and protection of our agriculture, our early warning systems, our health infrastructure, and our response to climate change. In order to predict large scale climate features, geophysicists have developed *Global Climate Models* (GCMs) which are numerical models that use the physical properties and behavior of the Earth's various components such as the atmosphere, the ocean, the cryosphere and the land surface. These models are able to simulate long term predictions of various climate components over coarse resolutions of space (250-600 km).

Based on equations representing the physical principles that govern the interactions and processes of the planet, the GCMs generate reproductions of key components of the climate system [19]. From these models, the general behaviour of greenhouse gas concentrations, temperature, rainfall and other climate attributes can be estimated over a wide time horizon but at a relatively coarse scale.

Due to the spatial and temporal sparsity of the simulations, relevant stakeholders and practitioners cannot directly make use of the generated

patterns which challenges the saliency of the output of GCMs [19]. In fact, several numerical models produce predictions for climate fields that are too coarse in resolution for immediate practical application [15]. In order to fully harness the utility of GCMs, we must use methods to recover the relationship between large scale climate patterns and region specific climate behaviour.

The process of converting large-scale climate features into finer-scale local climate features is called *downscaling*. Spatial downscaling can be used to recover detailed region-specific climate features from coarse GCM output while temporal downscaling can be used to extrapolate climate behaviour at a finer resolution than the available GCM output.

For example, yearly simulations of rain in Canada can be spatially downscaled to yearly simulations of rain in Montreal, temporally downscaled into daily predictions of rain in Canada, or both.

As can be expected, due to the versatility of downscaling, there have been many approaches developed to tackle this problem. Dynamic and statistical downscaling are two of the most popular methods of downscaling. Statistical downscaling aims to recover a meaningful statistical relationship between coarse GCM output data and local climate behavior. In contrast, dynamical downscaling is usually done though the application of regional climate models which make use of regional information and physical processes to increase the resolution of dynamic weather models.

This thesis explores learned dynamical downscaling through temporal climate fields in the context of precipitation. Although we focus on the application of deep learning methods, a survey of statistical and machine learning methods for downscaling is also provided.

2 Data and Notation

In this chapter, we first define uniform notation for climate variables used in this thesis. The described data notation will be used to construct general downscaling models from low resolution data to high resolution data. This is followed by the detailing and provenance of climate data used our experiments. All diagrams in this and subsequent chapters are original creations generated through either Microsoft Powerpoint or `https://madebyevan.com/fsm/`.

2.1 Notation

Consider the set of spatial locations $\mathcal{S} = \{s_l : l = 1, \ldots, L\} \subseteq \mathbb{R}^2$. Let $\mathcal{V}$ be the set of all climate variables. Finally, consider the set of time points

$$\mathcal{T} = \{t = (y, m, d, h) : 1 \leq h \leq 24, 1 \leq d \leq 31, 1 \leq m \leq 12, y \in \mathbb{N}\}$$

Here y represents years, m represents months, d represents days, and h represents hours. We consider an ordering on $\mathcal{T}$ identical to the ordering on calendars which amounts to the lexicographic ordering of the elements of $\mathcal{T}$. For each climate variable $v \in \mathcal{V}$, we define the high resolution measurement at each location $s_j \in \mathcal{S}$ taken at time $t \in \mathcal{T}$ as $Y_{v,t}(s_j)$. Similarly, we define the low resolution measurement of climate variable $v \in \mathcal{V}$ at each location $s_j \in \mathcal{S}$ taken at time $t \in \mathcal{T}$ as $X_{v,t}(s_j)$. Often considered are the mean, minimum and maximum of precipitation and temperature measured hourly, daily, or annually.

Let $\mathcal{P}$ and $\mathcal{W}$ be two ordered partitions of our time space $\mathcal{T}$ of finite

length defined as

$$\mathcal{P} = \{\varrho_1, \varrho_2, \ldots\} \qquad \mathcal{W} = \{\tau_1, \tau_2, \ldots\}$$

With the additional condition that

$$\varrho < \tau \qquad \forall \varrho \in \mathcal{P}, \tau \in \mathcal{W}$$

Usually, we will consider climate variables indexed by $\mathcal{P}$ and $\mathcal{W}$ as these denote past and future realizations of climate variables respectively.

The resolution of coarse variables will be defined as $N_X \times M_X$ and the resolution of high resolution variables will be defined $N_Y \times M_Y$. In order to downscale, we must define the same set of locations for for high resolution and low resolution climate variables. Since X represents the distribution of a coarse global variable, for each variable v, certain neighboring points in space $\mathcal{S}$ at the same time index t will share the same measurement. In contrast, the distribution of the high resolution climate variable will have more texture and variability.

In the data analyzed in this thesis, the spatial location is associated with a pixel on a grid. In order to have the same space $\mathcal{S}$ for both high and low resolution climate variables, the measurements of the coarse variable are repeated over certain pixels in the high-resolution grid, as illustrated in Figure 1.

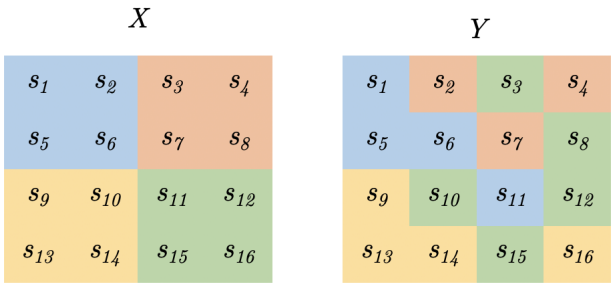

Figure 1: Color representations of low resolution (X) and high resolution (Y) distributed over space $\mathcal{S}$. X is represented by a 2×2 grid whereas Y is represented by a 4×4 grid. $N_X = M_X = 2$ and $N_Y = M_Y = 4$.

Recall that coarse, low resolution data is generated by GCMs and can cover hundreds of kilometers. While low resolution dynamic downscaling models can generate simulations of climate with finer resolution than GCMs, the resolution is still too coarse for a stakeholder use. This is the case for the low resolution data used in the models in this thesis. High resolution data can represent local climate station observations of climate or data simulated by a high resolution dynamic downscaling model.

2.2 Data

In this thesis, the Weather Research and Forecasting model version 3.3.1 is used to generate climate data. This regional climate model is used twice to simulate climate data over the contiguous United States at 12 km (fine) and 50 km (coarse) resolution over 3 hour intervals. Samples from these simulations can be seen in Figure 2. The coarse and fine resolutions of

data correspond to $N_X \times M_X$ =128×64 and $N_Y \times M_Y$=512×256 grid cells, respectively.

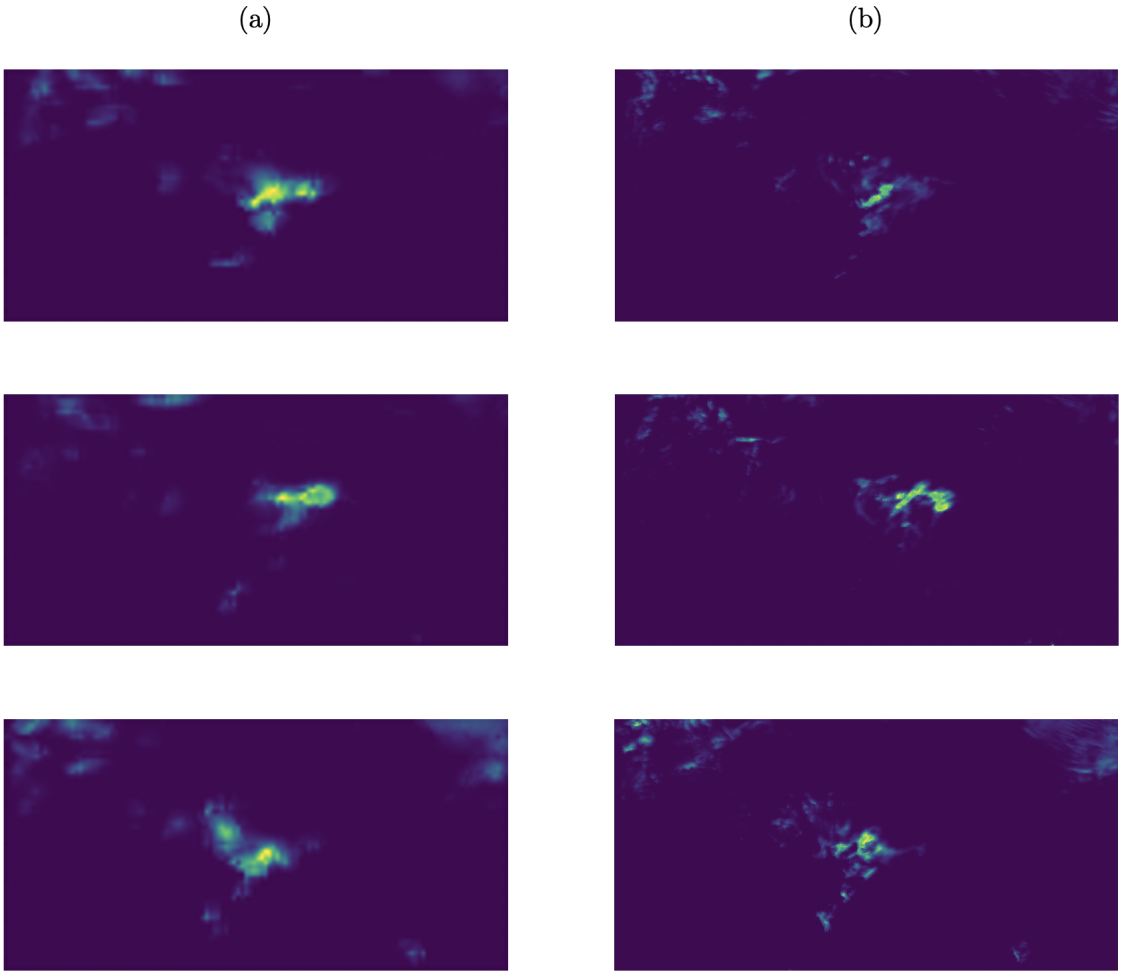

Figure 2: Low resolution (a) and corresponding high resolution (b) simulations of precipitation by the Weather Research Forecasting model.

This data is analyzed in [61] where it is used to learn a mapping between low resolution and high resolution regional climate models. Because of the

structure of the RCM, the difference in spacial resolution propagates into slight differences in spatial coverage, time scale, and simulated precipitation [61]. This means that, ideally, the two different resolutions should not be seen simply as high resolution and low resolution versions of the same simulation of a given climate variable but rather as slightly different simulations of the same climate variable at different resolutions. The difference between these two is a substantial challenge to the modeling approach used.

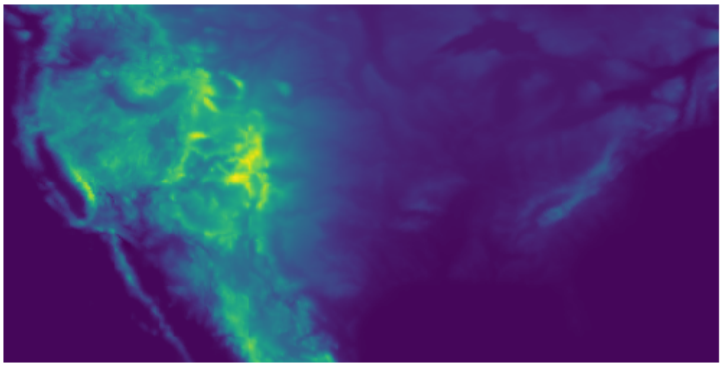

Figure 3: Elevation map of a region of interest (contiguous United States)

To recover the magnitude and variability of precipitation, other variables have also been collected that have shown high correlation with precipitation. Such climate variables are integrated water vapour (IWV), simulated sea level pressure (SLP), and simulated air temperature (T2). The topographical elevation map (TOP) of the region of interest is also included and is pictured in Figure 3.

Variable	Notation	Data Range	Encoding Range
50 km precipitation (mm/3 hr)	x_{precip}	[0.05, 13.62]	[0,27.85]
50 km sea level pressure (hPa)	x_{SLP}	[990.97, 1039.34]	[0,1044.66]
50 km integrated water vapour (cm)	x_{IWV}	[1.56, 116.46]	[0,136.19]
50 km temperature (K)	x_{T2}	[241.75, 310.35]	[0,318.88]
12 km topography (m)	x_{TOP}	[0, 3204.51]	[-63.07,3556.77]
12 km precipitation(mm/3 hr)	y_{precip}	[0.05, 15.66]	[0,31.62]

Table 1: Data Summary

Simulations of precipitation at 3 hour interval data have been provided for this thesis by the authors of [61]. In their article, precipitation is clipped between 0.05 mm/3hr and the 99.5^{th} percentile of rain at all locations in order to avoid light drizzles or extremes. The latter is unfortunate because any model developed with this data will be unable to properly recover the relationship between high and low resolution simulations of extreme precipitation.

For each climate variable, the value at each grid cell is numerically encoded as can be seen in Table 1. This encoding is to facilitate the use of deep learning models built for image processing. Unfortunately, only the encoded data is made available by the authors of [61].

3 Literature Review

The two most popular approaches to downscaling are dynamic downscaling and statistical downscaling. Dynamic downscaling is done through numerical models that use the output of GCMs as well as local regional structure and boundaries to simulate spatially downscaled climate data at an increased resolution. Statistical downscaling attempts to learn a mapping between historical observed climate data at a coarse resolution and historical observed climate data at a fine resolution. This mapping is then applied to GCM outputs to form downscaled high resolution simulations of data.

Dynamic downscaling tends to be complex and computationally intensive as the numerical model is autoregressive in nature and must sequentially generate simulations. In contrast, statistical downscaling can directly use the learned mapping on any GCM simulation. However, compared to dynamic downscaling approaches, statistical downscaling tends to have difficulties fully recovering the variability of climate events since statistical downscaling models are often built around assumptions of stationarity in the relationship between high resolution and low resolution simulations of climate variables [56].

Some approaches to downscaling involve both of these approaches in tandem to mitigate their flaws. The three methods of downscaling discussed in this thesis are depicted in Figure 4. For instance, statistical downscaling can be done to learn a mapping between GCM simulations and regional climate simulations. This creates a statistical model that can replicate the efficacy and accuracy of a dynamic downscaling model while leveraging computational efficiency of statistical models.

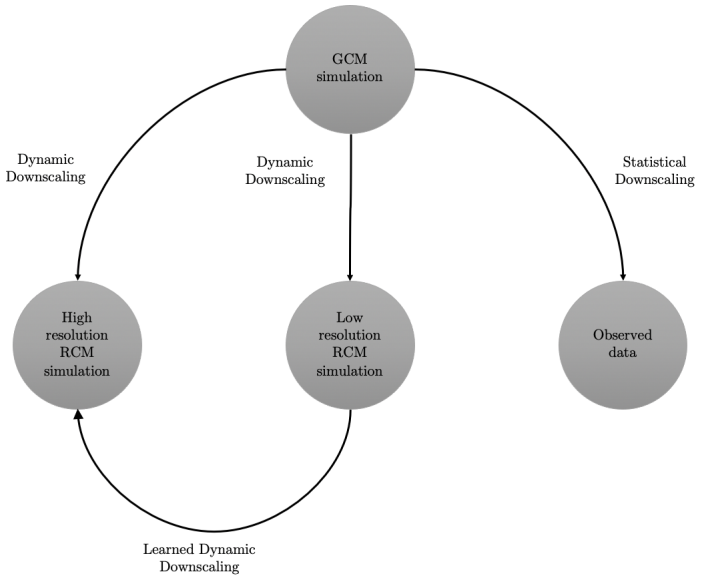

Figure 4: Downscaling Modes

3.1 Dynamic Downscaling

Following the assumption that global climate affects local climate, dynamic downscaling uses explicit physical processes specific to a location along with global simulations of weather to generate local simulations of weather.

Definition 1 (Regional Climate Models)**.** Regional climate models (RCMs) are numerical weather models that simulate local high resolution climate behaviour following the constraints of coarse resolution GCM simulation.

RCMs are completely defined by the set regional boundary conditions and the chosen methods to model local weather physics. As a result of this, in practice, RCMs have displayed certain shortcomings. First, RCMs are extremely sensitive to region size and location [49, 66]. Second, since RCMs are numerical weather models nested within a GCM, the initial boundary

conditions from the GCM tend to cause induced model variability within the RCM [49, 40]. In essence, the error in variance induced from the GCM propagates to the RCM. Finally, since RCMs are time-varying models, downscaling GCM simulations in the distant future can be computationally expensive when compared to alternative methods of downscaling [49, 66, 56].

3.2 Statistical Models for Downscaling

In this section, we will cover common statistical models that have been used for downscaling. For a given location s_0, in order to generate downscaled estimates of climate variable $\{\hat{Y}_{i,t}(s_0) : t \in \mathcal{W}\}$, statistical models aim to use coarse resolution climate variables $\{X_{v,t}(s_0) : v \in \mathcal{V}, t \in \mathcal{P} \cup \mathcal{W}\}$ and past realizations of high resolution climate variable $\{Y_{i,t}(s_0) : t \in \mathcal{P}\}$.

3.2.1 Delta Method

The delta method is one of the simpler methods of downscaling based on bias correction and only makes use of the coarse measurements of the target variable. This approach relies on the assumption that the temporal change in coarse climate variables generated by GCMs is the same as the temporal change in local climate variables. Following this assumption, the delta method forms estimates of the temporal change within the simulated coarse climate variables and creates downscaled local climate variables that follow the same temporal change [24].

