Summer 2024

A NOVEL k-NEAREST NEIGHBORS METHOD BASED ON GENERALIZED FEATURE OPTIMIZATION FOR PRECIPITATION FORECASTING

Sean Guidry Stanteen
University of Texas at Arlington

A NOVEL k-NEAREST NEIGHBORS METHOD BASED ON GENERALIZED FEATURE OPTIMIZATION FOR PRECIPITATION FORECASTING

by

SEAN CHARLES GUIDRY STANTEEN

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy at
The University of Texas at Arlington
August, 2024

Arlington, Texas

Supervising Committee:

Jianzhong Su, Supervising Professor

Hristo Kujouharov

Andrzej Korzeniowski

.

ACKNOWLEDGEMENTS

I thank Dr. Jianzhong Su, Dr. Paul Flannagan, and Dr. John Zhang for their guidance throughout these past five years of research, as well as for providing the materials and data to make it possible. This work was partially supported by NSF grant #1620630, as well as the USDA ARS - Research Apprenticeship Program at University of Texas at Arlington, the USDA NIFA HSI - ALFA-IoT: ALliance For Smart Agriculture in the Internet of Things Era, and the USDA ARS El Reno Oklahoma - Development of an Analogue Weather Forecast Using Machine Learning and K Nearest Neighbor Algorithms.

DEDICATION

I dedicate this dissertation to my mother Nicole Stanteen and father Jason Guidry, as well as my greatest friend Angela Avila, and my other friends and coworkers who stuck by me in this stressful and pivotal time. I would especially like to thank my supervising professor Dr. Su, who saw in me potential in spite of my peculiarities. I doubt it possible for a more patient and guiding professor to exist.

List of Figures

List of Tables

ABSTRACT

A NOVEL k-NEAREST NEIGHBORS METHOD BASED ON GENERALIZED FEATURE OPTIMIZATION FOR PRECIPITATION FORECASTING

Sean Charles Guidry Stanteen, PhD. Mathematics

The University of Texas at Arlington, 2024

Supervising Professor: Jianzhong Su

Abstract

This study introduces a novel k-nearest neighbors (kNN) method of forecasting precipitation at weather-observing stations. The method identifies numerous monthly temporal patterns to produce precipitation forecasts for a specific month. Compared to climatological forecasts, which average the observed precipitation over the prior thirty years, and other existing contemporary iterations of kNN, the proposed novel kNN method produces more accurate forecasts on a consistent basis. Specifically, the novel kNN method produces improved root mean square errors (RMSE), mean relative errors, and Nash-Sutcliffe coefficients when compared to climato- logical and other kNN forecasts at five weather stations in Oklahoma. Rather than looking at the daily data for feature vectors, this novel kNN method takes so many days and evenly groups them, using the resulting average as one feature each. All methods tested were lacking in the ability to forecast wet extremes; however, the novel kNN method produced more frequent high-precipitation forecasts compared to climatology and the two other kNN methods tested.

Contents

CHAPTER 1

INTRODUCTION

1.1 The Benefits and Difficulties with Forecasting Precipitation

Accurate seasonal forecasts would assist stakeholders in minimizing losses that might occur from planting crops that require more precipitation than will occur (Carberry et al. 2000; Jones et al. 2000; Meinke and Stone 2005). Reliable forecasts allow for more dynamic planning and have the potential to increase a field's production capabilities greatly (Bruno Soares and Dessai 2016; Klemm and McPherson 2018; Nicholls 1996). As it stands, the National Oceanic and Atmospheric Association (NOAA), specifically the Climate Prediction Center (CPC), provides seasonal forecasts for precipitation in the form of probabilistic distributions of 30-day and 90-day totals. However, these forecasts are not useful for agricultural stakeholders, due to the limited predictability and inadequate spatial scale they present (Garbrecht and Schneider 2007; Klemm and McPherson 2018; Schneider and Wiener 2009). As such, the exploration of different forecasting methods is warranted.

The issues with forecasting precipitation are twofold. To begin with, precipitation is one of the most volatile weather phenomena in the world. Of course, it is likewise not unpredictable. Several methods of forecasting using differential equations have come about that are able to accurately predict precipitation. However, these methods do so by taking in many variables such as sea surface temperatures, humidity, and solar radiation, among others. While they're useful, these variables are not available to many smaller stations, and while some of those stations may be able to begin collecting them, it would take several decades for them to amass the data typically used for such forecasts. As well, such forecasting is only accurate in the short term, typically dropping off after only forecasting five days ahead (Chakraborty 2010). In the interim, and for stations which lack the means to collect such data, an alternative is necessary which makes do with the data already available to most stations across the country.

1.2 Machine Learning

A machine learning algorithm is an algorithm which analyses data input by a user to create a prediction of the outcome caused by those inputs. While several forms of machine learning exist, the relevant category for this paper is *supervised learning*, in which a computer is given several sets of inputs (or predictors, often represented as vectors in the real space) already correlated to their "correct" predictands by a human. The computer then uses these to "learn" the patterns (if any) of the predictors and predictands, and uses them to predict the outputs of inputs not yet seen by the computer. (El Mrabet et al. 2021).

Examples of such include support vector machine (Cortes and Vapnik 1995), where for regression a hyperplane with the maximal margin between the training set's predictors' vectors is found to help classify inputs given later. When applied to regression, the method is instead referred to as support method regression (Drucker et al. 1996). Given the inputs (predictor and predictand) $(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}$ for $i = 1, ..., l$, $b \in \mathbb{R}$, some arbitrary $C > 0$, and the inner product function $\langle \cdot, \cdot \rangle : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ we optimize

$$\text{Minimize: } \frac{1}{2}\langle w, w \rangle + C \sum_{i=1}^{l} |\xi_i + \xi_i^*|$$

$$\text{Subject to: } \begin{cases} y_i - \langle x_i, w \rangle - b \leq \epsilon + \xi_i \\ \langle x_i, w \rangle + b - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

to get $w \in \mathbb{R}^p$. For any newly introduced predictor $x \in \mathbb{R}^p$, $\langle x, w \rangle + b$ is the resulting prediction.