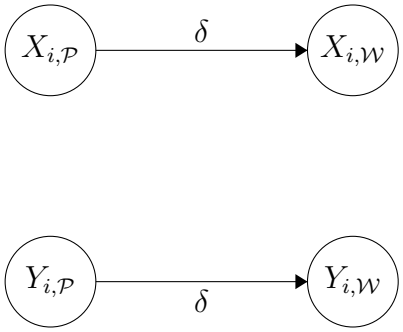

Figure 5: Assumption of the Delta method. δ is the shared linear relationship between past and future climate events.

Example 1 (Delta Method)**.** Let $\{X_{precip,t}(s_l) : t \in \mathcal{P} \cup \mathcal{W}\}$ and $\{Y_{precip,t}(s_l) : t \in \mathcal{P}\}$ be low and high resolution monthly measurements of precipitation at location s_l. The delta method can be used to spatially downscale monthly simulations of low resolution precipitation $\{X_{precip,t}(s_l) : t \in \mathcal{W}\}$ into high resolution precipitation $\{\hat{Y}_{precip,t}(s_l) : t \in \mathcal{W}\}$. Since we are considering monthly measurements, we can reconstruct the time index from (y, m, d, h) to (y, m). For ease of notation, we define $\mathcal{P}_{yr}$ and $\mathcal{W}_{yr}$ to be the number of years considered in past and future realizations respectively. Lastly, assume that $\mathcal{P}_{yr} = \mathcal{W}_{yr}$. In order to downscale GCM data from a simulated future low resolution sequence of monthly precipitation, the delta change parameter for each month m at location s_0 is estimated as

$$\delta_m^l = \frac{\frac{1}{\mathcal{W}_{yr}} \sum_{y:(y,m)\in\mathcal{W}} X_{precip,(y,m)}(s_l)}{\frac{1}{\mathcal{P}_{yr}} \sum_{y:(y,m)\in\mathcal{P}} X_{precip,(y,m)}(s_l)}$$

As a ratio of means, this gives a measure of the relative difference between the monthly precipitation in previously simulated climate and the monthly precipitation in future simulations. As illustrated in Figure 5, the primary

assumption of the delta method is that this change in monthly averages for coarse climate variables is reflected locally. From this assumption, the downscaled high resolution climate variables are estimated as

$$\{\hat{Y}_{precip,t}(s_l) : t \in \mathcal{W}, s_l \in \mathcal{S}\} = \{\delta_m^l Y_{precip,t}(s_l) : t = (m, y) \in \mathcal{P}, s_l \in \mathcal{S}\}$$

This generates spatially downscaled sequence of estimated monthly precipitation of a future unobserved sequence. $\bigtriangledown$

It should be noted that some applications of the delta method adopt an additive rather than multiplicative model [27]. The additive delta method is often used for temperature while the multiplicative approach is used for precipitation. The multiplicative delta method avoids negative estimated local precipitation whereas the additive delta method ensures negative temperatures are covered by the model. Following Example 1 using monthly temperature rather than precipitation, the monthly delta change parameter would become

$$\delta_m^l = \frac{1}{\mathcal{W}_{yr}} \sum_{y:(y,m)\in\mathcal{W}} X_{temp,(m,y)}(s_l) - \frac{1}{\mathcal{P}_{yr}} \sum_{y:(y,m)\in\mathcal{P}} X_{temp,(m,y)}(s_l)$$

The downscaled estimates of a new sequence can then be recovered through the function below:

$$\{\hat{Y}_{temp,t}(s_l) : t \in \mathcal{W}, s_l \in \mathcal{S}\} = \{\delta_m^l + Y_{temp,t}(s_l) : t = (m, y) \in \mathcal{P}, s_l \in \mathcal{S}\}$$

One of the main advantages of the delta method is its simplicity. Compared to RCMs, the computations required to build a delta method downscaling model are quickly obtained. Additionally, this model preserves the local climate variability which tends to be one of the weaker aspects of statistical

downscaling models. This conservation of variance arises from the fact that downscaled values for local climate are a simple linear transformation of previously observed climate. This means the variance of simulated models will closely resemble the variance of previously observed climate events. When using the multiplicative delta method, the variance of a generated sequence of monthly climate events differs from the variance of observed climate events by a factor of $\left(\frac{1}{\delta_m^l}\right)^2$. Of course, in the case of the additive delta method, the variances are exactly the same.

However, as is apparent from its simplicity, this model lacks flexibility and thus has some predictable limitations. Firstly, the model suffers from sensitivity to the chosen time horizon $\mathcal{P}_{yr}$ [12]. Additionally, while the full recovery of climate variability is important in downscaling models, the delta method's inability to drastically deviate from previously observed data means that all future downscaled simulations will follow the same climate patterns as before. This follows from the variance argument described above. This stable assumption in variance is restrictive and unrealistic especially with the recent volatility induced by climate change. Furthermore, the delta method does not make use of the spatial correlation between observations. All regional climate fields are downscaled independently of the regions surrounding them.

Finally, since GCM models produce coarse output over a large spatial grid, the simulated climate could straddle several heterogeneous climate zones. The delta method does not account for the fact that any coarse, globally simulated climate event could correspond to two high resolution regions with drastically different climate patterns.

By only using the mean of the simulated variables, the delta method fails

to make use of the full probability distributions of both the GCM output variables and the observed variables.

3.2.2 Bias Correction

While the delta method can be seen as downscaling by increasing the similarity between the change in means of the observed data and GCM simulated data, bias correcting methods increase the similarities between the two distributions by making use of the assumption of stationary differences between low resolution and high resolution climate variables [38].

As shown in Figure 6, a bias correction model assumes a constant bias through time between the low resolution climate variables and the high resolution climate variables. Using distributional transformations, the bias in the low resolution random variables is estimated and corrected. This correction is done with the use of quantile mapping (QM) to eliminate the bias between future simulations of high resolution climate variable X and low resolution estimates $\hat{Y}$.

Definition 2 (Quantile Mapping)**.** Given two variables $Z_S \sim F_{Z_S}$ and $Z_O \sim F_{Z_O}$ where Z_O and Z_S represent observations and simulations of Z respectively, the quantile mapping from Z_S and Z_O is defined as

$$Z_O = F_O^{-1}(F_S(Z_S))$$

where F_O and F_S are the cumulative distribution functions (CDF) of the corresponding variables.

Quantile mapping, as used for bias correction and BCSD, is a composition of the probability integral transform of the distribution of low resolution data

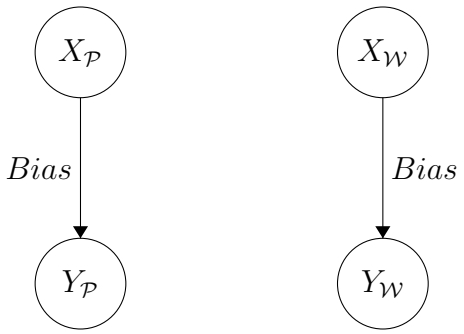

Figure 6: Diagram of Bias Correction downscaling

with the quantile transformation of the high resolution data used to correct the low resolution climate data.

Definition 3 (Bias Correction)**.** Let $\{Y_{v,t}(s_0) \sim F_Y : t \in \mathcal{P}\}$ and let $\{X_{v,t}(s_0) \sim F_X : t \in \mathcal{P}\}$. Methods that use bias correction downscale through the quantile mapping:

$$\{\hat{Y}_{v,t}(s_0) = F_Y^{-1}(F_X(X_{v,t}(s_0))) : t \in \mathcal{W}\}$$

Since the cumulative distribution functions are monotonically increasing, this mapping makes the variables $\hat{Y}$ and X comonotonic at a given time and location i.e. creates strong positive association between the variables while shifting the CDF of X to more closely resemble that of the high resolution data Y. Several approaches for the estimation of the quantile mapping function are discussed next.

Example 2 (Distributional QM)**.** With distributional quantile mapping, the transformation function is estimated using known distributions. This method is mostly used to downscale precipitation where the chosen distribution is a

mixture of the Binomial and Gamma distributions or a simple Gamma distribution. If used, the Binomial distribution identifies the probability of precipitation and the Gamma distribution identifies the intensity of precipitation [14]. Other frequently used distributions include a mixture of Bernoulli and Log-Normal as well as a Bernoulli and Exponential [21]. It should be noted that extreme value distributions would allow for better modeling of the tail behaviour of the low resolution and high resolution data rather than Gamma, Exponential, or Log-Normal which are light tailed [13].

Let us assume an Exponential distribution for past realisation of low resolution and high resolution data at location s_0:

$$\{X_{v,t}(s_0) : t \in \mathcal{W}\} \sim \mathcal{E}(\lambda_{0,1})$$
$$\{Y_{v,t}(s_0) : t \in \mathcal{W}\} \sim \mathcal{E}(\lambda_{0,2})$$

Using estimates for $\lambda_{0,1}$ and $\lambda_{0,2}$ generated through maximum likelihood estimation, the QM can be constructed. The learned quantile mapping between the GCM observed precipitation yields the following transformation:

$$\left\{\hat{Y}_{v,t}(s_l) = \frac{\hat{\lambda}_{l,1}}{\hat{\lambda}_{l,2}} X_{v,t}(s_l) : t \in \mathcal{W}, s_l \in \mathcal{S}\right\}$$

Depending on the goal of the downscaling model, more complex distributions can be used. For example, [30] use the Generalized Pareto distribution to create a cumulative distribution mapping that can downscale low resolution extremes of precipitation to high resolution extremes. With this choice of transformation, the generated high resolution samples of precipitation are not accurate simulations of day-to-day precipitation but are accurate simulations of the tail of high resolution precipitation. $\bigtriangledown$

Example 3 (Parametric QM)**.** Another approach to QM is to assume a parametric relationship between the two variables of interest. In essence, the function $F_Y^{-1}(F_X(\cdot))$ can be estimated directly rather than though the sequential use of cumulative distribution functions.

Some commonly used parametric functions [14] for dowscaling a location s_0 are listed below:

$$\begin{aligned}
\hat{Y}_{v,t}(s_0) &= a_0 X_{v,t}(s_0) \\
\hat{Y}_{v,t}(s_0) &= a_0 X_{v,t}(s_0) + b_0 \\
\hat{Y}_{v,t}(s_0) &= a_0 X_{v,t}(s_0)^{c_0} \\
\hat{Y}_{v,t}(s_0) &= (a_0 X_{v,t}(s_0) + b_0)\left(1 - \exp\left\{\frac{X_{v,t}(s_0)}{\gamma_0}\right\}\right)
\end{aligned}$$

The parameters a_0, b_0, c_0, γ_0 for each location can recovered through least squares estimation using the past sequence $t \in \mathcal{P}$. Let $\widehat{F_{Y_0}^{-1}(F_{X_0}}(\cdot))$ be the learned mapping at location s_0. Future high resolution realizations of data are then generated as

$$\{\hat{Y}_{i,t}(s_l) = \widehat{F_{Y_l}^{-1}(F_{X_l}}(X_{i,t}(s_l)) : t \in \mathcal{W}, s_l \in \mathcal{S}\}$$

$\triangledown$

Example 4 (Empirical QM)**.** Lastly, the functions F_Y, F_X can be estimated individually at each location using the method of empirical cumulative distribution function. This simple non-parametric approach to quantile mapping makes fewer assumptions about the distribution of the observed climate variable.

We can recover the empirical cumulative distribution functions at location s_0:

$$\hat{F}_{X_0}(\tilde{x}) = \frac{\sum_{t \in \mathcal{P}} \mathbb{1}_{X_{i,t}(s_0) < \tilde{x}}}{n}$$
$$\hat{F}_{Y_0}(\tilde{y}) = \frac{\sum_{t \in \mathcal{P}} \mathbb{1}_{Y_{i,t}(s_0) < \tilde{y}}}{n}$$

A spline interpolator is used to smooth the empirical cumulative distribution functions. Finally, quantiles of the low resolution data are mapped to corresponding quantiles of high resolution data.

$$\{\hat{Y}_{i,t}(s_l) = \hat{F}_Y^{-1}(\hat{F}_X(X_{i,t}(s_l))) : t \in \mathcal{W}, s_l \in \mathcal{S}\}$$

One downside of this approach is that large simulations of data are usually matched to the 99^{th} quantile of observed data. This restricts the range of the data that can be simulated with this model. $\bigtriangledown$

Definition 4 (Bias Correction Spatial Dissagregation)**.** As outlined in Figure 7 and introduced in [69], bias correction spatial dissagregation (BCSD) first aggregates the high resolution data to be on the same scale as the low resolution data, estimates and eliminates the bias in the simulated data, then spatially disaggregates the unbiased low resolution simulation data into estimates of high resolution data.

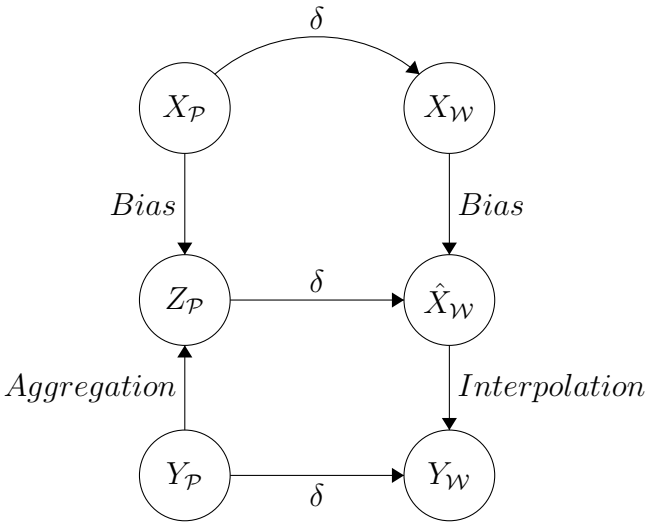

Figure 7: Diagram of BCSD downscaling.

First, neighboring spatial locations of the high resolution data are averaged and used to create a new climate variable Z such that $N_Z = N_X$ and $M_Z = M_X$.

$$\{\text{MeanNeighbors}(Y_{v,t}(s_l)) = Z_{v,t}(s_l) \sim F_Z : t \in \mathcal{P}, s_l \in \mathcal{S}\}$$

This creates a low resolution climate variable from the high resolution climate variable by averaging the over adjacent locations. The bias correction required is then approximated by using the model

$$Z_{v,t}(s_l) = F_Z^{-1}(F_X(X_{v,t}(s_l))) \qquad t \in \mathcal{P}, s_l \in \mathcal{S}.$$

The function composition $F_Z^{-1}(F_X(\cdot))$ is estimated with any of the previously covered methods. The learned quantile mapping is used to eliminate the bias at the coarse level of future simulated data.

$$\{\hat{X}_{v,t}(s_l) = \widehat{F_Z^{-1}(F_X(}X_{v,t}(s_l)) : t \in \mathcal{W}, s_l \in \mathcal{S}\}$$

Finally, the future simulated data is interpolated to have the same spatial variance as the past observed data. This spatial interpolation can be done using the delta method described in the previous section but instead of estimating δ between the original coarse variables $X_{\mathcal{P}}$ and $X_{\mathcal{W}}$, the δ parameter is estimated between the aggregated high resolution variable $Z_{\mathcal{P}}$ and the bias corrected coarse simulations $X_{\mathcal{W}}$.

Bias correction allows for matching moments between future and past high resolution climate data. However, BCSD relies on the assumption that there exist a direct quantile to quantile mapping between global simulations of climate and local observations of climate in addition to the assumption that changes in low resolution data are reflected in high resolution data. Neither of these assumptions hold over long periods of time [59]. The model also does not readily make use of any complex temporal or spatial relationships within the observed and simulated samples and does not have the flexibility to generate future simulations of data that are significantly different from previously observed data.

3.2.3 Multiple Linear Regression

In simple linear regression (SLR), we assume the following relationship between the coarse predictors X and the local predictand Y:

$$Y = \epsilon + \beta X \qquad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

As it is defined above, this model is not considerably different than the parametric quantile mapping function. The appeal of linear regression comes

from the extension into multiple linear regression model (MLR). Let

$$\{v_1, \ldots, v_c\} = \nu \subseteq \mathcal{V}$$

be a set climate variables. Given low resolution climate variables

$$\{X_{v,t}(s_l) : v \in \nu, s_l \in \mathcal{S}, t \in \mathcal{P} \cup \mathcal{W}\}$$

and target high resolution measurement of a climate variable $v_1 \in \mathcal{V}$

$$\{Y_{v_1,t}(s_l) : t \in \mathcal{P}, s_l \in \mathcal{S}\}$$

we can use MLR models of the form

$$Y_{v_1,t}(s_l) = \epsilon + \beta_0 + \sum_{v \in \nu} \beta_v X_{v,t}(s_l) \qquad s_l \in \mathcal{S}, t \in \mathcal{P}, \epsilon \sim \mathcal{N}(0, \sigma^2)$$

In order to estimate β. Due to the assumption of normal errors, most methods make use of some transformation of the target variable $Y_{v_t,t}$ in order to ensure pseudo-normality in the data. Such methods include mean-variance normalization [66] and Anbscomebe residuals [29].

From there, it is simple to generate downscaled realizations of climate variable $Y_{v_1,t}$ at location s_l for any $t \in \mathcal{W}$ as

$$\{\hat{Y}_{v_t,t}(s_l) = \hat{\beta}_0 + \sum_{v \in \nu} \hat{\beta}_v X_{v,t}(s_l) : s_l \in \mathcal{S}, t \in \mathcal{W}\}$$

As is standard, the parameters β_v can be estimated through likelihood maximization or gradient based optimization. In MLR, there is still a linearity assumption on the relationship between GCM simulation data and observed data. Unlike previously covered models, MLR can leverage several different climate variables. This structure allows the model to learn the linear relationship between multiple coarse climate variables and local climate variables.

Example 5 (MLR)**.** The natural correlation between temperature, surface pressure and precipitation could be used to build the multiple linear regression model below:

$$Y_{precip} = \epsilon + \beta_0 + \beta_{precip}X_{precip} + \beta_{temp}X_{temp} + \beta_{press}X_{press} \qquad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

$\bigtriangledown$

In practice, many MLR downscaling models adequately target the mean of the local climate variable but are unable to capture its variance [46]. This means in particular that the MLR models, as described here, might not be appropriate for the downscaling of extreme climate events. There have been some improvements in the downscaling of extremes with methods that enforce similarities between the variance of observed data and the variance of the MLR model. This is done by adding the constraint

$$(\beta X)^T \beta X = Y^T Y$$

to the model parameter optimization [7]. While this implies that enforcing similarities between the variance of the downscaled variable and the local variable leads to be better coverage of extremes, the MLR model still assumes a Gaussian distribution for the local variable Y.

Example 6 (SDSM)**.** One of most often used tools for downscaling precipitation is the Statistical DownScaling Model (SDSM) [66] which is a hybrid model created to make use of MLR and stochastic weather generation [67]. This duality arises from the use of global scale climate patterns to generate precipitation events.

Consider a single location s_0 and climate variable subset ν as before. The SDSM process is as follows: first, all input predictors are normalized to have mean 0 and standard deviation 1.

$$u_{v,t}(s_0) = \frac{X_{v,t}(s_0) - \hat{\mu}_{X_{v,t}(s_0)}}{\hat{\sigma}_{X_{v,t}(s_0)}} \qquad v \in \nu, t \in \mathcal{P} \cup \mathcal{W}$$

Here $\hat{\sigma}_{X_{v,t}(s_0)}, \hat{\mu}_{X_{v,t}(s_0)}$ are the sample mean and sample standard deviation respectively. This is motivated in the model's white paper as a method that has been empirically shown to increase model performance [67].

Following this normalization, an autoregressive regression model for the conditional occurrence probability of daily precipitation ω_t is developed from the coarse climate variables and the past value of ω.

$$\omega_t(s_0) = \alpha_0 + \alpha_1 \omega_{t-1}(s_0) + \sum_{v \in \nu} \alpha_v u_{v,t}(s_0) \qquad t \in \mathcal{P}$$

After the estimation of parameters through least squares optimization, a sequence of standard uniform random numbers $\{r_t(s_0) \sim \mathcal{U}(0,1) : t \in \mathcal{W}\}$ is generated and used to simulate precipitation events. This allows for a probabilistic model for the observance of precipitation which adds texture to the variability of the model and results in a stochastic weather generator. At any $t \in \mathcal{W}$, future precipitation events are generated if $\hat{\omega}_t(s_0) \leq r_t(s_0)$ where

$$\{\hat{\omega}_t(s_0) = \hat{\alpha}_0 + \hat{\alpha}_1 \omega_{t-1} + \sum_{v \in \nu} \hat{\alpha_v} u_{v,t}(s_0) : t \in \mathcal{W}\},$$

As explained below, the MLR model is used for past precipitation amounts, viz.

$$Y_{precip,t}(s_0) = \exp\left\{\beta_0 + \sum_{v \in \nu} \beta_v u_{v,t}(s_0) + \xi\right\}, \qquad t \in \mathcal{P},$$

where ξ is some model error with no distributional assumptions and the exponential function is used to ensure non-zero amounts of precipitation. Finally, for $t \in \mathcal{W}$, downscaled simulations of high resolution precipitation are given by

$$\hat{Y}_{precip,t}(s_0) = \begin{cases} \exp\left\{\hat{\beta}_0 + \sum_{v\in\nu} \hat{\beta}_v u_{v,t}(s_0)\right\} & \text{if } \hat{\omega}_t(s_0) \leq r_t(s_0) \\ 0 & o.w. \end{cases}$$

This process is repeated for different locations $s_l \in \mathcal{S}$. $\bigtriangledown$

Example 7 (Multisite Downscaling)**.** As introduced in [29], MLR can also be used to downscale high resolution climate data at multiple locations simultaneously. For climate variable i measured over space $\mathcal{S}$, we define the matrix of past high resolution data as

$$\mathbf{Y}_{i,\mathcal{P}} = \begin{bmatrix} \mathbf{Y}_{i,\varrho_1} \\ \vdots \\ \vdots \\ \mathbf{Y}_{i,\varrho_{|\mathcal{P}|}} \end{bmatrix} = \begin{bmatrix} Y_{i,\varrho_1}(s_1) & Y_{i,\varrho_1}(s_2) & \dots & Y_{i,\varrho_1}(s_L) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \dots & \ddots & \vdots \\ Y_{i,\varrho_{|\mathcal{P}|}}(s_1) & \dots & \dots & Y_{i,\varrho_{|\mathcal{P}|}}(s_L) \end{bmatrix}_{|\mathcal{P}|\times L}$$

We define the design matrix of past low resolution data as:

$$\mathbf{X}_{\mathcal{P}} = \begin{bmatrix} \mathbf{X}_{*,\varrho_1} \\ \vdots \\ \vdots \\ \mathbf{X}_{*,\varrho_{|\mathcal{P}|}} \end{bmatrix} = \begin{bmatrix} < \mathbf{X}_{v_1,\varrho_1}, \ldots, \mathbf{X}_{v_c,\varrho_1} > \\ \vdots \\ \vdots \\ < \mathbf{X}_{v_1,\varrho_{|\mathcal{P}|}}, \ldots, \mathbf{X}_{v_c,\varrho_{|\mathcal{P}|}} > \end{bmatrix}_{|\mathcal{P}| \times (L*c)}$$

$$= \begin{bmatrix} X_{v_1,1}(s_1) & \ldots & X_{v_1,1}(s_L) & \ldots & X_{v_c,1}(s_1) & \ldots & X_{v_c,1}(s_L) \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ X_{v_1,|\mathcal{P}|}(s_1) & \ldots & X_{v_1,|\mathcal{P}|}(s_L) & \ldots & X_{v_c,|\mathcal{P}|}(s_1) & \ldots & X_{v_c,|\mathcal{P}|}(s_L) \end{bmatrix}$$

Because the dimension of $\mathbf{X}_{\mathcal{P}}$ can lead to issues with multi-colinearity, we first process this design matrix. This can be done through any dimensionality reduction technique such as principal component analysis. Let $\tilde{\mathbf{X}}_{\mathcal{P}}$ be the resulting processed matrix of dimension $|\mathcal{P}| \times k$. The multivariate MLR model uses the model

$$\mathbf{Y}_{i,\mathcal{P}} = \tilde{\mathbf{X}}_{\mathcal{P}}\boldsymbol{\beta} + \boldsymbol{\mathcal{E}},$$

where $\boldsymbol{\beta}$ is a matrix of $[k \times L]$ coefficients to be estimated and $\boldsymbol{\mathcal{E}}$ is a $[|\mathcal{P}| \times L]$ matrix of model errors. Following standard regression practice, the coefficient matrix can be estimated as

$$\hat{\boldsymbol{\beta}} = (\tilde{\mathbf{X}}_{\mathcal{P}}^T \tilde{\mathbf{X}}_{\mathcal{P}})^{-1} \tilde{\mathbf{X}}_{\mathcal{P}}^T \mathbf{Y}.$$

Similarly to the matrix used for calibration, the design matrix $\mathbf{X}_{\mathcal{W}}$ must first undergo dimensionality reduction. High resolution downscaled realizations of climate variable i with temporal and spatial correlation can then be generated through

$$\hat{\mathbf{Y}}_{i,\mathcal{W}} = \tilde{\mathbf{X}}_{\mathcal{W}}\hat{\boldsymbol{\beta}}.$$

The authors of [29] remark that the variance of high resolution samples generated with this method is smaller than the true variance of the high resolution samples which is expected as the MLR model targets the mean of the high resolution and will have difficulties recovering extremes. $\bigtriangledown$

3.2.4 Generalized Linear Regression

As previously mentioned, some distributions, specifically the Gamma distribution, are better suited for modeling precipitation than the normal distribution. Generalized linear models (GLM) allow for the use of a linear estimator to target the mean of a random variable following a non-normal distribution. Given data from the previous example, the corresponding GLM would be

$$g(\mu_{precip,t}(s_l)) = \beta_0 + \sum_{v \in \nu} \beta_v X_{v,t}(s_l) \qquad Y_{precip,t}(s_l) \sim \mathcal{G}(\mu_{precip,t}, \theta), s_l \in \mathcal{S}, t \in \mathcal{P}$$

where g is the link function such as the identity or log link. After parameter estimation, a future sequence of high resolution precipitation downscaled from low resolution simulation of precipitation would be estimated as

$$\left\{ \hat{\mu}_{precip,t}(s_l) = g^{-1}\left(\hat{\beta}_0 + \sum_{v \in \nu} \hat{\beta}_v X_{v,t}(s_l) \right) : t \in \mathcal{W}, s_l \in \mathcal{S} \right\}$$

The Gamma GLM with log link function g has been shown to outperform both MLR and SDSM on select tasks [45]. However, since the GLM targets the mean and the Gamma distribution is light tailed, this precipitation downscaling model is unable to recover the extremes of climate events.

Even with their increased flexibility and the ability to include multiple climate features, the Gamma GLM makes assumptions on the variance, thereby restricting the data that can be modeled accurately.

3.2.5 Copula Dependence Modeling

Recently, copula modeling of climate variables has been used to generate spatio-temporally correlated downscaled realizations of high resolution data. This is because copulas can be used to increase the flexibility of the dependence structure of generated data.

Definition 5 (Copula)**.** A Copula is a cumulative distribution function whose univariate margins are standard uniform.

Copulas are used to create a link between a joint distribution function and its univariate margins [18] by application of Sklar's theorem [42].

Theorem 1 (Sklar's Theorem)**.** Given random variables $Z_1, \ldots, Z_d$ with corresponding marginal cumulative distribution functions $F_1, \ldots, F_d$, there always exists a copula C such that:

$$P(Z_1 \leq z_1, \ldots, Z_d \leq z_d) = C(F_1(z_1), \ldots, F_d(z_d)) \qquad \forall z_1 \ldots, z_d \in \mathbb{R}$$

This means that the joint cumulative distribution of random variables $\{Z_i\}_{i=1}^d$ can be represented by a function of the marginal probability distributions.

Example 8 (Bivariate Copula)**.** This example illustrates the application of bivariate copulas for downscaling as used in [72, 58]. Let $C_\theta(u, h)$ be a bivariate copula parameterized by θ. Consider climate variables at one location s_0

$$\{X_{v,t}(s_0) \sim F_{X_v,0} : t \in \mathcal{P} \cup \mathcal{W}\} \qquad \{Y_{v,t}(s_0) \sim F_{Y_v,0} : t \in \mathcal{P}\}$$

As seen in QM, the cumulative distribution functions can be estimated and used as probability integral transforms. This yields the following sequences:

$$\{u_{v,t}(s_0) = \hat{F}_{X_v,0}(X_{v,t}(s_0)) : t \in \mathcal{P} \cup \mathcal{W}\} \quad , \quad \{h_{v,t}(s_0) = \hat{F}_{Y_v,0}(Y_{v,t}(s_0)) : t \in \mathcal{P}\}$$

The copula C_θ is then fitted to these approximately uniform marginal variables to estimate θ. For copulas belonging the Archimedian family of copulas, this can be done through the use of Kendall's τ correlation coefficient [41], for example, or more generally, using pseudo maximum likelihood estimation [17].

Furthermore, as explained in [51], the conditional CDF of a bivariate distribution can be recovered from a copula as:

$$F_{H|U}(h|u) = P(H \leq h|U = u) = \frac{\partial C_\theta(u, h)}{\partial u}.$$

This gives a formulation for the conditional CDF of the high resolution climate variable given the low resolution climate variable after both have undergone the probability integral transform. With an estimate $C_{\hat{\theta}}$, low resolution samples generated at $t \in \mathcal{W}$ can be downscaled into high resolution samples. First, uniform transformations of the future low resolution climate variable are used along with the conditional CDF derived from the copula to simulate uniform samples of the high resolution climate variable.

$$\{\hat{h}_{v,t}(s_0) \sim \hat{F}_{H|U=u_{v,t}(s_0)} : t \in \mathcal{W}\} \qquad \hat{F}_{H|U} = \frac{\partial C_{\hat{\theta}}(u, h)}{\partial u}$$

Then, the inverse of the approximated CDF of the high resolution climate variable is used to generate scaled future representations.

$$\{\hat{Y}_{i,t}(s_0) = \hat{F}^{-1}_{Y_i,0}(\hat{h}_{v,t}(s_0)) : t \in \mathcal{W}\}$$

This copula model is an improvement over QM methods bias correction method described in Section 3.2.2. This is because the bivariate copula downscaling method allows for the conservation of dependence between high

resolution and low resolution data, but no longer assumes comonoticity between these variables. ▽

Example 9 (Multivariate Copula)**.** Higher dimensional copulas can be used in tandem with CDF estimation methods to generate sequences that preserve more of the spatial and temporal variability of high resolution climate variables. As proposed in [6], this method of downscaling has shown its ability to produce more realistic downscaled high resolution climate data than Multisite MLR based downscaling.

Consider the climate variable matrices defined in Example 7. As before, matrices $\mathbf{X}_{\mathcal{P}}$ and $\mathbf{X}_{\mathcal{W}}$ must be processed with dimensionality reduction into matrices $\tilde{\mathbf{X}}_{\mathcal{P}}$ and $\tilde{\mathbf{X}}_{\mathcal{W}}$ respectively.

First, the conditional cumulative distribution function of the target variable at each location is estimated. This is done through conditional quantile regression. As introduced in [32], this model is an extension to median regression. Let $\{Z_{y,i}\}_{i=1}^{n}, \{Z_{x,i}\}_{i=1}^{n}$ be n observations of two variables. The p^{th} quantile of Z_y conditional on Z_x is modeled as

$$Q_p(Z_y|Z_x) = Z_x\beta.$$

In [32] it is shown that the beta parameter can be estimated by:

$$\hat{\beta} = \min_{\beta\in\mathbb{R}} \left[\sum_{i=1}^{n} p|Z_{y,i} - Z_{x,i}\beta| \cdot \mathbb{1}_{x_i\beta\leq y_i} + \sum_{i=1}^{n}(1-p)|Z_{y,i} - Z_{x,i}\beta| \cdot \mathbb{1}_{x_i\beta>y_i} \right].$$

Consider location $s_0 \in \mathcal{S}$ and probability p on a gridded partition of the unit interval. Quantile regression is used to model the p^{th} quantile of the high resolution climate at a specific location conditional on the reduced design matrix.

$$Q_p(\mathbf{Y}_{i,\mathcal{P}}(s_0)) = \tilde{\mathbf{X}}_{\mathcal{P}}\beta_{p,s_0}$$

The parameter β_{p,s_0} is estimated as above. This process is repeated for equally spaced values of p on the unit interval in order to generate multiple quantile coefficients. After parameter estimation, the conditional quantile of $Y_{i,t}$ is estimated as

$$\hat{Q}_{Y_{i,t}(s_0)|\mathbf{X}_{*,t}}(p) = \tilde{\mathbf{X}}_{*,t}\hat{\beta}_{p,s_0}$$

These quantile estimates are then smoothed and interpolated to build the conditional cumulative distribution function $\hat{F}_{\mathbf{Y}_{i,t}(s_0)|\tilde{\mathbf{X}}_{*,t}}$. Finally, this procedure is repeated for all locations to produce location and time specific CDF estimates

$$\{\hat{F}_{\mathbf{Y}_{i,t}(s_l)|\tilde{\mathbf{X}}_{*,t}} : s_l \in \mathcal{S}, t \in \mathcal{P} \cup \mathcal{W}\}$$

Using these CDFs, the high resolution variable is converted onto the unit interval

$$\{u_{i,t}(s_l) = \hat{F}_{\mathbf{Y}_{i,t}(s_l)|\tilde{\mathbf{X}}_{*,t}}(Y_{i,t}) : s_l \in \mathcal{S}, t \in \mathcal{P}\}$$

Next, [6] apply the Gaussian copula to estimate the spatial dependence of the transformed high resolution climate variables. The Gaussian copula takes the form given, for all $h_1, \ldots, h_d \in (0, 1)$ by

$$C_\Sigma(h_1, \ldots, h_d) = \Phi_d(\Phi^{-1}(h_1), \ldots, \Phi^{-1}(h_d)|\Sigma),$$

where Φ is the standard normal CDF and Φ_d is the d-dimensional multivariate normal CDF with $d \times d$ correlation matrix parameter Σ that controls the dependence structure of the variables. Assuming that

$$\left\{\left(\Phi^{-1}(u_{i,t}(s_1)), \ldots, \Phi^{-1}(u_{i,t}(s_L))\right) \sim \mathcal{N}_L(0, \Sigma) : t \in \mathcal{T}\right\}$$

The matrix parameter Σ of the Gaussian copula is estimated by the sample correlation of the rows of matrix $\mathbf{H}$.

$$\mathbf{H} = \begin{bmatrix} \Phi^{-1}(u_{i,\varrho_1}(s_1)) & \dots & \Phi^{-1}(u_{i,\varrho_1}(s_L)) \\ \vdots & \ddots & \vdots \\ \Phi^{-1}(u_{i,\varrho_{|\mathcal{P}|}}(s_1)) & \dots & \Phi^{-1}(u_{i,\varrho_{|\mathcal{P}|}}(s_L)) \end{bmatrix}_{|\mathcal{P}| \times L}$$

This completes the calibration step of the model. In order to generate downscaled observations of climate, the model first generates samples from the Gaussian copula:

$$\{(\hat{u}_{i,t}(s_1), \dots, \hat{u}_{i,t}(s_L)) \sim C_{\hat{\Sigma}} : t \in \mathcal{W}\}$$

The copula generates uniform future realizations of high resolution climate variables generated with a similar spatial structure to those used in the calibration step. These are then transformed into the appropriate scale by using the estimated CDFs which model the temporal variability of observations in $\mathcal{P}$:

$$\left\{ \hat{Y}_{i,t} = \hat{F}^{-1}_{Y_{i,t}(s_l)|\tilde{\mathbf{X}}_{*,t}}(\hat{u}_{i,t}(s_l)) : t \in \mathcal{W}, s_l \in \mathcal{S} \right\}$$

This model performs significantly better than multisite MLR models when it comes to the recovery of extreme events [6]. However, the CDF estimation step assumes that the quantile mapping stays stable through time. As previously discussed in Secion 3.2.2, this assumption does not hold for wide temporal windows.