Long short-term memory (Hochreiter and Schmidhuber 1997) is a method of recurrent learning (that is a method of machine learning which updates its parameters at every step). Let $m, n, p, T \in \mathbb{Z}^+$ be the batch size, number of features, number of units in a cell, and number of time steps desired respectively. Given matrices $x_t \in \mathbb{R}^{m \times n}$ (the input at time step t), the input weight matrices $W_i, W_f, W_g, W_o \in \mathbb{R}^{n \times p}$, recurrent weight matrices $R_i, R_f, R_g, R_o \in \mathbb{R}^{p \times p}$, and biases $b_i, b_f, b_g, b_o \in \mathbb{R}^p$, along with hidden and cell states $h_0, c_0 \in \mathbb{R}^{m \times p}$, then for $t = 1, ..., T$

$$\begin{aligned} i_t =& \quad \sigma\left(W_i x_t + R_i h_{t-1} + b_i\right) \\ f_t =& \quad \sigma\left(W_f x_t + R_f h_{t-1} + b_f\right) \\ g_t =& \quad \tanh\left(W_g x_t + R_g h_{t-1} + b_g\right) \\ o_t =& \quad \sigma\left(W_o x_t + R_o h_{t-1} + b_o\right) \\ c_t =& \quad f_t \circ c_{t-1} + i_t \circ g_t \\ h_t =& \quad o_t \circ \tanh(c_t) \end{aligned}$$

where $\circ$ is the Hadamard product (that is the element-wise multiplication of vectors/matrices from the same vector space) and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function applied element-wise to its matrices. The hidden states h_t for $t = 1, ..., T$ are the resulting output. The purpose then is to give an initial set of training input matrices to optimize the input weight and recurrent weight matrices and biases.

Finally, random forests (Ho 1995) are an extension of random decision trees. In the context of regression, N trees are created by bootstrapping the predictors for each tree. Trees are built using specific methods such as CART (Choubin et al. 2018) and iterated through until the N subsets have given their predictions. Those predictions are then averaged together to give the random forest's ultimate prediction.

While all these are potentially useful algorithms to forecast precipitation, seeing as all are based in detecting patterns in the real space, they are not the primary focus of this paper for reasons to be discussed briefly in the beginning of the next chapter, along with the used method's unique difference which lead to its prioritization.

CHAPTER 2

k-NEAREST NEIGHBORS

k-nearest neighbors (kNN) is a non-parametric method of pattern classification/regression, respectively denoting data with categorical labels (dry, wet, stormy, etc.) and numerical labels (0 mm, etc.). This algorithm utilizes a set of predictands, each of which have a set of numerical representations of certain properties called predictors or *features*, stored in a *feature vector*. We have our operational (target) data whose feature vector is known, but its predictand is not (in our case because the target has not yet occurred). The objective of kNN is to find which k labeled feature vectors are most similar to, or "nearest," the target's feature vector. Once those labeled feature vectors are found, their labels are used to predict the predictand of the target feature vector. For classification, whichever label is most represented by the k nearest neighbors is predicted to apply best to our target feature vector. For regression, a linear combination of the k nearest neighbors' predictands is taken and used as the average prediction, which is what was done for this study.

The foundations of kNN were first laid in 1951 Fix and Hodges (1989). Initially, the method was conceived as the compliment to Fix and Hodges's naive kernel estimate, which is discussed in both the original paper and the commentary released shortly thereafter (Fix and Hodges (1989); Silverman and Jones (1989)). It wasn't until the next year that the researchers would introduce the terminology which gives the method its modern name (Fix and Hodges (1952)). In summary, given two distributions F and G with an equal number of p-dimensional samples ($p = M + 1$ where M is the number of predictors), a p-dimensional sample with an unknown distribution (our target feature vector), and an odd positive integer k (to prevent a tie), the distances of all samples with known distributions from that of the one whose distribution is unknown are found. Whichever of the two distributions owns a majority of the nearest k samples, it is predicted the target belongs to that distribution. In addition to this, they gave two important findings; the sample size has a negative correlation with the probability of error, while the number of features in a feature vector has – at least, at its simplest form possible – a positive correlation with the probability of error. In 1967, Cover and Hart proved the upper bound on the method's probability of error was twice that for Bayes's method when $k = 1$ (Cover and Hart (1967)). While using only the single nearest neighbor k=1 may make the most intuitive sense, it runs the risk of allowing noise or outliers to have an undue effect, especially when several distributions are available, or the data in question are particularly volatile. In 1970, Hellman published his proposed solution and brought us one step closer to the method we know today; the (k, k') nearest neighbor method (Hellman (1970)). Given two positive integers k', k such that $k' < k$, the target is predicted to have a label if at least k' of the k nearest neighbors share it. This allowed for sample sets to be composed of samples from more than just two distributions, while retaining the requirement that a significant number of the neighbors should be the same.

As valuable as all this information is, its usefulness is mitigated by one simple fact; an estimation of future precipitation is desired, rather than a classification that denotes a range of quantities. For this method to be of any use, it needs to be adaptable to not just a continuous space, but a time series. Fortunately, in 1968 Cover was able to extend his upper error bound from classification to regression. He showed that the large sample risk of the nearest neighbor method was at least less than or equal to half the risk presented by probability distributions such as normal, uniform, and Gaussian (Cover (1968)).

As such, several attempts at producing an algorithm to predict precipitation with this method have been made by several individuals (e.g. Bannayan and Hoogenboom (2007); Bannayan and Hoogenboom (2008); Zhang (2004); Yates et al. (2003); Huang et al. (2017)), by comparing feature vectors of daily precipitation and other variables to predict the precipitation of the next day, adding that prediction to the data set and using it to predict the day after, and so on until we have a prediction of the desired length. Attempts at reproducing this have found that while it can be technically accurate on a day-to-day basis, the actual quantity of forecasts can be unduly impacted by extreme bouts of precipitation in the past. Indeed, looking to prior attempts, while kNN's ability to predict temperature is impressive, its predictions of precipitation, while promising, can leave something to be desired (Bannayan and Hoogenboom (2008)). While they can certainly present impressive results, said results tend to only be for a single year, which leaves it open to whether the method is of quality, or simply a favorable year for the method. Given how volatile precipitation can be, improvements are needed for kNN to be useful for precipitation prediction.

This paper proposes a novel solution. Rather than looking at the daily data for feature vectors, for some target day t, this novel kNN method takes so many days and evenly groups them, using the resulting average as one feature each, which is referred to as the (a, b) pair. In this context, $a \in \mathbb{Z}^+$ is the number of groups, while $b \in \mathbb{Z}^+$ is the number of days in each group. Given this method of building a feature vector, the novel KNN results will be used to forecast the total precipitation that occurs 30 days after t.

While the methods mentioned in Section 1.2 all have merit to them and can be used effectively to create forecasts, their complexity compared to kNN would dramatically increase the time needed to train them as later described in Algorithm 1.

2.1 Predictands and Features

Let W be a set where each $w \in W \subseteq \mathbb{R}^q$ is a *predictand*, where q is the dimensionality of the space, or equivalently the number of entries in the predictand. In a temporal context such as the one being discussed, this could be the data from a number of days past the given date times the number of variables being forecasted. However, only the average precipitation a number of days after the target date is being considered for this discussion. As such, W is simply a subset of the real space $\mathbb{R}$.

Each predictand has a set of predictors, or *features*, which is a real numerical representation of its properties. For the weather, this may be the precipitation, the maximum and minimum temperature, etc., all of which being assigned to their predictand.

A *feature vector*, therefore, is a vector containing some or all of these features. In other words, if we were to have p of these features available for each label, the feature vector of $w \in W$ would be $v \in V \subseteq \mathbb{R}^p$. Finally, let Φ be a set of ordered pairs of predictands and features vectors, or for the aforementioned v and w, $(v, w) \in \Phi \subseteq V \times W$. It is expedient to give the elements of Φ a name, however what that name would be can vary depending on the features and predictands being used. Given the fact that, as will be discussed in Chapters 3 and 4, features are drawn from the days prior to a specific date and predictands from the data post that date, elements of Φ shall be referred to as *dates.* Moreover, for the sake of intuitive correlation, if $d \in \Phi$ is a date, then $\vec{d} \in V$ is its feature vector and $\mathbf{d} \in W$ its predictand. In other words,

$$(\vec{d}, \mathbf{d}) = d \in \Phi. \tag{2.1}$$

All feature vectors have a predictand associated with them, however these predictands are not always known. This can be due to a myriad of reasons, however in the context of meteorology and time series in general it is quite simple. As was just stated, the predictand of a date comes from the data of days after it. If those days have not yet occurred, their data, and by extension the predictand of the date, cannot be known. Given the temporal nature of the data being used, feature vectors with known predictands ($\mathbf{h} \in W$) shall be called *historical feature vectors* ($\vec{h} \in V$), while a feature vector associated with an unknown predictand ($\mathbf{t} \in W$) is called a *target feature vector* ($\vec{t} \in V$). The associated dates then are the *historical dates* ($h \in \Phi$) and *target date* ($t \in \Phi$) respectively. It is in this situation that k-nearest neighbor is used to compare historical feature vectors to the target to find which are the most similar, and then forecast the unknown predictand as being some linear combination of the k known predictands whose feature vectors are closest to the target feature vector. How this is determined will be discussed in the next section.

2.2 Euclidean Distance

This brings up the question of how one knows how close, or how similar, two feature vectors are. The typical way to figure this out is to take the Euclidean distance of the two vectors, meaning for dates t^* and h, the function $e : \Phi \times \Phi \to \mathbb{R}$

$$e(t^*, h) = \|D(\vec{t^*} - \vec{h})\|_2 \tag{2.2}$$

is the distance between their respective feature vectors $\vec{t^*}$ and $\vec{h}$. D meanwhile is a diagonal matrix such that allows certain features to be of greater import. Specifically, this is done by assigning such features greater values. In the event that some features are considered more important predictors than others, a diagonal matrix $D \in \mathbb{R}^{p \times p}$ where for $i = 1, ..., p,\ D_{ii} > 0$ and the trace of D, that is

$$tr(D) = \sum_{i=1}^{p} D_{ii} = 1. \tag{2.3}$$

The purpose of D is to allow values with greater the i^{th} feature to determining the target predictand. However, as it is done in contemporary writings to which this novel method shall be compared (Bannayan and Hoogenboom 2008,) $D_{ii} = \frac{1}{p}$ for $i = 1, .., p$.

2.3 Finding Neighbors

Finally, we can calculate the nearest neighbors. Let t^* be the target date, and for each historical date $h \in \Phi$, let

$$e_h = e(t^*, h) \tag{2.4}$$

be the distance between t^* and h. Once these are calculated, sort them from least to greatest and take note of the k lowest e_h. Equivalently, we can say we are looking for the k dates with the lowest distance from t^*. In either case, these are our k nearest neighbors. What is done from here depends on if the method is being used for classification or regression.

Since the purpose in this case is to forecast via regression, we shall take a weighted sum of the k nearest neighbors (Dudani 1976). For $i = 1, ..., k$, if h has the i^{th} lowest distance from t^*, let $E_i = \frac{1}{e_h}$ and $k_i = \mathbf{h}$. Then, with

$$E = \sum_{i=1}^{k} E_i \tag{2.5}$$

our forecast of the precipitation to occur after t^* shall be

$$K_{t^*} = \sum_{i=1}^{k} \frac{E_i k_i}{E} \tag{2.6}$$

The inverse of e_h is taken to ensure that neighbors closest to t^* have the greatest impact on its forecast. In the event that $e_h = 0$ for some date h, the respective E_i are set to the inverse of the lowest positive distance available. In the unlikely event that all the lowest k $e_h = 0$, $E_i = 1$ for $i = 1, ..., k$.

CHAPTER 3

GEM

The information found in the previous chapter is a useful step-by-step demonstration of kNN, however it leaves several things unclear, especially for forecasting. This requires a clarification of what variables are used as features in the feature vectors, how we decide the span of time included, and how we decide whether a certain amount of data either introduces too much noise or removes too much information.

3.1 (a, b) Pairs

To begin with, we must identify the features used for the feature vectors. Let $P : \Phi \to \mathbb{R}^p$ be a function which maps a date $d \in \Phi$ to a vector containing all the normalized data points which occurred on that date. As an example, our method uses precipitation, minimum temperature, and maximum temperature, so $P(d)$ is a vector containing the normalized precipitation, minimum temperature, and maximum temperature to occur on date d. It is with P as well as what follows that we shall build our feature vectors.

Before that, however, it is best to allow the following abuse of notation. For $d \in \Phi$ and $r \in \mathbb{Z}^+$, allow $d - r$ to be the date which occurs r days before d. So if d were January 2, 2024, $d - 1$ would be January 1, 2024, $d - 2$ would be December 31, 2023, etc.

Finally, the feature vectors. Generalized feature vectors (GFV) are denoted generally by the ordered pair (a, b), which in this context can be read as "a spans of b days," where a span is simply a length of time. For example, a week is one span of seven days, so $(1, 7)$, three weeks is $(3, 7)$, one month is $(1, 30)$, thirty days is $(30, 1)$, etc. A GFV is then formed by averaging the normalized data of the b days for each of the a spans. Suppose we have an ordered pair $(a, b) \in \mathbf{\Omega} \subseteq \mathbb{Z}^+ \times \mathbb{Z}^+$ and a date $d \in \Phi$.

$$\vec{d_y} = \frac{1}{b}\sum_{j=0}^{b-1} P(d - (j + yb)) \text{ for } y = 0, ..., a - 1. \tag{3.1}$$

With the individual vectors $\vec{d_y}$ we are able to build the feature vector of date d, that being

$$\vec{d} = [\vec{d_0}, ..., \vec{d_{a-1}}]. \tag{3.2}$$

A visualization of this process is given in Figure 3.1.