When modeling the temporal distribution, the spatial dependence is ignored. Similarly, when estimating and generating samples from the copula, dependence over time is ignored. This two step approach means that the joint spatio-temporal dependence of high resolution and low resolution climate variables is not modeled exactly. Additionally, the choice of Gaussian

copula is restrictive because of the spacial dependence it can model. Newer high dimension extreme value copulas may be more useful though harder to estimate. For example, [36] use empirical copulas to estimate the dependence structure between high resolution and low resolution climate. $\triangledown$

3.3 Machine Learning Models for Downscaling

Many machine learning (ML) techniques have been applied to the task of downscaling. In this section we provide a brief overview of such methods. This overview is done in the context of general machine learning models where Y is a predictand and $\mathbf{X} = (X_1, \ldots, X_M)$ is a set of predictors. The models are calibrated and trained by estimating the relationship f:

$$Y = \{Y_{i,t}(s_l) : s_l \in \mathcal{S}, t \in \mathcal{P}\} = f(\mathbf{X})$$
$$\mathbf{X} = \{X_{j,t}(s_l) : s_l \in \mathcal{S}, t \in \mathcal{P}, j = 1, \ldots, M\}$$

As before, downscaled realizations of future simulations are generated as

$$Y = \{\hat{Y}_{i,t}(s_l) : s_l \in \mathcal{S}, t \in \mathcal{W}\} = \hat{f}(\mathbf{X})$$
$$\mathbf{X} = \{X_{j,t}(s_l) : s_l \in \mathcal{S}, t \in \mathcal{W}, j = 1, \ldots, M\}$$

If the predictand Y represents observations of a climate variable to be downscaled, then ML models are used to approximate statistical downscaling methods. As in the case in this thesis, when Y represents high resolution simulations (such as RCM output), the ML models are used to approximate the RCM and thus can be seen as approximations of dynamic downscaling models.

This thesis focuses on the application of deep learning methods for downscaling that have recently gained popularity. Most statistical methods are built around distributional assumptions that hold on the actual range of climate variables. As discussed in Section 2, the data available is an encoding of these climate variables. Applying statistical downscaling models to the distribution of color scale encoding of the pixels that make up the image of climate events would result in an inadequate comparison of the modeling techniques.

3.3.1 Genetic Programming

Genetic programming (GP), first introduced by [33], can be used as a non-parametric regressor when the nature of the relationship between the predictor and the predictand is not well understood. In contrast to standard statistical modeling and linear regression methods, GP can allow for non-linear relationships between the predictor and the predictand, does not rely on assumptions of variable independence, and does not rely on the standard MLR assumptions [8].

Given a set of functions (+, -) and a set of operations ($\ln, \sin, \ldots$), a GP algorithm randomly generates a generation. A generation is a set of models that uses the input functions and a subset of the coarse predictor variables $\mathbf{X}$ to model the high resolution variable Y. Members of a generation are represented by graphical trees where the nodes constitute of coarse variables or constants, and the edges correspond to a function from the set. A GP algorithm generates populations until a stopping criteria is reached and then selects the best performing member of the final population.

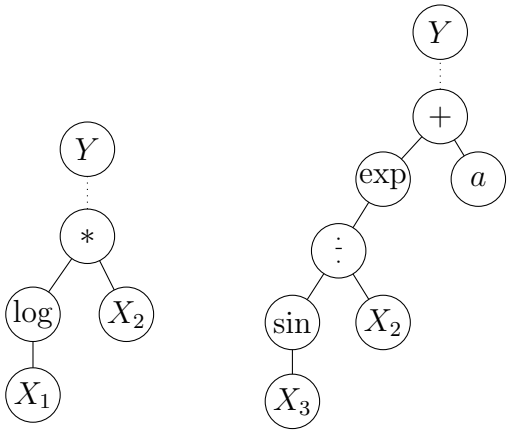

Figure 8: Sample members of a generation from GP algorithm with operation set $\{*, +, \div\}$, function set $\{\log, \exp, \sin\}$, coarse features $\mathbf{X} = (X_1, X_2, X_3)$, and target local feature Y.

Models from a generation are then evaluated by assessing their ability to properly downscale the local variable. The better performing models are then kept in future generations while the worse performing models are purged. At every generation, slight perturbations can happen to the current members. They can be mutated, which changes the value in a node of the graph, or crossover, which swaps any two branches from different trees.

Genetic models have been shown to have better performance than MLR methods for the downscaling of extreme rainfall [23]. When it comes to the actual prediction of the precipitation levels, GP algorithms tend to underestimate downscaled precipitation variables, though their estimate is still more accurate than MLR approaches [26].

3.3.2 Support Vector Machines

With support vector machines (SVM), the target local variable Y is approximated with the use of the model

$$f(\mathbf{X}) = \mathbf{w}^T \Phi(\mathbf{X}) + b + \epsilon$$

where $\mathbf{w}, b$ are parameters to be estimated and Φ is a non-linear function and ϵ is our error tolerance. Usually, SVMs are optimized through the use of Lagrange multipliers as the function we wish to optimize consists of solving a constrained minimization problem.

Depending on downscaling location, SVM and its variants have shown the potential to outperform MLR approaches in terms of accuracy [50]. They have also shown marginal improvement in the downscaling of extreme precipitation when compared to MLR models [10].

3.4 Deep Learning Models for Downscaling

Deep learning (DL) models can be used for downscaling in several ways. Like statistical downscaling approaches, DL can be used to map the output of a GCM to the ground truth, i.e. a function that estimates the relationship between the coarse global climate variables and the fine local observations. Alternatively, DL can also be used to approximate the RCM output. In this case, the function that is approximated is the regional forcing induced by the RCM to coerce GCM output into high resolution local variables. If this mapping is effective, DL approaches can benefit from the accuracy of RCMs while maintaining the benefit of computational efficiency from the statistical approaches to downscaling.

Furthermore, in the field of deep learning, there have been many models proposed for image super-resolution. This tasks involves learning a mapping from a low quality blurred image to its high resolution counterpart. It is natural to apply deep learning models to the task of downscaling as downscaling can be thought of as a form of image super-resolution where the coarse GCM simulations represent a blurred view of a climate variable and the climate variable at various local regions corresponds to the fine-grain high resolution view. Approaching downscaling as a super resolution problem allows for the use of new tools in this task.

This section outlines some deep learning modules and architectures used in the model proposed in this thesis.

3.4.1 Feed Forward Neural Networks

Feed forward neural networks are deep learning models that, according to the universal approximation theorem [5], can represent a wide array of functions. This means that if we assume a functional relationship between coarse global and granular local climate variables, such networks are well suited for estimation of this function.

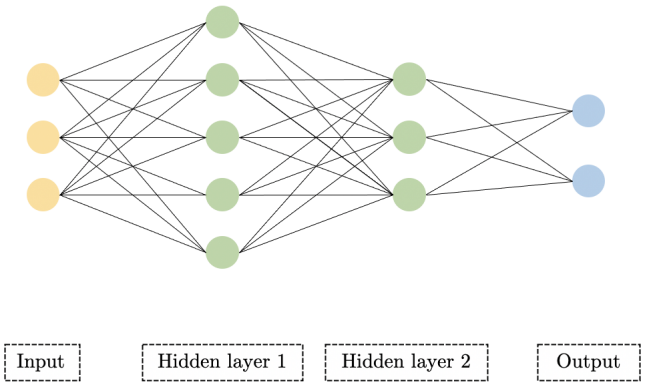

Figure 9: Example of a feed forward neural network with two hidden layers

Multilayer perceptrons (MLP), are a class of feed forward neural networks that consist of linear layers composed with some form of activation function. To define an MLP, let $x \in \mathbb{R}^n$ be a vector of inputs, $y \in \mathbb{R}^m$ be a vector of outputs, and $l \in \mathbb{R}$ be the number of layers in the model. Each layer can be represented as a vector function

$$L_i(z) = F_i(\mathbf{W}_i z + b_i) \qquad i = 1, \ldots, l$$

Where z is an input vector to layer L_i and F_i is a linear or non-linear function. Each layer has vector b_i and matrix $\mathbf{W}_i$ which are parameters to be learned through estimation procedures. An MLP can then be defined as the model of y through the application of the composition of layers on x.

$$y = L_l(L_{l-1}(\ldots(L_1(x))))$$

MLPs have shown to be superior to MLR models for downscaling temperature and precipitation [43]. Additionally, with non-linear choices for the

activation functions F_i, these models can also produce less biased predictions for output y. In practice, MLPs have shown to be more adequate for downscaling precipitation extremes than models like GPs or SVMs [50].

When working with spatially related data such as images or maps, the analogue to an MLP is a convolutional neural network (CNN). These models leverage the spatial relationship of data through a convolutional layer. Convolutions can be extended to multiple dimension,s though the definition below focuses on the two dimensional case. Let $\mathbf{z}$ be a matrix in $\mathbb{R}^{h\times w}$ and let $\mathbf{W}$ be a matrix in $\mathbb{R}^{A\times B}$ such that $h > A$ and $w > B$. $\mathbf{W}$ is termed the convolutional filter. A convolutional operation in a convolution layer C that takes in $\mathbf{z}$ as input and learns parameter $\mathbf{W}$ can be represented as a transformation $C : \mathbb{R}^{h\times w} \mapsto \mathbb{R}^{h-A+1\times w-B+1}$.

$$C(\mathbf{z}) = \mathbf{z} * \mathbf{W} \tag{$\square$}$$

$$C(\mathbf{z})[i,j] = \sum_{r=1}^{A}\sum_{c=1}^{B} \mathbf{z}[i+r-1, j+c-1]\mathbf{W}[r,c]$$

As depicted in Figure 10, a convolution reduces the dimensions of the input matrix. However, the input can be padded with 0s and artificially augmented to ensure equal dimensions in the input and output.

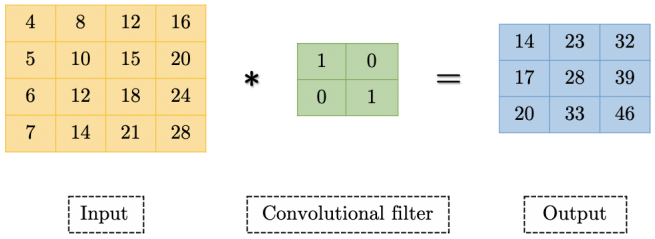

Figure 10: Example of a convolution on a matrix

By leveraging spatial information, CNNs can outperform standard MLPs and have also been shown to perform better than standard regression-based approaches [52]. One downside of these models is that the CNN structure could require high resolution input and output data which could make the model extremely sensitive to the chosen data resolution and greatly increase training time [4].

3.4.2 Recurrent Neural Networks

Recurrent Neural Networks (RNN) are widely used in deep learning to leverage the time dependencies within time series data. These networks use a hidden state h which represents historical information accumulated by the network and generate future state information using current input x. There are many schemes to update the hidden vector. For example, at step t, the network generates the hidden state using the function

$$h_t = R(h_{t-1}, x_t)$$

Different choices for the function R give rise to different types of recurrent neural networks. We focus on the gated recurrent unit (GRU) architecture proposed in [11] and its convolutional variant (ConvGRU) proposed in [3].

A GRU updates the hidden representation of past data using an update gate which signals the extent to which the hidden representation should change:

$$\mathbf{z_t} = \sigma(\mathbf{W_z x_t} + \mathbf{U_z h_{t-1}})$$

With σ as the sigmoid function applied to each element of vector $\mathbf{z}$, viz.

$$\sigma(a) = \frac{1}{1 + e^{-a}} \quad , \quad a \in \mathbb{R}$$

GRUs also make use of a reset gate which signals the extent to which past representations should be forgotten:

$$\mathbf{r_t} = \sigma(\mathbf{W_r x_t} + \mathbf{U_r h_{t-1}})$$

Using the reset gate, a candidate hidden state is created:

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W x_t} + \mathbf{U}(\mathbf{r_t} \odot \mathbf{h_{t-1}}))$$

Here, $\odot$ refers to the Hamard product. The final hidden state is an interpolation (governed by the update gate) between the candidate hidden state and the previous hidden state.

$$\mathbf{h_t} = (\mathbf{1} - \mathbf{z_t}) \odot \mathbf{h_{t-1}} + \mathbf{z_t} \odot \tilde{\mathbf{h}}_\mathbf{t}$$

With $\mathbf{1}$ as a vector of ones. Here, $\mathbf{W}_z, \mathbf{W}_r, \mathbf{W}, \mathbf{U}_z, \mathbf{U}_r$ and $\mathbf{U}$ are all matrix parameters to be learned through an MLP. The inputs $\mathbf{h_{t-1}}$, $\mathbf{x_t}$ and output $\mathbf{h_t}$ are all vectors.

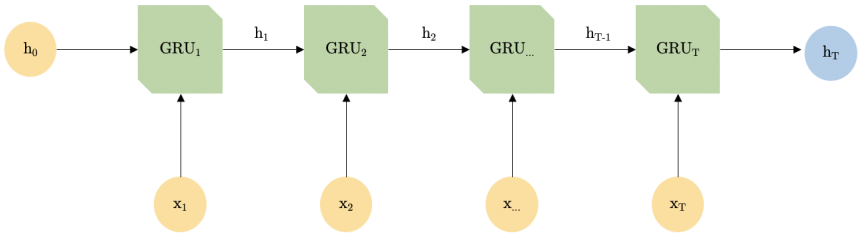

Figure 11: Sequence of GRUs

In order to leverage both spatial and temporal dependence, the authors of [3] propose using convolutions rather than standard matrix multiplication. The algorithm for updating the hidden representation at time t is then changed into

$$\begin{aligned}
\mathbf{z_t} &= \sigma(\mathbf{W}_z * \mathbf{x_t} + \mathbf{U_z} * \mathbf{h_{t-1}}) \\
\mathbf{r_t} &= \sigma(\mathbf{W}_r * \mathbf{x_t} + \mathbf{U_r} * \mathbf{h_{t-1}}) \\
\tilde{\mathbf{h}}_t &= \tanh(\mathbf{W} * \mathbf{x}_t + \mathbf{U} * (\mathbf{r_t} \odot \mathbf{h_{t-1}})) \\
\mathbf{h_t} &= (\mathbf{1} - \mathbf{z_t}) \cdot \mathbf{h_{t-1}} + \mathbf{z_t} \cdot \tilde{\mathbf{h}}_\mathbf{t}
\end{aligned}$$

and $\mathbf{W}_z, \mathbf{W}_r, \mathbf{W}, \mathbf{U}_z, \mathbf{U}_r$ and $\mathbf{U}$ are learned convolutional filters. The inputs $\mathbf{h_{t-1}}$, $\mathbf{x_t}$ and output $\mathbf{h_t}$ are all matrices. In order to preserve spatial dimensions of the data, convolutional GRUs use padded convolutions.

Recurrent modules are flexible in their applications to models. For example, GRUs can be used in a sequential manner as depicted in Figure 11 or, they can also be used in a stacked fashion as seen in Figure 12. The stacked GRUs architecture allows models to increase the complexity and pertinency of the learned hidden representations of data.

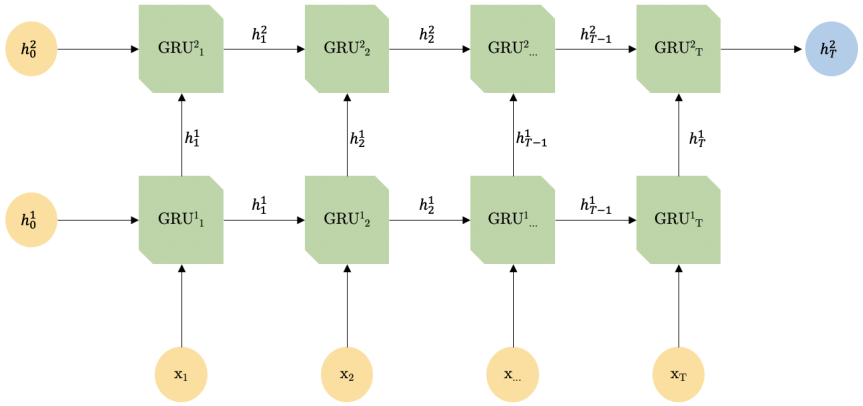

Figure 12: Stacked GRU architecture of depth 2 and length T

Standard GRUs have been shown to perform competitively with other models on the task of downscaling [55] while convolutional GRUs have been shown to more accurately represent the variance of the target variable [35]. These improvements do come with an increase in model complexity as attempting to include spatial dependence in a model magnifies the number of parameters to train and the amount of data necessary.

3.4.3 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a class of models introduced in [20] to improve the efficacy of deep generative models. These models, often used in imaging or video tasks, can be used to generate or evaluate samples from a probability space. The innovative aspect of these networks comes from the fact that they pit two models against each other.

A GAN consists of two models: a generator G and a discriminator D. Let

$\mathbf{Y} \sim p_{data}(\mathbf{y})$ be the target distribution to simulate. The generator is a model that takes in a sample from latent variable Z and creates a mapping from Z to $\mathbf{Y}$ denoted as $G(z)$. The discriminator, defined as $D(\tilde{y})$, is a model that takes in a sample $\tilde{y}$ and distinguishes whether or not $\tilde{y}$ represents a sample from $\mathbf{Y}$. Here, G and D are differentiable functions which can be chosen to be deep neural networks.

With these two adversaries defined, the aim is to train the generator G to "fool" the discriminator D, while training the discriminator D to identify fraudulent samples from $\mathbf{Y}$. This training paradigm enables model G to generate pseudosamples from the distribution of $\mathbf{Y}$ that are indistinguishable by discriminator D from authentic samples. This is done through the optimization of the following loss function:

$$\min_G \max_D V(G, D) = \mathbb{E}_{\mathbf{Y}}[\log D(\mathbf{y})] + \mathbb{E}_Z[\log(1 - D(G(z)))]$$

For the Discriminator, the goal is to maximize this loss function while for the generator, the goal is to minimize it. In game theory, this corresponds to a two player min-max game with value function $V(G, D)$.

As a natural extension to GANs, conditional GANs allow for the inclusion of an external variable x on which to condition the generator and the discriminator. By feeding in additional input x to both the discriminator and the generator, the previous loss function becomes:

$$\min_G \max_D V(G, D) = \mathbb{E}_{\mathbf{Y}}[\log D(\mathbf{y}|x)] + \mathbb{E}_Z[\log(1 - D(G(z|x)))]$$

While GANs can be effective at generating samples, their adversarial architecture leaves them vulnerable to a few different failure modes. These include vanishing gradients, mode collapse, and failure to converge. Many

modifications to the standard GAN have been proposed and drastically improve model training [2, 22].

When applied to downscaling, GANs aim to learn the distribution of a local climate variable $\mathbf{Y}$ conditioned on the coarse global climate variables x. Here $\mathbf{Y}$ corresponds to the distribution of a climate variable over high resolution space and time and x corresponds to coarse simulations of climate events.

Conditional GANs devised for image super resolution have shown promise when applied to downscaling [65, 64]. Newer GAN architectures are also successful when applied to the use of GCM output for the approximation of high resolution RCM output [61] and for direct application to regional downscaling [1, 35, 9]. In this thesis, we focus on the model proposed by [61] and improve on its performance in recovering temporal dependencies within weather patterns.

4 Learned Dynamical Downscaling Summary

High resolution RCM data is computationally expensive to generate. In [61] the authors model high resolution RCM data from low resolution RCM data as this would allow stakeholders to save on resources by producing low resolution RCM data and using it to model high resolution RCM data. This is referred to as as learned dynamical downscaling. The model learns an approximation of a mapping between low resolution RCM simulation and high resolution RCM data.

4.1 Model

The authors of [49] explore a few different models, though this thesis focuses on their best model for downscaling. This model is a conditional GAN that makes use of 12 km resolution topography, simulated 50 km resolution sea level pressure (SLP), integrated water vapour (IWV), temperature (T2) and precipitation as input climate variables and the corresponding 12 km resolution rain simulation as a target. This model is henceforth referred to as FALDD.

4.1.1 Modules

The success of both the generator G and discriminator D of FALDD can be attributed to the modules used to form its architecture. This model makes use of inception modules, introduced in [54], which are widely used in the field of computer vision and have been shown to increase performance on vision tasks while maintaining computational costs low. This is done by the

stacking of multiple convolutional layers of different kernel sizes which allows the model to learn a wide feature maps of the input features. A diagram of this module is available in Figure 13. For this and subsequent diagrams, a legend is provided in Figure 26.

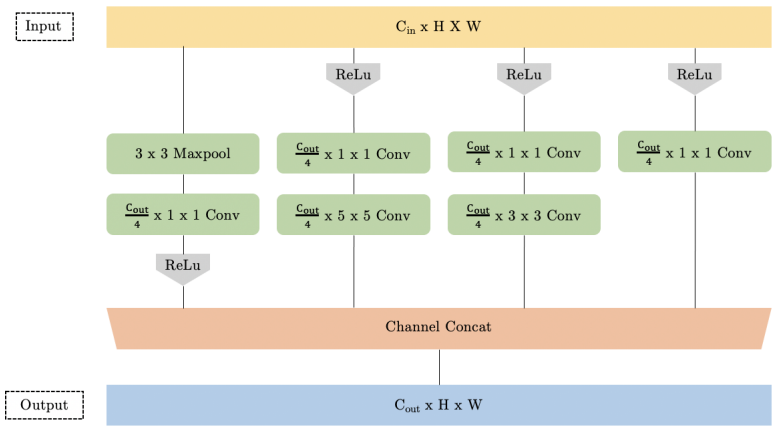

Figure 13: Inception Module Diagram

Information about the extracted feature maps is gathered through the use of convolutional block attention modules (CBAM) introduced in [68]. A CBAM module is an attention module that makes use of both a spatial attention module [57] and a channel attention module [28]. In deep learning, much like in humans, attention is used to increase the model's attentiveness on some areas of the feature space, while decreasing it on others.

The channel attention module is a multilayer perceptron which models the relationship between the maximum and average of each feature map. This relationship is then used to perform a re-weighting of these feature maps.

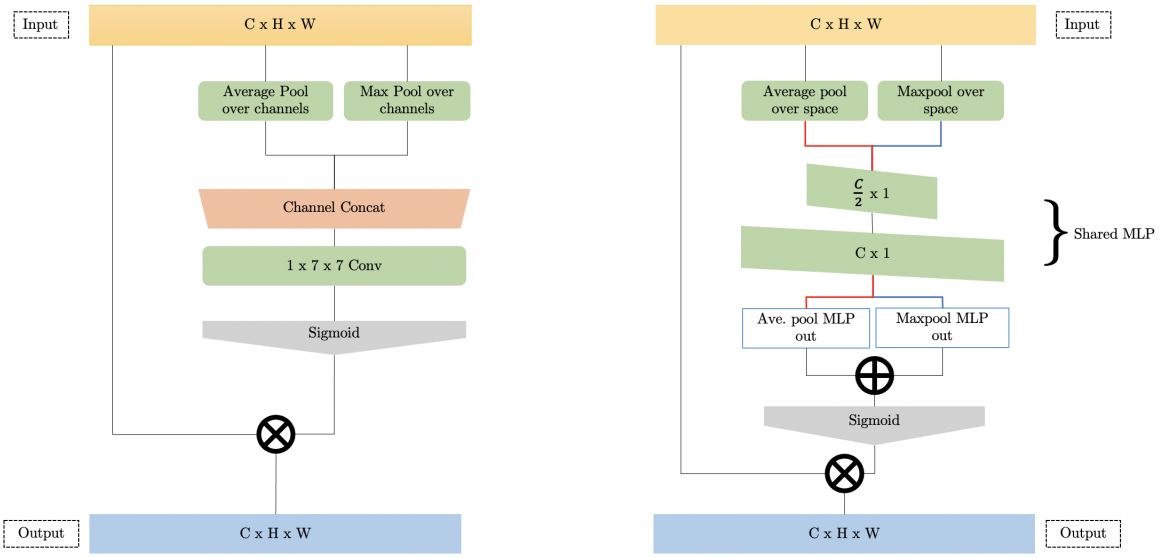

Figure 14: Spatial attention module (left) and channel attention module (right)

In contrast, the spatial attention module, uses convolutional network which learns the relationship between spatial locations within the feature maps. This spatial relationship is then used to re-weight all location features across all maps. Both of the previous modules are used sequentially to form the CBAM attention module.

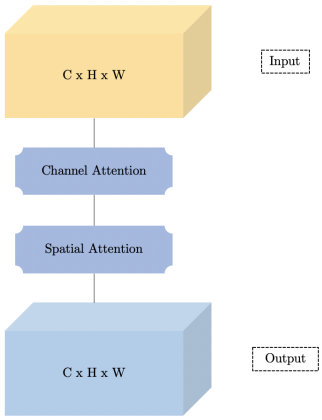

Figure 15: CBAM

4.1.2 Discriminator

Compared to the generator, the discriminator architecture is rather simple. The discriminator is a convolutional neural network which outputs a 4×8 matrix of probabilities.

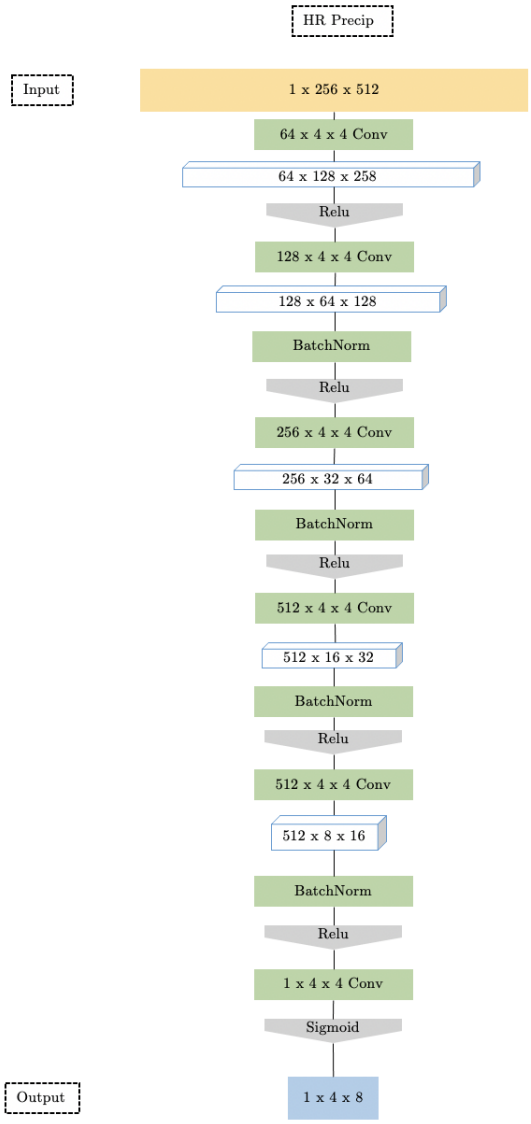

Figure 16: Discriminator Diagram

The authors opted for the use of the standard conditional GAN loss for

the discriminator D with parameters θ_D ,

$$\ell(\theta_D) = \frac{1}{n}\sum_{i=1}^{n} D(G(x_i, v_{slp}, v_{iwv}, v_{t2}, v_{top})) - D(y_i).$$

4.1.3 Generator

In the proposed GAN, all climate variables are processed separately using similar convolutional encoders. These convolutional encoders serve as extractors to ensure that characteristics of the features collected are similar across all the variables. These features are then concatenated into a 3 dimensional tensor.

The generator G uses the previously defined modules to downscale coarse precipitation images into high resolution images.

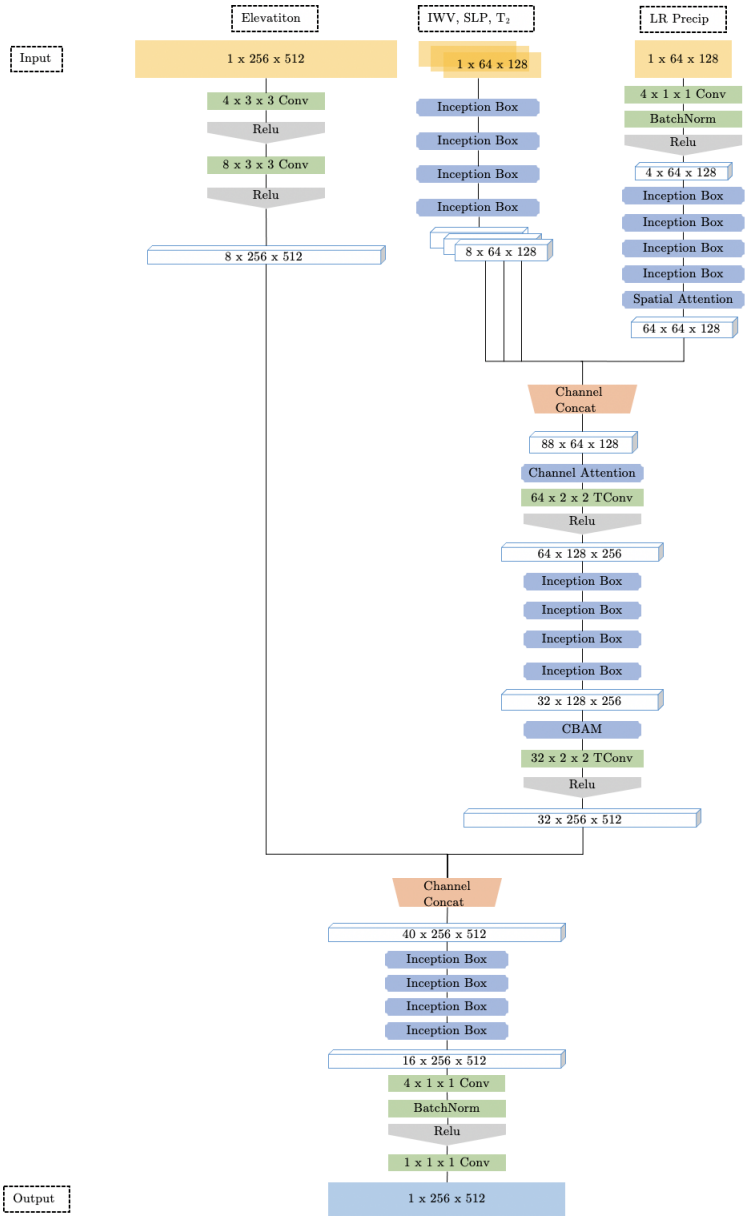

Figure 17: Generator Diagram

The loss function of G is a composition of an adversarial loss and the ℓ_1

loss to optimize its parameters θ_G. The ℓ_1 loss is used to enforce a structural similarity between the generated images and the target images.

$$\begin{aligned}\ell(\theta_G) = &- \frac{\lambda}{n}\sum_{i=1}^{n} D(G(x_{precip,i}, x_{slp,i}, x_{iwv,i}, x_{t2,i}, x_{top,i})) \\ &+ \frac{\alpha}{n}\sum_{i=1}^{n} \ell_1(G(x_{precip,i}, x_{slp,i}, x_{iwv,i}, x_{t2,i}, x_{top,i}) - y_i)\end{aligned}$$

Where λ, α are weighting parameters for the adversarial and content loss respectively. These are set to $\lambda = 1$ and $\alpha = 5$ for the training of the generator.

5 Experiments and Approach

The goal for this thesis is to improve upon the model proposed in [61] through the inclusion of time dependencies. There are few statistical downscaling approaches that take both spatial and temporal dependencies into account. Even the complex copula model described in Example 9 is not able to model joint spatio-temporal dependencies.

Furthermore, none of the methods surveyed modeled spatio-temporal dependence for learned dynamical downscaling. Our contribution to downscaling comes from the use of deep recurrent neural networks and attention modules to model joint spacio-temporal dependence in the context of learned dynamical downscaling.

This is done first by the incorporation of recurrent networks in FALDD to build a model FALDDGRU, and then by the creation of a novel model which uses a collection of tools inspired by research on both climate forecasting and climate downscaling. This model is termed DRDD as it is a deep recurrent dynamical downscaling model. It is available at `https://github.com/jdjame/GRU-DD`.

5.1 Model I: FALDDGRU

The only difference between FALDDGRU and FALDD is the addition of a GRU sequence to recover time dependencies within the data. This is done explicitly to assess the effects of the naive implementation of recurrent neural networks in downscaling models.

Our input is of size $T \times 1 \times 64 \times 128$ where T is the time horizon we

are contemplating. After the low resolution rain, sea level pressure, water vapour, and temperature are preprocessed, their feature maps are passed through to a series of GRUs (identical to the series depicted in Figure 11) to recover a time-dependent representation of each of the variables feature maps.

After this step, the FALDDGRU and FALDD models are identical.