3.2 Scoring and Selecting

Now that we have discussed how the dates' feature vectors are constructed, we can explain how we decide which to use. Let T_{t^*} contain an arbitrarily chosen set of dates from the same calendar month as t^* such that their predictands are known. Most simply, this would be . Then, let $K_t^{(a,b)}$ be the forecast generated for date $t \in T_{t^*}$ when its feature vectors are built using $(a, b) \in \mathbf{\Omega}$, and let C_t be the climatological forecast for t (the method to find which being described in Chapter 4). Finally, let Ω be the piece-wise function

$$\Omega(t, a, b) = \begin{cases} 1 & |K_t^{(a,b)} - \mathbf{t}| < |C_t - \mathbf{t}| \\ 0 & \text{otherwise} \end{cases}. \tag{3.3}$$

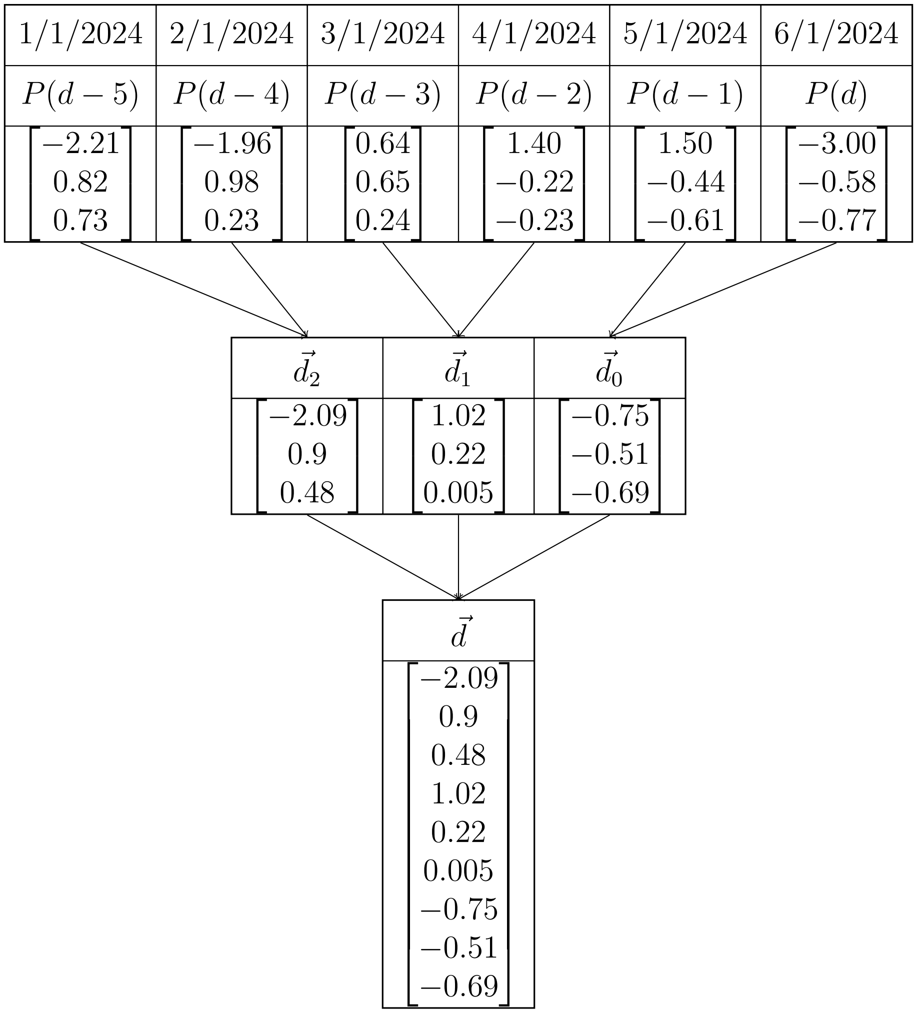
Put simply, Equation 3.3 takes note of if the forecast produced by kNN using that (a, b) has a lower absolute error than (is a superior forecast to) C_t. Which pair is used is then selected

```
Data: Target Date $t^*$, Training Target Dates $T_{t^*}$, Potential Analogues $\Phi$, Potential
      GFVs $\mathbf{\Omega}$, Days Forward $f$
Result: Ideal GFV $(a^*, b^*)$, Forecast $K_{t^*}$
Function GEM-$k$NN-Forecast($t$, $(a,b)$, $\Phi$, $\mathbf{\Omega}$, $f$):
    foreach $t \in T_{t^*}$ do
        $C \leftarrow$Control($t$, $f$)  // Climatological control to compare results to
        $\Phi_t \leftarrow \Phi$ with dates after $t$ removed
        foreach $(a,b) \in \mathbf{\Omega}$ do
            $K \leftarrow k$NN-Forecast($t$, $(a,b)$, $\Phi_t$, $f$)            // see Algorithm 2
            if $|K - \mathbf{t}| < |C - \mathbf{t}|$ then
                Give $(a,b)$ 1 point          // Reward $(a,b)$ if more accurate than
                 control
            end
        end
    end
    $(a^*, b^*) \leftarrow (a,b) \in \mathbf{\Omega}$ with the most points
    $K_{t^*} \leftarrow k$NN-Forecast($t^*$, $(a^*, b^*)$, $\Phi$, $f$)            // see Algorithm 2
    return $(a^*, b^*), K_{t^*}$
Procedure Control($d$, $f$):
    $C \leftarrow \vec{0}$
    for $i \leftarrow 1, ..., 30$ do
        $d' \leftarrow$ same calendar day as $d$, but $i$ years ago
        $C \leftarrow C+$Prediction($d'$, $f$)                          // see Algorithm 3
    end
    $C \leftarrow \frac{1}{30}C$
    return $C$
```