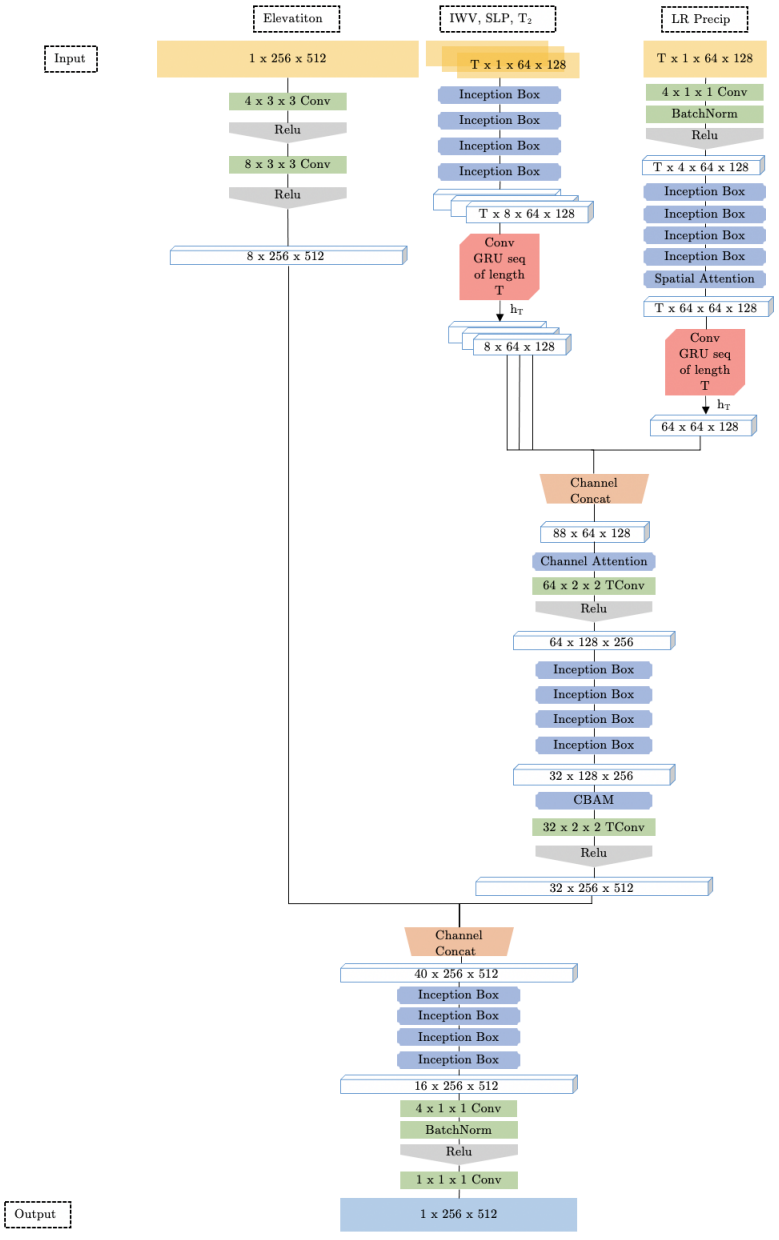

Figure 18: FALDDGRU generator

The same loss function and discriminator architectures are used in the

FALDDGRU model and the original FALDD model.

5.2 Model II: DRDD

There are a few significant architectural differences between the proposed model and the original FALDD model. The three most important ones are: the chosen upsampling scheme, the inclusion of time dependencies, and the use of a WGAN loss rather than a GAN loss.

5.2.1 Modules

Residual networks and connections have experienced success for the task for image super resolution [34, 62] and are thus applied in DRDD. These connections allow for an effectively deeper model without suffering stability issues.

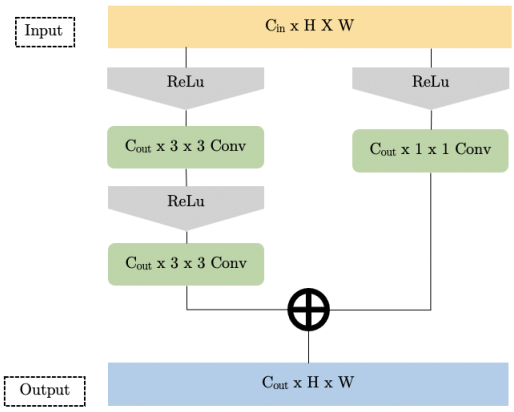

Figure 19: Standard Residual Block

Rather than using transposed convolutions to increase the dimensions of the target image, DRDD makes use of upsampling layers within residual blocks. The general idea is that nearest neighbor upsampling followed by two convolutions will more appropriately estimate the mapping between low resolution features and high resolution features. This approach is inspired by the work done in [35] to downscale time evolving temporal fields. The method is also extended to downsampling blocks. The structure of both of these modules can be observed in Figure 20.

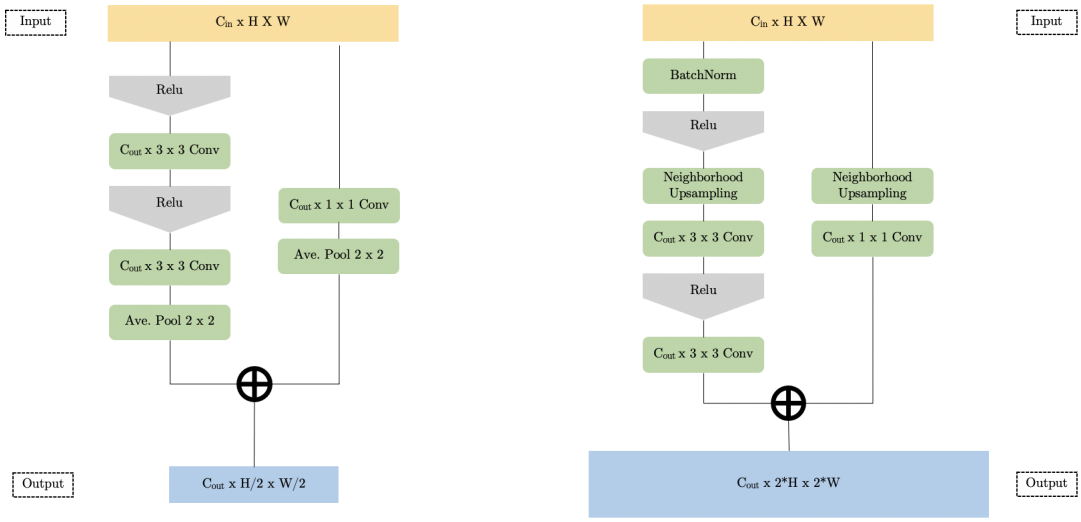

Figure 20: Residual Downsampling Block (left) and Residual Upsampling Block (right)

The DRDD model also makes use of the attention modules introduced in Section 4.1.1.

5.2.2 Critic

In order to improve model stability, DRDD follows the Wasserstein GAN architecture introduced in [2]. In practice, WGANs have been shown to have significantly better performance when compared to GANs. This slight change from the standard GAN architecture means that the evaluating model D that was previously termed the discriminator is instead called a critic. This distinction is due to the fact that the model no longer outputs probabilities.

The critic in the DRDD model makes use of both the high resolution precipitation and the corresponding T length temporal sequence of low resolution precipitation. This is done so that the critic can learn the distinction between fake and real images within the context of temporally correlated low resolution images. This approach is motivated by the temporal downscaling GAN described in [35]. The low resolution sequence is processed through residual downsampling blocks integrated within a stacked GRU sequence. The final hidden feature map from this process is used in conjunction with the high resolution precipitation field to determine the critic score associated with the high resolution precipitation field.

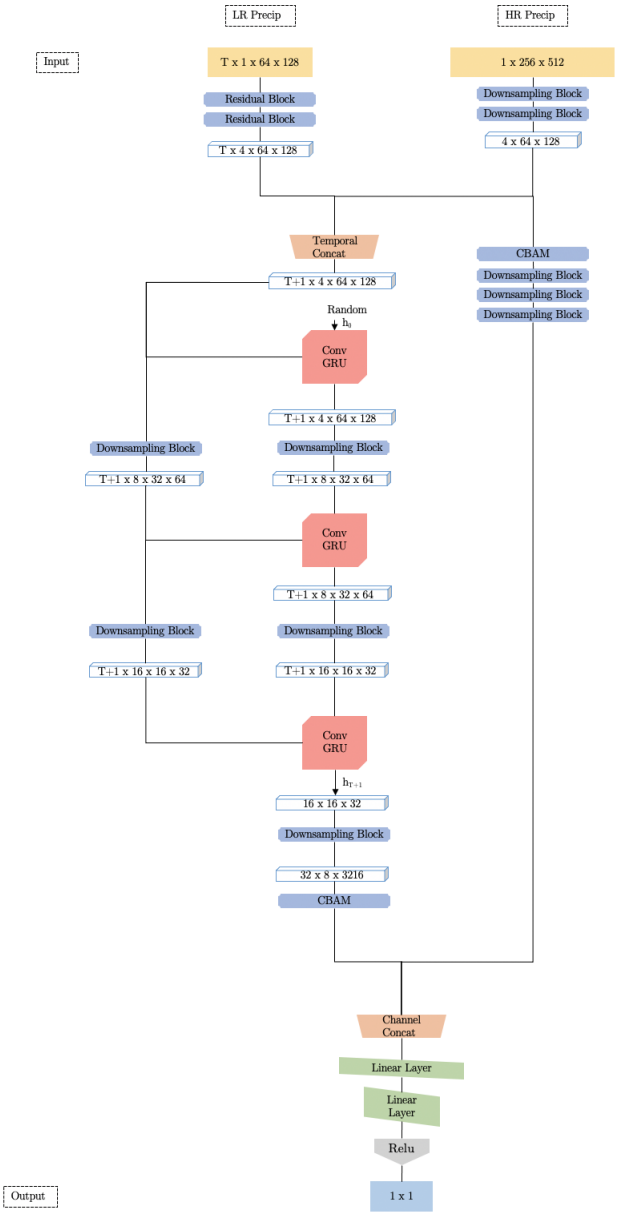

Figure 21: Full critic architecture

The critic optimizes the WGAN loss with gradient penalty as proposed

by [22] which penalizes the gradient of the critic in order to enforce model stability,

$$\begin{aligned}\ell(\theta_D) =& \frac{1}{n}\sum_{i=1}^{n}[D(G(x_{precip,i}, x_{slp,i}, x_{iwv,i}, x_{t2,i}, x_{top,i}), x_{precip,i}) - D(y_i, x_{precip,i})] \\ &+ \sum_{i=1}^{n}(||\nabla_{\hat{y}} D(\hat{y}_i, x_{precip,i})||_2 - 1)^2\end{aligned}$$

where $\hat{y} = \epsilon y_i + (1-\epsilon)G(x_{precip,i}, x_{slp,i}, x_{iwv,i}, x_{t2,i}, x_{top,i})$ and $\epsilon \sim \mathcal{U}(0,1)$.

5.2.3 Generator

Much like FALDD, the generator of the DRDD model uses convolutional networks to prime the input and extract feature maps. However, instead of inception modules, residual networks are used to extract high dimension features from low resolution precipitation, integrated water vapour, surface level pressure, temperature and elevation.

The feature maps of the climate variables are then concatenated and passed through an a temporal upsampling scheme. This series of modules is inspired by the stacked GRU architecture used in [48] for precipitation forecasting. The upsampling modules are inter-weaved into the stacked GRU architecture in order to give the recurrent network the ability to learn temporal dependencies as the precipitation field expands spatially.

As in [35], all of the convolutional GRU layers use a transformation of a single time step as the initial hidden state (h_0) of the recurrent layer in order to produce hidden representation (h_T). Both h_0 and h_T are then passed through a residual upsampling block before being used by next convolutional GRU layer in the stack. The upsampled previous initial state serves as the

initial state for this GRU higher up on the stack and the upsampled previous terminal state serves as input.

After some repetitions of the GRU-upsampling scheme, the representations of the climate variables are collated with a representation of the elevation map and further processed in order to generate the final output.

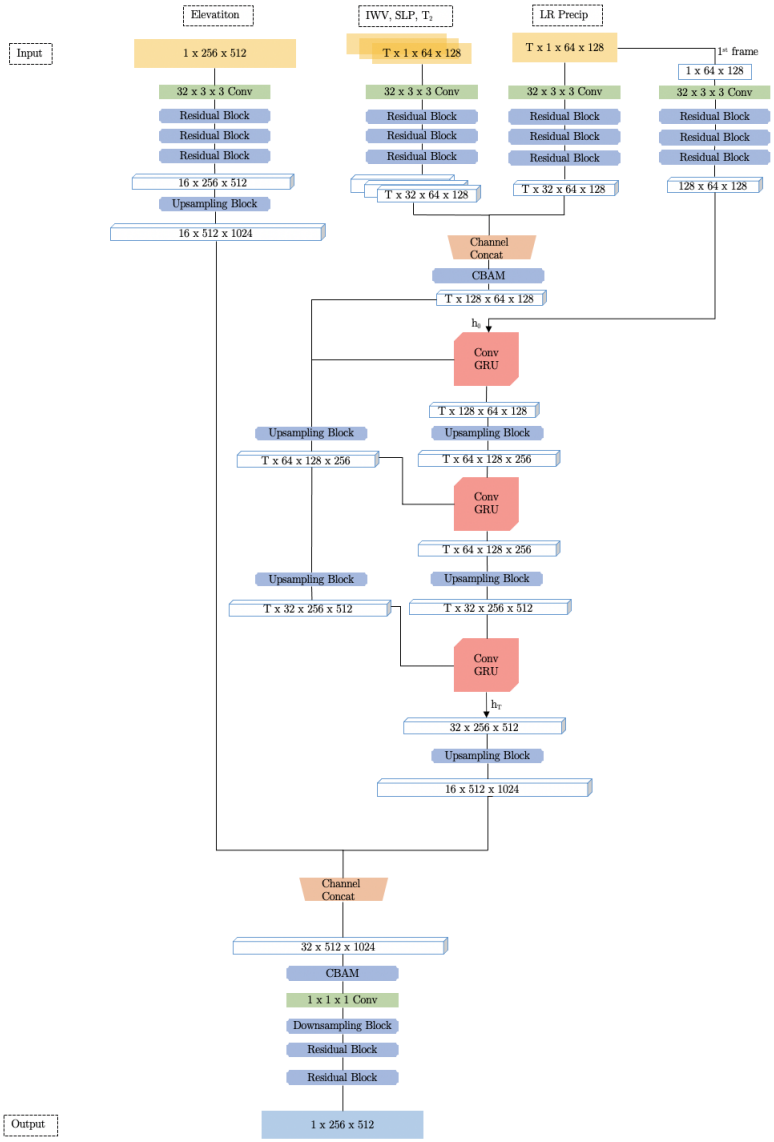

Figure 22: Full generator architecture

As with FALDD, the generator loss is simply a sum of the adversarial loss and the content loss. However, due to convergence difficulties the content

loss is calculated using the ℓ_2 loss rather than the ℓ_1 loss. This decision has negative downstream effects on the generated climate fields but greatly improved the model stability.

$$\ell(\theta_G) = -\frac{\lambda}{n}\sum_{i=1}^{n} D(G(x_{precip,i}, x_{slp,i}, x_{iwv,i}, x_{t2,i}, x_{top,i}), x_{precip,i}) + \frac{\alpha}{n}\sum_{i=1}^{n} ||G(x_{precip,i}, x_{slp,i}, x_{iwv,i}, x_{t2,i}, x_{top,i}) - y_i||_2$$

After some exploration, the chosen values for the loss weighting parameters were $\lambda = 1$ and $\alpha = 5$.

5.3 Model Training

With the inclusion of the GRUs as temporal modules and an increased depth due to the use of residual networks, both the critic and the discriminator of our proposed model require more parameters to fit than the model from [61].

The proposed model takes about 17 hours to train on a distributing calculations to cluster of 3 NVIDIA V100 Volta GPUs. The evaluation of 3 months of data takes about 1.5 hours using two CPU cores.

6 Evaluation metrics

This thesis uses a suite of metrics used both in the evaluation of hydrological models and image super resolution models. For the definitions of the metrics below, we will assume that $y_i \in \mathbb{R}^d$ is the vector representation target variable of interest and $G(x_i)$ represents the generated model estimate for y_i for time sequence $i = 1, \ldots, n$. More specifically, y_{ij} corresponds to observed (or simulated) high resolution precipitation at time i and location j. In contrast, $G(x_i)_j$ corresponds to downscaled estimation of precipitation at time i and location j. The predictor x_i here is overloaded to include both the low resolution precipitation, and the climate variables $x_{SLP,i}, x_{IWV,i}, x_{T2,i}$ and $x_{TOP,i}$.

6.1 Mean Squared Error

The mean squared error (MSE) is a standard metric which is used to evaluate the Euclidean distance between model predictions and the true values.

$$MSE = \frac{1}{nd} \sum_{i=1}^{n} \sum_{i=j}^{d} (y_{ij} - G(x_i)_j)^2$$

6.2 Pearson's Correlation Coefficient

Pearson's correlation coefficient ρ is used to measure the linear relationship between two variables.

$$\rho(G(x_i), y_i) = \frac{\sum_{j=1}^{d} (G(x_i)_j - \overline{G(x_i)})(y_{ij} - \bar{y_i})}{\sqrt{\sum_{j=1}^{d} (G(x_i)_j - \overline{G(x_i)})^2 \sum_{j=1}^{d} (y_{ij} - \bar{y_i})^2}}$$

Where $\overline{G(x_i)}$ and $\bar{y_i}$ represent the mean of the high resolution precipitation at time i and the mean of the true high resolution at time i respectively taken over spatial locations $j = 1, \ldots, d$.

The mean Pearson correlation between RCM model outputs and the GAN predictions is used as a metric to assess model performance.

6.3 Nash-Sutcliffe Efficiency

The Nash-Sutcliffe efficiency (NSE) is a metric used to compare the performance of a predictive hydrological model against the mean of the observed hydrological event. At a particular location j, it can be calculated as

$$NSE(G(x_-)_j, y_{-j}) = 1 - \frac{\sum_{i=1}^{n}(G(x_i)_j - y_{ij})^2}{\sum_{i=1}^{n}(y_{ij} - \bar{y_j})^2}$$

As can be seen from the formula, if the hydrological model performs as well as the mean for the prediction of precipitation, the NSE will be 0. If it performs better, the NSE will approach 1. Finally, if the model is significantly worse than the observed mean, the NSE will approach $-\infty$. For communication of results, we rescale the NSE to [0,1] by using the normalized NSE:

$$NNSE = \frac{1}{2 - NSE}$$

The mean of the NNSE is reported on as a metric to assess model performance.

6.4 Peak Signal-to-Noise Ratio

The peak signal-to-noise ratio is used to compare the quality between an image and its approximation through a weighting of maximum intensity of

the signal with the mean squared error between the two images. In our case, that corresponds to weighting maximum simulation of precipitation by the mean difference between GAN precipitation predictions and RCM simulations.

$$PSNR(G(x_i), y_i) = 20 \log_{10} \left(\frac{\max(y_i)}{\frac{1}{d} \sum_{j=1}^{d} (G(x_i)_j - y_{ij})^2} \right)$$

This metric is widely used in the evaluation of image compression and super-resolution models. The mean PSNR is used as a metric for GAN evaluation.

6.5 Structural Similarity Index

As the name implies, the structural similarity index (SSIM) assesses the similarities between two images through a system that compares their properties of luminance, contrast, and structure. Let $z, u \in \mathbb{R}^d$ be the vector representation of two images, the previously mentioned image properties are defined by [63] as follows:

$$\mu_z = \frac{1}{d} \sum_{j=1}^{d} z_j \qquad \text{(luminance)}$$

$$\sigma_z = \left(\frac{1}{d-1} \sum_{i=1}^{d} (z_i - \mu_z)^2 \right)^{1/2} \qquad \text{(contrast)}$$

$$\sigma_{zu} = \frac{1}{d-1} \sum_{i=1}^{d} (z_i - \mu_z)(u_i - \mu_u) \qquad \text{(structure comp)}$$

Intuitively, luminance can be seen as the mean pixel brightness of an image, contrast can be seen as the standard deviation of pixels within an image and the structure comparison equation can be seen to measure the pixel covariance between two images. All these image properties are then put

together to create corresponding comparison functions

$$l(z,u) = \frac{2\mu_z\mu_u + C_1}{\mu_z^2 + \mu_u^2 + C_1} \qquad \text{(luminance)}$$

$$c(z,u) = \frac{2\sigma_z\sigma_u + C_2}{\sigma_z^2 + \sigma_u^2 + C_2} \qquad \text{(contrast)}$$

$$s(z,u) = \frac{\sigma_{zu} + C_3}{\sigma_z\sigma_u + C_3} \qquad \text{(structure)}$$

$$SSIM(z,u) = [l(z,u)]^\alpha [c(z,u)]^\beta [s(z,u)]^\gamma \qquad \text{(SSIM)}$$

Here, C_1, C_2, C_3 are coefficients for the stability of the ratios used for comparison functions. α, β and γ are weighting parameters for each comparison. In this thesis, we use the mean SSIM as a metric to compare model performance:

$$MSSIM(G(X), Y) = \frac{1}{n}\sum_{i=1}^{n} SSIM(G(x_i), y_i)$$

7 Results

All models are trained on precipitation data from January to September and evaluated on precipitation data from October to December. We report on the choice of a 9 hour time horizon for both the DRDD and the FALDGRU models which corresponds to 3 frames of our mm/3hr data. Smaller time horizons fail to fully leverage the recurrent nature of the model and the temporal correlation degrades as the time interval widens. Due to the inclusion of time, FALDGRU and DRDD are trained and evaluated on 2 fewer samples than FALD.

The results make a promising case for the inclusion of temporal dependencies in dynamical downscaling models. However, as can be seen in Table 2, the method by which these are included makes a definite difference in model performance.

Metric	**FALD**	**FALDGRU**	**DRDD**
MSE	0.262	0.316	**0.233**
MSSIM	**0.946**	0.937	0.938
Mean PSNR	38.082	37.18	**38.50**
Mean NNSE	0.724	0.683	**0.733**
Mean Pearson Correlation	0.703	0.648	**0.730**

Table 2: Model metrics

The results from our reproduction of the FALD model are similar to those reported in [61] and thus should serve as an appropriate proxy to evaluate the

differences between temporal and standard approaches to learned dynamical downscaling.

Overall, the FALDGRU model performed worse than the other two models. Downscaled precipitation fields from this model had visual artifacts previously unseen in precipitation fields. This model has the highest MSE while also having the lowest MSSIM, PSNR, NNSE, and mean correlation. Of course, this makes sense as this model was not built to incorporate temporal dependencies. This experiment serves to show that the temporal correlation within climate fields cannot be modeled through naive integration into current downscaling approaches but must instead be incorporated through original, time-based downscaling models.

In contrast, the DRDD model performs better than the FALDD model on all metrics except the SSIM. The low observed SSIM is very likely due to the choice of content loss in the generator of our proposed model. Due to it's complexity, training DRDD with the ℓ_1 loss led to exploding gradients and a divergence in errors. This is not an unusual occurrence when working with recurrent neural networks [47]. Using the ℓ_2 loss allowed for faster and more stable training since this loss has a stable analytical solution but produced overly smooth realizations of precipitation. This phenomenon has been observed in multiple image generation tasks [71]. Some newer deep learning models for downscaling make use of hydrological metrics as loss functions in lieu of pixel to pixel comparisons [9].

Lower MSE for the DRDD model is expected as the using the ℓ_2 loss equates to minimizing the ℓ_2 distance between the target images and the generated image. However, we also observe higher measures of PSNR and

Pearson correlation with the DRDD model which could signify that this model is better able to capture the variance (around the mean) within the simulated data than the standard FALDD model. The slight difference in the NNSE of the two models of interest also signifies that the proposed model better predicts location-specific precipitation over time. It should however be noted that the difference in the NNSE score of the FALDD and DRDD models is modest.

As mentioned by the authors of [61], metrics like the MSE are not always a definitive measure of performance when it comes to downscaling. For this reason, we also include some downscaled samples from each of the models before moving on to a distributional comparison of both models.

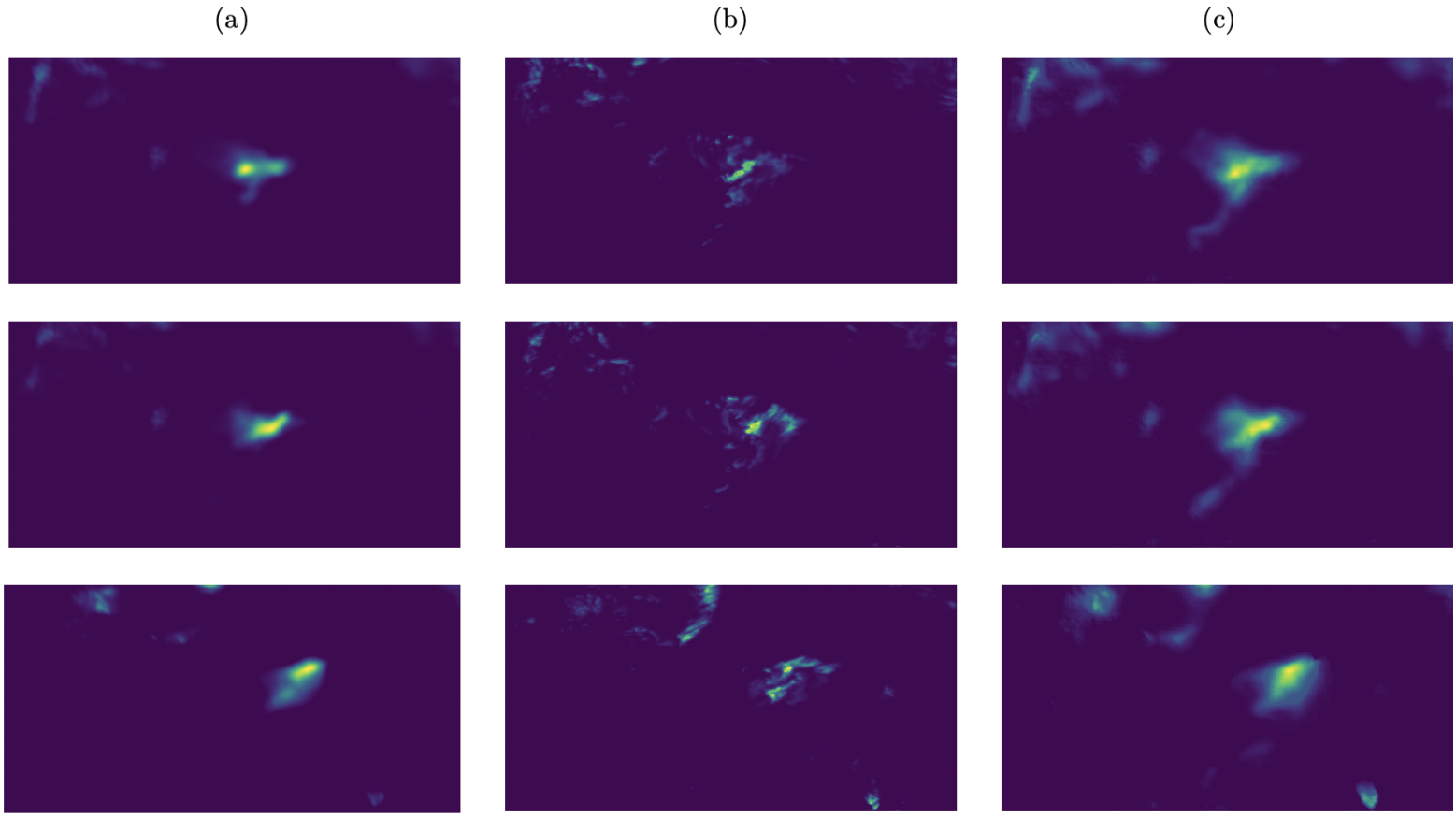

Figure 23: A comparison of downscaled precipitation by FALDD (a) to original RCM output (b) and proposed model DRDD (c) for 3 precipitation events.

In Figure 23, we can see three downscaled samples generated from FALDD, some downscaled samples from DRDD and compare them to the RCM simulations. Firstly, it is immediately apparent that both of the GANs generate low quality, blurry precipitation fields. However, both models are able to recover the general structure of large precipitation fields in the RCM simu-

lation.

From the three precipitation events we are considering, it appears that DRDD is better able to model faint precipitation events which are ignored by the FALDD model. We can also see that the DRDD model is more likely to create artifacts which are not present in the RCM simulation. Both of these occurences are due to the choice of the ℓ_2 loss into the DRDD model as slight deviations are more penalized than in the FALDD model.

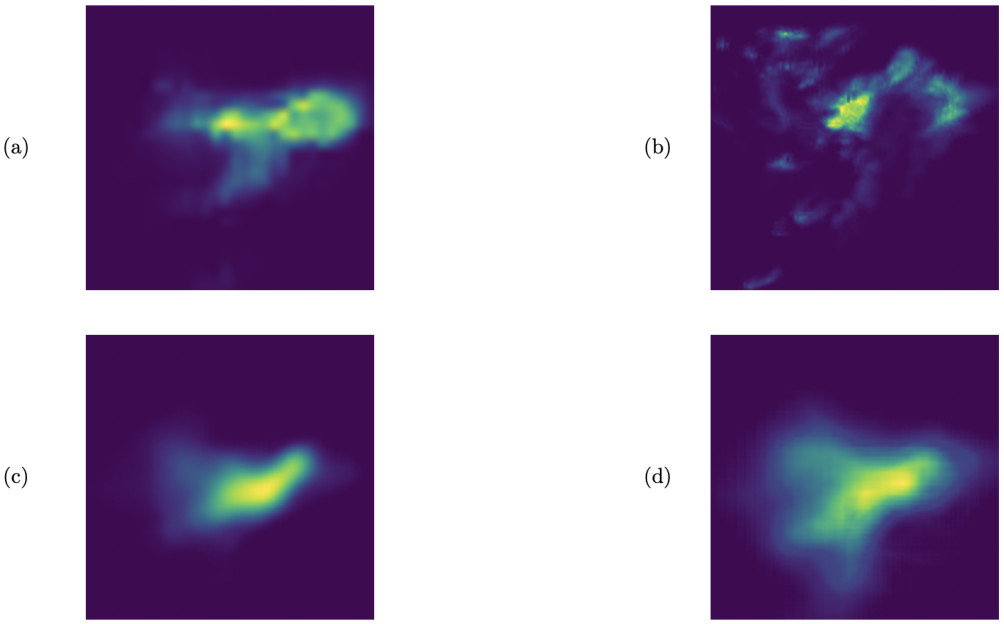

Figure 24: Crops of low resolution precipitation (a) high resolution RCM output (b) downscaled precipitation by FALDD (c) and downscaled precipitation by DRDD (d).

From Figure 24, we can note that both models have difficulties downscal-

ing the complete texture of localized extremes. This is due to the coarseness of the low resolution input which does not reflect the pattern observed in the high resolution image. This observation is certainly a limitation of super-resolution models applied to downscaling as these models will have a hard time learning climate patterns that are not present in the low resolution image.

Finally, the DRDD model seems to generate wide precipitation events covering simulated precipitation whereas the FALDD model generates smaller precipitation events focused around the simulated extremes. To assess the recovery of extremes, we perform a distributional comparison of the data simulated by an RCM and the data downscaled by the generative models.

This comparison evaluates the distribution of the pixel intensity of the images corresponding to precipitation fields simulated by an RCM, FALDD, and DRDD. In this process, the spatial information is ignored in order to gauge the most intense precipitation event each model can simulate.

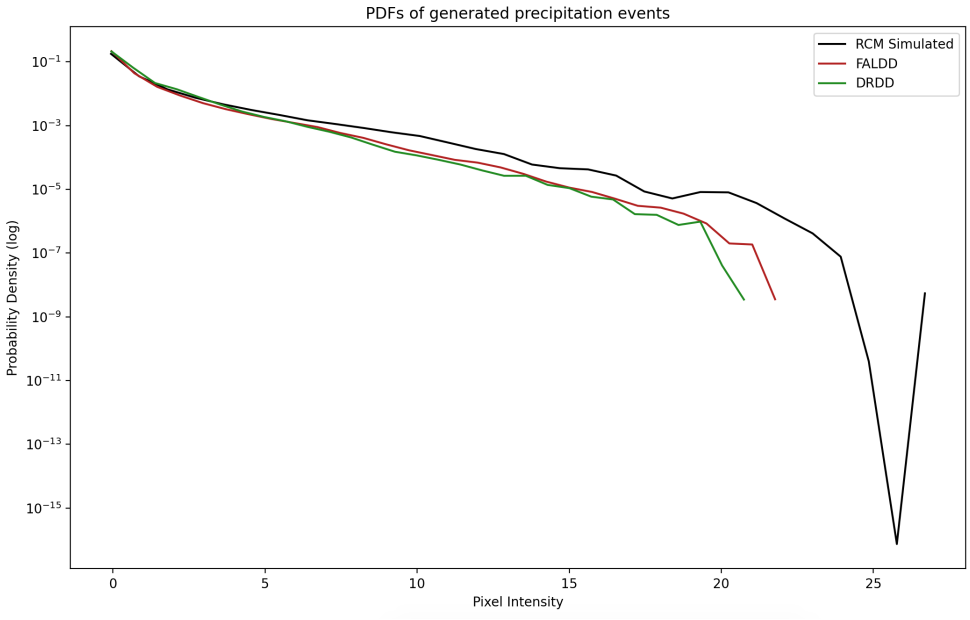

Figure 25: Distributional comparison of RCM, FALDD, and DRDD precipitation fields

As can be seen from Figure 25, neither FALD nor DRDD properly recover the extremes of the images generated by the RCM. Both models have distributions with a flatter right tail than the RCM simulated data. However, there is a slight difference between the pixel distribution of the precipitation of the images generated by FALDD and DRDD. The distribution of FALDD-generated samples is slightly closer to the distribution of RCM samples. This means that FALDD has a slight advantage when it comes to generating distributional-accurate precipitation events.

Upon close scrutiny Figure 25, one can note that the DRDD distribution has a slightly fatter left tail than the other distributions which means the model tends to overestimate low events of precipitation. This is in line with

the previous observation that the DRDD generated data tends to have low-value clouds surrounding precipitation events.

The distributional comparison of the models is done to reflect the model evaluation done in [61]. However, it should be noted however that the distributional comparison does not evaluate the spatial spread of precipitation. Furthermore, since only the pixel encoded data is made available, our evaluation can only evaluate the distributions of pixels which is different from the distribution of precipitation.

8 Discussion and Future Work

From our results, it appears that there are benefits to the strategic inclusion of time dependencies when approaching dynamical downscaling with deep learning models. The DRDD model was better able to recover entire precipitation events simulated by the RCM while the FALDD model focused on areas with high precipitation. However, the model was not able to fully capture the distribution of the simulated RCM data in the upper tail and would therefore not serve in applications concerning the extremes of precipitation. There are a few aspects of the modeling process to which this shortcoming can be attributed.