Algorithm 1: Pseudocode for performing GEM-kNN.

```
Data: Target Date $t$, GFV $(a, b)$, Potential Analogues $\Phi$, Days Forward $f$
Result: Forecast $K_t^{(a,b)}$
Function $k$NN-Forecast($t$, $(a, b)$, $\Phi$, $f$):
    $k \leftarrow \left\lfloor \sqrt{|\Phi|} \right\rfloor$    // Square root of the number of potential analogues,
     rounded down
    $\vec{t} \leftarrow$Feature Vector($t$, $(a, b)$, $f$)    // Target feature vector
    foreach $h \in \Phi$ do
        $\vec{h} \leftarrow$Feature Vector($h$, $(a, b)$, $f$)    // Historical feature vector
        $e_h \leftarrow \|\vec{t} - \vec{h}\|_2$
    end
    $E \leftarrow 0$
    for $i \leftarrow 1, ..., k$ do
        $h \leftarrow$ The potential analogue with the $i^{th}$ lowest $e_h$
        $k_i \leftarrow$Prediction($h$, $f$)
        if $e_h = 0$ then
            $E_i \leftarrow 1$
        else if any prior $e_h = 0$ then
            $E_i \leftarrow 0$
        else
            $E_i \leftarrow \frac{1}{e_h}$
        end
        $E \leftarrow E + E_i$
    end
    $K_t^{(a,b)} \leftarrow 0$
    for $i \leftarrow 1, ..., k$ do
        $K_t^{(a,b)} \leftarrow K_t^{(a,b)} + \frac{E_i}{E} k_i$    // Build weighted mean
    end
    $K_t^{(a,b)} \leftarrow \frac{1}{f} K_t^{(a,b)}$
    return $K_t^{(a,b)}$
Procedure Feature Vector($d$, $(a, b)$, $f$):
    $\vec{d} \leftarrow []$    // Start with empty feature vector
    $r \leftarrow 0$
    for $y \leftarrow 0, ..., a - 1$ do
        $\vec{d_y} \leftarrow \vec{0} \in \mathbb{R}^p$    // Set $\vec{d_y}$ to $\vec{0}$ (zero vector)
        for $j \leftarrow 0, ..., b - 1$ do
            $\vec{d_y} \leftarrow \vec{d_y} + P(d - r)$    // See current Section for $P$
            $r \leftarrow r + 1$
        end
        $\vec{d_y} \leftarrow \frac{1}{b} \vec{d_y}$
        Append $\vec{d_y}$ to $\vec{d}$
    end
    return $\vec{d}$
```

Algorithm 2: Pseudocode for performing kNN with a fixed GFV.



1/1/2024	2/1/2024	3/1/2024	4/1/2024	5/1/2024	6/1/2024
$P(d-5)$	$P(d-4)$	$P(d-3)$	$P(d-2)$	$P(d-1)$	$P(d)$
$\begin{bmatrix}-2.21\\0.82\\0.73\end{bmatrix}$	$\begin{bmatrix}-1.96\\0.98\\0.23\end{bmatrix}$	$\begin{bmatrix}0.64\\0.65\\0.24\end{bmatrix}$	$\begin{bmatrix}1.40\\-0.22\\-0.23\end{bmatrix}$	$\begin{bmatrix}1.50\\-0.44\\-0.61\end{bmatrix}$	$\begin{bmatrix}-3.00\\-0.58\\-0.77\end{bmatrix}$

$\vec{d_2}$	$\vec{d_1}$	$\vec{d_0}$
$\begin{bmatrix}-2.09\\0.9\\0.48\end{bmatrix}$	$\begin{bmatrix}1.02\\0.22\\0.005\end{bmatrix}$	$\begin{bmatrix}-0.75\\-0.51\\-0.69\end{bmatrix}$

$\vec{d}$
$\begin{bmatrix}-2.09\\0.9\\0.48\\1.02\\0.22\\0.005\\-0.75\\-0.51\\-0.69\end{bmatrix}$

First, for the date d (which in this case is January 6^{th}) $P(d-r)$ are found for $r = 0, ..., 5$ since our feature vector requires $3 \times 2 = 6$ days. Since 2 is the second value of the ordered pair, we take the average of the first two vectors ($P(d)$ and $P(d-1)$) to get $\vec{d_0}$. The same is done with $P(d-2)$ and $P(d-3)$ to get $\vec{d_1}$, and $P(d-4)$ and $P(d-5)$ to get $\vec{d_2}$. These vectors are then joined together to get $\vec{d}$. All values are normalized.

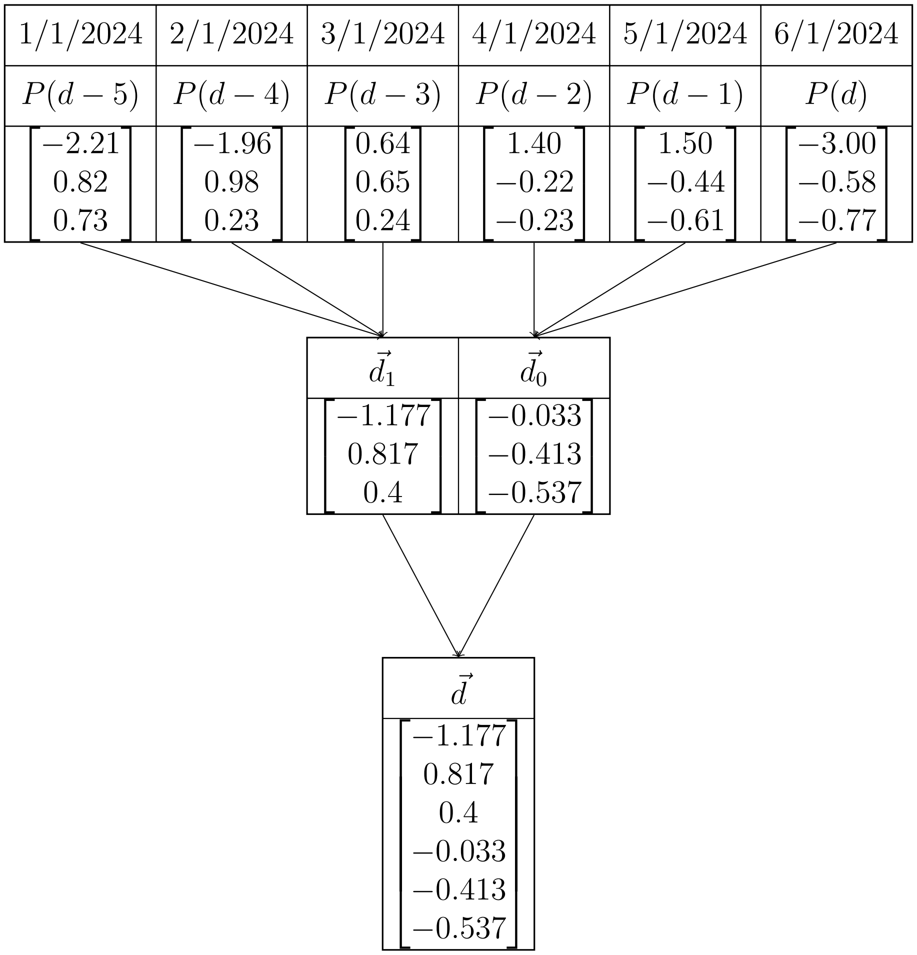
Figure 3.1: Example of GFV (3, 2).