Firstly, due to restrictions with the data available, our proposed model only leverages simulations that are 3 hours apart. This could mean that the temporal dependence which serves as the anchor to our approach is not as notable as it would be if the data was gathered at smaller time lags. Furthermore, in our experiments, the stacked GRU module was kept relatively simple as the temporal dependence was assumed to be weak. With data that is more temporally correlated, the DRDD model architecture could change to include a wider and more complex stacked GRU module. Given data collected at smaller time intervals, the performance of a deep recurrent dynamical downscaling model could more greatly outperform a standard deep learning model and perhaps better represent a high resolution RCM.

Furthermore, our training data was somewhat small for the use of a generative adversarial network. An issue with GANs is that when training is done with fewer than ideal samples, the discriminator can overfit to training data [31]. Since climate is location specific, standard computer vision

based methods of data augmentation such as image rotation, translation, and distortion cannot be readily utilized for downscaling temporal precipitation fields. Therefore, generating more RCM precipitation data may be necessary to improve model performance.

Another avenue for progress is the sharpness of the generated climate fields. Both FALDD and DRDD generate blurry simulations of precipitation due to the structure of their generator. Some GAN architectures such as [48] have had success generating crisp representations of precipitation using Sub-Pixel Convolutional Neural Network introduced in [53] though both of these models require significantly more data than was available to us in this thesis.

Also, the model training procedure does not actively utilize any known physical properties of climate. We train the model on data from January to September and evaluate it using data from October to December. An ideal training scheme would replicate statistical downscaling approaches like SDSM and create separate models of precipitation for either each month or each season. Increasing the amount of models certainly increases the amount of computational resources and data but it could also increase performance and specificity.

Finally, for ease of comparison with FALDD, DRDD is trained on the clipped data described in Section 2. Using unclipped data could benefit the model into better recovering the extremes of the RCM simulation data. Accurate simulations of extreme occurrences of precipitation tend to be more pertinent to stakeholders as they help in resource allocation and disaster planning. Therefore, a hurdle before dynamical downscaling models such

as DRDD and even FALDD can be used in conjunction with low resolution RCMs as replacement for high resolution RCMs will be their performance on the extremes unclipped data. Until then, models founded on extreme value theory [60, 16, 39, 37] should be considered.

9 References

[1] Gabriele Accarino, Marco Chiarelli, Francesco Immorlano, Valeria Aloisi, Andrea Gatto, and Giovanni Aloisio. Msg-gan-sd: A multi-scale gradients gan for statistical downscaling of 2-meter temperature over the euro-cordex domain. *AI*, 2(4):600–620, 2021.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 06–11 Aug 2017.

[3] Nicolas Ballas, Li Yao, Christopher Joseph Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *CoRR*, abs/1511.06432, 2016.

[4] Jorge Baño-Medina, Rodrigo Manzanas, and José Manuel Gutiérrez. On the suitability of deep convolutional neural networks for continental-wide downscaling of climate change projections. *Climate Dynamics*, 57(11):2941–2951, 2021.

[5] Andrew Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, 1993.

[6] Mohamed Ali Ben Alaya, Fateh Chebana, and Taha Ouarda. Multisite and multivariable statistical downscaling using a gaussian copula quantile regression model. *Climate Dynamics*, 47(5):1383–1397, 2016.

[7] Gerd Bürger, Trevor Murdock, Arelia Werner, Stephen Sobie, and Alex Cannon. Downscaling extremes—an intercomparison of multiple statistical methods for present climate. *Journal of Climate*, 25(12):4366 – 4388, 2012.

[8] Flor Castillo, Arthur Kordon, Jeff Sweeney, and Wayne Zirk. Using genetic programming in industrial statistical model building. In *Genetic Programming Theory and Practice II*. Springer US, Boston, MA, 2005.

[9] Chiranjib Chaudhuri and Colin Robertson. Cligan: A structurally sensitive convolutional neural network model for statistical downscaling of precipitation from multi-model ensembles. *Water*, 12(12), 2020.

[10] Shien-Tsung Chen, Pao-Shan Yu, and Yi-Hsuan Tang. Statistical downscaling of daily precipitation using support vector machines and multivariate analysis. *Journal of Hydrology*, 385(1):13–22, 2010.

[11] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014.

[12] Jacqueline Diaz Nieto and Robert Wilby. A comparison of statistical downscaling and climate change factor methods: Impacts on low flows in the river thames, united kingdom. *Climatic Change*, 69:245–268, 04 2005.

[13] Paul Emberchts, Claudia Klüppelberg, and Thomas Mikosch. *Statistical Methods for Extremal Events*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997.

[14] Maedeh Enayati, Omid Bozorg-Haddad, Javad Bazrafshan, Somayeh Hejabi, and Xuefeng Chu. Bias correction capabilities of quantile mapping methods for rainfall and temperature variables. *Journal of Water and Climate Change*, 12(2):401–419, 4/3/2022 2020.

[15] Gregory Flato, Jochem Marotzke, Babatunde Abiodun, Pascale Braconnot, Sin Chan Chou, William Collins, Peter Cox, Fatima Driouech, Seita Emori, Veronica Eyring, Chris Forest, Peter Gleckler, Eric Guilyardi, Christian Jakob, Vladimir Kattsov, Chris Reason, Markku Rummukainen, Isaac Held, Andrew Pitman, Serge Panton, and Zong ci Zhao. *Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*, chapter 9. Cambridge University Press, Cambridge,United Kingdom and New York, NY, USA, 2013.

[16] Petra Friederichs. Statistical downscaling of extreme precipitation events using extreme value theory. *Extremes*, 13(2):109–132, 2010.

[17] Christian Genest, Ghoudi, and L.-P. Rives. A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika*, 82(3):543–552, 4/14/2022 1995.

[18] Christian Genest and Johanna Nešlehová. Copulas and copula models. In SAbdel El-Shaarawi and Walter Piegorsch, editors, *Encyclopedia*

of Environmetrics, Second Edition, pages 541–553. Wiley, Chichester, 2013.

[19] Andrew Gettelman and Richard B. Rood. *Demystifying Climate Models: A Users Guide to Earth System Models*. Springer, 2016.

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[21] Lucas Gudmundsson, John Bremnes, Jan Haugen, and Torill Engen-Skaugen. Technical note: Downscaling rcm precipitation to the station scale using statistical transformations ; a comparison of methods. *Hydrology and Earth System Sciences*, 16(9):3383–3390, 2012.

[22] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[23] Sahar Hadipour, Shamsuddin Shahid, Sobri Bin Harun, and Xiao-Jun Wang. Genetic programming for downscaling extreme rainfall events. In *2013 1st International Conference on Artificial Intelligence, Modelling and Simulation*, 2013.

[24] Alan Hamlet, Eric Salathé, and Pablo Carrasco. Statistical downscaling techniques for global climate model simulations of temperature and precipitation with application to water resources planning studies. Technical report, Climate Impacts Group, Center for Science in the Earth System, Joint Institute for the Study of the Atmosphere and Ocean, University of Washington, Seattle, 2010.

[25] Michael B. J. Harfoot, Derek P. Tittensor, Sarah Knight, Andrew P. Arnell, Simon Blyth, Sharon Brooks, Stuart H. M. Butchart, Jon Hutton, Matthew I. Jones, Valerie Kapos, Jörn P.W. Scharlemann, and Neil D. Burgess. Present and future biodiversity risks from fossil fuel exploitation. *Conservation Letters*, 11(4), 2018.

[26] Muhammad Z. Hashmi, Asaad Y. Shamseldin, and Bruce W. Melville. Statistical downscaling of watershed precipitation using gene expression programming (gep). *Environmental Modelling & Software*, 26(12):1639–1646, 2011.

[27] Lauren E. Hay, Robert L. Wilby, and George H. Leavesley. A comparison of delta change and downscaled gcm scenarios for three mountainous basins in the united states1. *JAWRA Journal of the American Water Resources Association*, 36(2):387–397, 2000.

[28] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018.

[29] Dae Il Jeong, André St-Hilaire, Taha Ouarda, and Philippe Gachon. A multivariate multi-site statistical downscaling model for daily maximum and minimum temperatures. *Climate Research*, 54:129–148, 09 2012.

[30] Malaak Kallache, Mathieu Vrac, Philippe Naveau, and Paul-Antoine Michelangeli. Nonstationary probabilistic downscaling of extreme precipitation. *Journal of Geophysical Research: Atmospheres*, 116(D5), 2011.

[31] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12104–12114. Curran Associates, Inc., 2020.

[32] Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

[33] John R. Koza. Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, 4(2):87–112, 1994.

[34] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2016.

[35] Jussi Leinonen, Daniele Nerini, and Alexis Berne. Stochastic super-resolution for downscaling time-evolving atmospheric fields with a gen-

erative adversarial network. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9):7211–7223, Sep 2021.

[36] Xin Li and Vladan Babovic. Multi-site multivariate downscaling of global climate model outputs: an integrated framework combining quantile mapping, stochastic weather generator and empirical copula approaches. *Climate Dynamics*, 52(9):5775–5799, 2019.

[37] Elizabeth C. Mannshardt-Shamseldin, Richard L. Smith, Stephan R. Sain, Linda O. Mearns, and Daniel Cooley. Downscaling extremes: A comparison of extreme value distributions in point-source and gridded precipitation data. *The Annals of Applied Statistics*, 4(1):484–502, 2010.

[38] Douglas Maraun. Bias correcting climate change simulations - a critical review. *Current Climate Change Reports*, 2(4):211–220, 2016.

[39] Douglas Maraun, Fredrik Wetterhall, Andrew Ireson, Richard Chandler, Elizabeth Kendon, Martin Widmann, Suzanne Brienen, Henning Rust, Tobias Sauter, Matthias Themeßl, Victor Venema, Kwok Chun, Clare Goodess, Richard Jones, Christian Onof, Mathieu Vrac, and Insa Thiele-Eich. Precipitation downscaling under climate change: Recent developments to bridge the gap between dynamical models and the end user. *Reviews of Geophysics*, 48(3), 2010.

[40] John McGregor. Regional climate modelling. *Meteorology and Atmospheric Physics*, 63(1):105–117, 1997.

[41] Roger Nelsen. *Archimedean Copulas*, pages 109–155. Springer New York, New York, NY, 2006.

[42] Roger B. Nelsen. *An Introduction to Copulas*. Springer Science Business Media, LLC, 2^{nd} edition, 2006.

[43] Vahid Nourani, Zahra Razzaghzadeh, Aida Hosseini Baghanam, and Amir Molajou. ANN-based statistical downscaling of climatic parameters using decision tree predictor screening method. *Theoretical and Applied Climatology*, 137(3):1729–1746, 2019.

[44] Stephen Ornes. How does climate change influence extreme weather? impact attribution research seeks answers. *Proceedings of the National Academy of Sciences*, 115(33):8232–8235, 2018.

[45] Yassin Z. Osman and Mawada E. Abdellatif. Improving accuracy of downscaling rainfall by combining predictions of different statistical downscale models. *Water Science*, 30(2):61–75, 2016.

[46] Hamid Pahlavan, Banafsheh Zahraie, Mohsen Nasseri, and Amin Mahdipour Varnousfaderani. Improvement of multiple linear regression method for statistical downscaling of monthly precipitation. *International Journal of Environmental Science and Technology*, 15(9):1897–1912, 2018.

[47] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks, 2012.

[48] Suman Ravuri, Karel Lenc, Matthew Willson, Dmitry Kangin, Remi Lam, Piotr Mirowski, Megan Fitzsimons, Maria Athanassiadou, Sheleem Kashem, Sam Madge, Rachel Prudden, Amol Mandhane, Aidan Clark, Andrew Brock, Karen Simonyan, Raia Hadsell, Niall Robinson, Ellen

Clancy, Alberto Arribas, and Shakir Mohamed. Skilful precipitation nowcasting using deep generative models of radar. *Nature*, 597(7878):672–677, 2021.

[49] Markku Rummukainen. State-of-the-art with regional climate models. *WIREs Climate Change*, 1(1):82–96, 2010.

[50] Dhanapala Sachindra, Kamal Ahmed, Mamunur Rashid, Shamsuddin Shahid, and Chris Perera. Statistical downscaling of precipitation using machine learning techniques. *Atmospheric Research*, 212:240–258, 2018.

[51] Gianfausto Salvadori, Carlo De Michele, Nathabandu T. Kottegoda, and Renzo Rosso. Bivariate analysis via copulas. In *Extremes in Nature: An Approach Using Copulas*, Dordrecht, 2007. Springer Netherlands.

[52] Yingkai Sha, David John Gagne II, Gregory West, and Roland Stull. Deep-learning-based gridded downscaling of surface meteorological variables in complex terrain. part i: Daily maximum and minimum 2-m temperature. *Journal of Applied Meteorology and Climatology*, 59(12):2057 – 2073, 2020.

[53] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. *CoRR*, abs/1609.05158, 2016.

[54] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew

Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[55] Chuyin Tian, Guohe Huang, Yanli Liu, Denghua Yan, Feng Wang, and Ruixin Duan. Long-term maximum and minimum temperature projections over metro vancouver, canada. *Frontiers in Earth Science*, 9, 2021.

[56] Sylwia Trzaska and Emilie Schnarr. A review of downscaling methods for climate change projections: African and latin american resilience to climate change project. *USAID*, 09 2014.

[57] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[58] Niko E. C. Verhoest, Martinus Johannes van den Berg, Brecht Martens, Hans Lievens, Eric F. Wood, Ming Pan, Yann H. Kerr, Ahmad Al Bitar, Sat K. Tomer, Matthias Drusch, Hilde Vernieuwe, Bernard De Baets, Jeffrey P. Walker, Gift Dumedah, and Valentijn R. N. Pauwels. Copula-based downscaling of coarse-scale soil moisture observations with implicit bias correction. *IEEE Transactions on Geoscience and Remote Sensing*, 53(6):3507–3521, 2015.

[59] Mathieu Vrac, Michel Stein, Katharine Hayhoe, and X.-Z. Liang. A general method for validating statistical downscaling methods under future climate change. *Geophysical Research Letters*, 34(18), 2007.

[60] Matthieu Vrac and Philippe Naveau. Stochastic downscaling of precipitation: From dry events to heavy rainfalls. *Water Resources Research*, 43(7), 2007.

[61] Jiali Wang, Zhengchun Liu, Ian Foster, Won Chang, Rajkumar Kettimuthu, and Rao Kotamarthi. Fast and accurate learned multiresolution dynamical downscaling for precipitation, 2021.

[62] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaoou Tang. Esrgan: Enhanced super-resolution generative adversarial networks, 2018.

[63] Zhou Wang, Alan Bovik, Hamid Sheikh, and Eero Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[64] Campbell D. Watson, Chulin Wang, Timothy Lynar, and Komminist Weldemariam. Investigating two super-resolution methods for downscaling precipitation: ESRGAN and CAR. *ArXiv: Atmospheric and Oceanic Physics*, 2020.

[65] Brian White, Alok Singh, and Adrian Albert. Downscaling numerical weather models with gans, 2019.

[66] Robert Wilby, Christian Dawson, and Elaine Barrow. Sdsm — a decision support tool for the assessment of regional climate change impacts. *Environmental Modelling & Software*, 17(2):145–157, 2002.

[67] Robert Wilby, Oliver Tomlinson, and Christian Dawson. Multisite simulation of precipitation by conditional resampling. *Climate Research - CLIMATE RES*, 23:183–194, 04 2003.

[68] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, Cham, 2018. Springer International Publishing.

[69] Andrew Wood, Ruby Leung, Varadharajan Sridhar, and Dennis Lettenmaier. Hydrologic implications of dynamical and statistical approaches to downscaling climate model outputs. *Climatic Change*, 62(1):189–216, 2004.

[70] Alistair Woodward. Climate change: Disruption, risk and opportunity. *Global Transitions*, 1:44–49, 2019.

[71] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2017.

[72] Xiong Zhou, Guohe Huang, Xiuquan Wang, Yurui Fan, and Guanhui Cheng. A coupled dynamical-copula downscaling approach for temperature projections over the canadian prairies. *Climate Dynamics*, 51(7):2413–2431, 2018.

Appendix

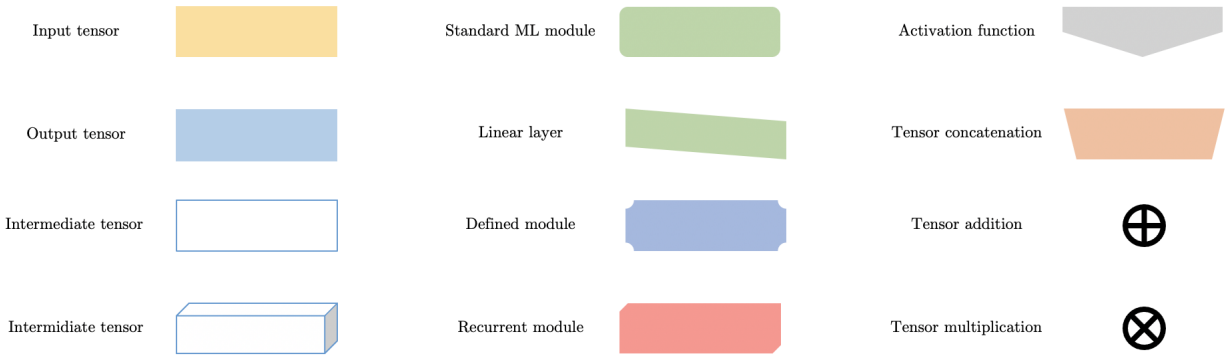

Figure 26: Legend of shapes used in deep learning architecture.