as follows:

$$(a^*, b^*) = \arg \max_{(a,b)\in\mathbf{\Omega}} \sum_{t\in T} \Omega(t, a, b). \tag{3.4}$$

Once (a^*, b^*) is chosen, use them to build the feature vectors for t^* and the historical dates and use the k nearest neighbors to calculate K_{t^*}. Or put simply,

$$K_{t^*} = K_{t^*}^{(a^*,b^*)}. \tag{3.5}$$



1/1/2024	2/1/2024	3/1/2024	4/1/2024	5/1/2024	6/1/2024
$P(d-5)$	$P(d-4)$	$P(d-3)$	$P(d-2)$	$P(d-1)$	$P(d)$
$\begin{bmatrix}-2.21\\0.82\\0.73\end{bmatrix}$	$\begin{bmatrix}-1.96\\0.98\\0.23\end{bmatrix}$	$\begin{bmatrix}0.64\\0.65\\0.24\end{bmatrix}$	$\begin{bmatrix}1.40\\-0.22\\-0.23\end{bmatrix}$	$\begin{bmatrix}1.50\\-0.44\\-0.61\end{bmatrix}$	$\begin{bmatrix}-3.00\\-0.58\\-0.77\end{bmatrix}$

$\vec{d_1}$	$\vec{d_0}$
$\begin{bmatrix}-1.177\\0.817\\0.4\end{bmatrix}$	$\begin{bmatrix}-0.033\\-0.413\\-0.537\end{bmatrix}$

$\vec{d}$
$\begin{bmatrix}-1.177\\0.817\\0.4\\-0.033\\-0.413\\-0.537\end{bmatrix}$

First, for the date d (which in this case is January 6^{th}) $P(d-r)$ are found for $r = 0, ..., 5$ since our feature vector requires $2 \times 3 = 6$ days. Since 3 is the second value of the ordered pair, we take the average of the first three vectors ($P(d)$, $P(d-1)$, and $P(d-2)$) to get $\vec{d_0}$. The same is done with $P(d-3)$, $P(d-4)$, and $P(d-5)$ to get $\vec{d_1}$. These vectors are then joined together to get $\vec{d}$. All values are normalized.

Figure 3.2: Example of GFV (2, 3).

```
Data: Date $d$, Days Forward $f$
Result: Forecast $\mathbf{d}$
Function Prediction($d$, $f$):
    $\mathbf{d} \leftarrow \vec{0} \in \mathbb{R}^p$
    for $r \leftarrow 1, ..., f$ do
        $\mathbf{d} \leftarrow \mathbf{d} + P(d+r)$
    end
    $\mathbf{d} \leftarrow \frac{1}{f}\mathbf{d}$
    Remove unimportant forecasted variables from $\mathbf{d}$
    return $\mathbf{d}$
```

Algorithm 3: Pseudocode for the prediction given by the given date d for the next f days.

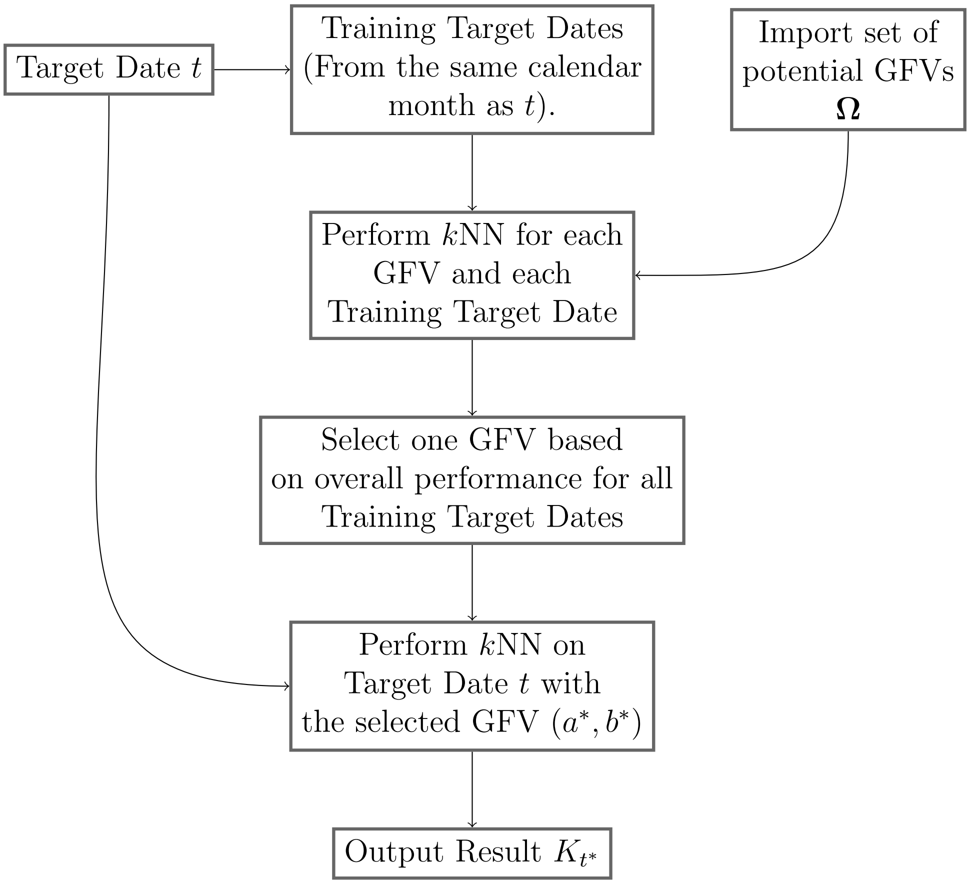

Target Date t
Training Target Dates (From the same calendar month as t).
Import set of potential GFVs $\mathbf{\Omega}$
Perform kNN for each GFV and each Training Target Date
Select one GFV based on overall performance for all Training Target Dates
Perform kNN on Target Date t with the selected GFV (a^*, b^*)
Output Result K_{t^*}

Figure 3.3: Chart of Good Enough Method (GEM) Progression.

CHAPTER 4

OTHER METHODOLOGY

The initial implementation of our methodology comes courtesy of Gerrit Hoogenboom et al. (Bannayan and Hoogenboom 2008,) which is much more straightforward. In fact, its implementation can be explained using the same (a, b) used in the previous section. For a given target date t^*, let $b = 1$, and let a be the number of days between t^* and January 1 of the same calendar year, inclusive. This means that for January 2, $a = 2$, for February 2, $a = 33$, etc. Under these conditions, we shall say that the forecasted precipitation after target date t^* given by Hoogenboom et al.'s method, denoted S_{t^*}, is

$$S_{t^*} = K_{t^*}^{(a,1)} \tag{4.1}$$

where $K_{t^*}^{(a,1)}$ is as it was defined in Section 3.2. As for climatology, for the target day t^* (for this example, April 15^{th}, 2022) let c_{fr} denote the total precipitation f days after April 15^{th}, $2022 - r$ for $r = 1, ..., 30$. Then, $c = \frac{1}{30} \sum\limits_{r=1}^{30} c_{fr}$ is called the climatological forecast, which is used as a baseline to evaluate the usefulness and skills of each forecast. The purpose of these controls is to show the necessity of an adaptable methodology (Wolpert and Macready 1997) by contrasting GEM with its more rigid counterparts.

CHAPTER 5

EVALUATION

5.1 Test

This method was tested on five stations from Oklahoma, USA, each with at least 90 years of precipitation, minimum temperature, and maximum temperature data, where missing values were filled using data from adjacent stations. The annual average of each station, as well as the database period is given in Table 5.1. Lahoma, Weatherford, and Chandler are all located in central Oklahoma, with southerly flow driving increases in humidity and related precipitation during the warm season and producing less harsh winter temperatures. Hooker, the northwest-most station located in the Oklahoma panhandle, is an arid region, where precipitation comes in bursts due to isolated thunderstorm activity and infrequent convective systems. Idabel is meanwhile the southeast-most station and likewise the most humid and whose larger precipitation totals come primarily from organized convective systems synoptic wave activity. For target dates, the $9^{th}, 12^{th}, 15^{th}, 18^{th}$, and 21^{st} of each month of the 5 most recent years available (2004-2008 for all but Lahoma, which is 2002-2006) were used to ensure diverse results for validation. This gave 25 test target dates per month per station, 300 per station, or 1,500 in total. The k nearest neighbors were identified for each target date to forecast the average precipitation over the next 30 days.

As mentioned in the introduction, a large volume of tests were completed to provide robust quantification of the skill of this method compared to climatology and other kNN methods. Before this, however, a GFV must be selected. For each month the same 5 days ($9^{th}, 12^{th}, 15^{th}, 18^{th}$, and 21^{st}) were used in the 45 years prior to the testing years for validation, resulting in 225 validation target dates per month used to select a GFV via GEM from the following set

$$\{(a,b) \mid a \in \{1,2,...,180\},\ b \in \{1,2,...,20\},\ 30 \leq ab \leq 365\}$$

which was then used for the precipitation forecasts found in the results.

Table 5.1: Station Data Table.

Station	Latitude (N)	Longitude (W)	Precipitation (mm)	Min. Temp. (°C)	Max. Temp. (°C)	Database Period
Chandler	35”42’	-96”53’	886.5	9.49	22.78	1902-2009
Hooker	36”51’	-101”12’	473.8	5.32	22.02	1907-2009
Idabel	33”53’	-94”49’	1175.1	10.93	24.96	1907-2009
Lahoma	36”5’	-96”55’	790.9	9.14	22.15	1909-2007
Weatherford	35”31’	-98”42’	754.4	8.85	22.7	1905-2009

Physical location, historical annual average of minimum temperature (Min. Temp.), maximum temperature (Max. Temp.), total annual precipitation, and the period of data available for each study site.

To test our method’s reliability, several indicators of quality were used.

5.2 Nash-Sutcliffe

A comparison to the overall observed precipitation average was implemented, with said average being

$$\bar{T}^* = \frac{1}{|T^*|} \sum_{t^* \in T^*} \mathbf{t}^* \tag{5.1}$$

where $|T^*|$ is the cardinality of T^*. Then, the *Nash-Sutcliffe coefficient* (NS), given by

$$NS = 1 - \frac{\sum\limits_{t^* \in T^*} (K_{t^*} - \mathbf{t}^*)^2}{\sum\limits_{t^* \in T^*} (\bar{T}^* - \mathbf{t}^*)^2} \tag{5.2}$$

is a comparison with the observed average, with a range $(-\infty, 1]$. If NS is negative, the forecasting method tested performed poorly, the observed average being generally superior. If $0 \leq NS < 1$, the method performed at least as well as the observed average, with greater NS correlating to greater performance. An $NS = 1$ means the method forecasted all target dates perfectly, meaning it had an RMSE of 0.

5.3 Root Mean Square Error (RMSE)

The *root mean squared error* (RMSE) is used to assess the quality of forecasts. Let e be the set of forecasts (either those found with GEM, SotA, typical, or climatology) and o the corresponding set of observed data.

$$RMSE = \sqrt{\sum_{t^* \in T^*} \frac{(K_{t^*} - \mathbf{t}^*)^2}{|T^*|}} \tag{5.3}$$

5.4 Significance Testing

Significance testing was performed via the Mann-Whitney U test (Mann and Whitney 1947) to ensure that the differences between the results were significant. To perform this test, let $(P, \leq)$ be an ordered set such that

$$P = \{K_{t^*} | t^* \in T^*\} \bigcup \{\mathbf{t}^* | t^* \in T^*\} \tag{5.4}$$

with P's elements being ordered such that $P_i \leq P_j$ if $i \leq j \ \forall i, j$.

5.5 Reliability Graphs

And finally, reliability graphs (Roberts 2023) were composed to look at how consistently the predictions performed well. To create these graphs, all forecast data, and their corresponding observed values, are binned into discrete 10 mm wide bins. Then, the average values of both the binned forecasts and their corresponding observations are taken. The final reliability graph depicts the average forecast versus average observed values for those forecasts to show how "reliable" the forecast system is in reproducing specific binned values of data. Observed averages below the forecast average represent an overestimation for that specific forecast bin and when the observed average is higher than the forecast average that means the forecast system is underestimating the observed values for that range of forecast values.

Table 5.2: Example of how each point of the reliability graphs is calculated. The forecasts that belong to each bin are averaged together and compared to the average of their corresponding observed data. All values in this table are given in mm.

Bins	Forecast Data	Observed Data	Forecast Average	Observed Average
20-30	21, 24, 29, 22, 26	32, 10, 25, 29, 40	24.4	27.2
30-40	32, 35, 37, 38	51, 39, 28, 45	35.5	40.75
40-50	42, 49, 45	32, 44, 53	45.3	43

5.6 Results

All stations achieved superior RMSE, MRE, and NS when using GEM compared to SotA or climatology. This indicates GEM not only produced superior forecasts, but when it did, these forecasts had a non-trivial improvement over climatology.

It should be noted that there is a positive correlation between average total annual precipitation and RMSE, in that for $i = 1, ..., 5$, the station with the i^{th} greatest total annual precipitation also had the i^{th} greatest RMSE. This is the case regardless of the forecasting method used.

Looking at Figures 5.2 and 5.3 gives important insights into these statistical measures. Climatology is extremely prone to underestimation, typically several centimeters short of the observed amounts on higher precipitation days. Using SotA does fix this to an extent, especially regarding the cold season, but it is only when GEM is implemented that more extreme forecasts are made. Unfortunately, this also brings about an opposite problem:

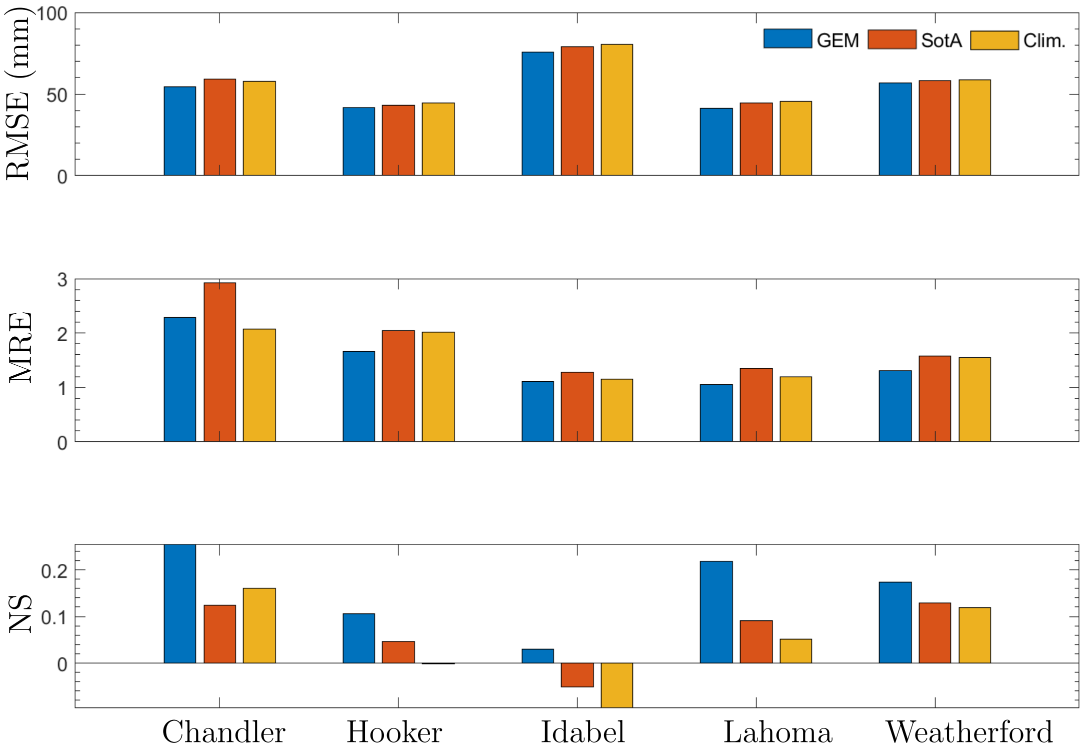
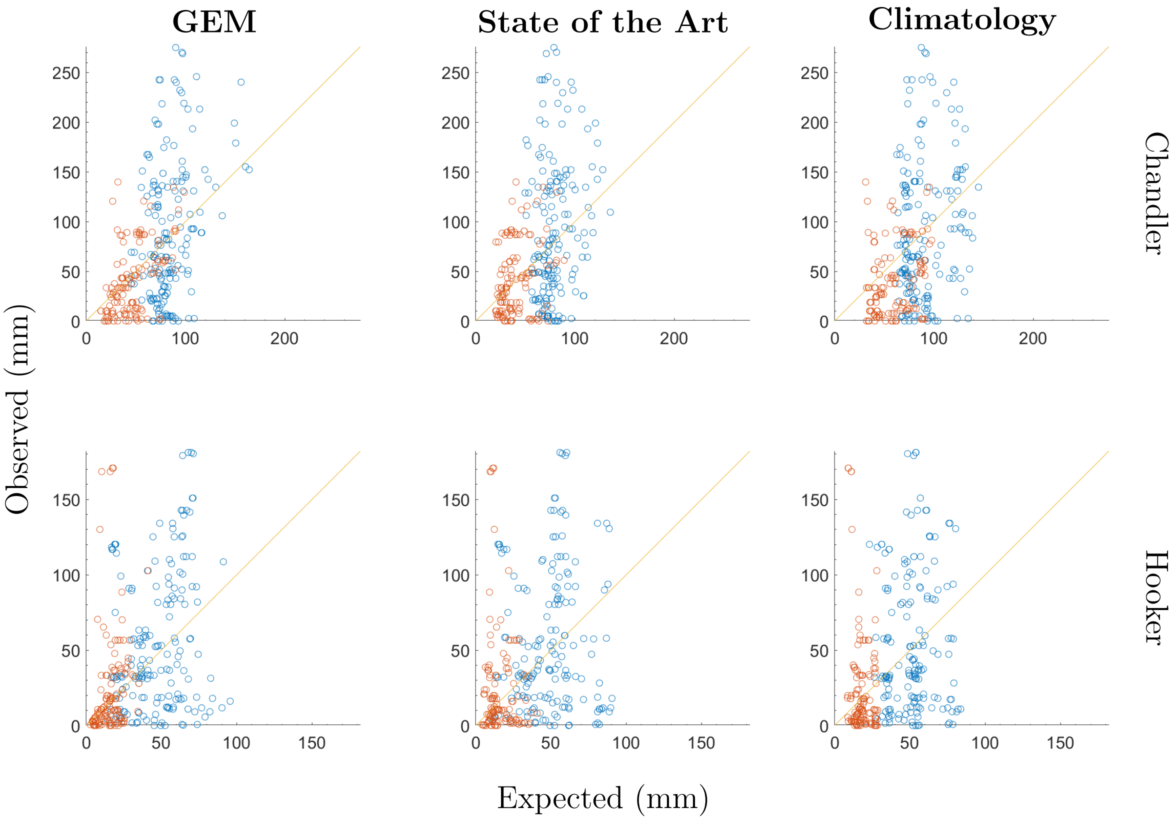


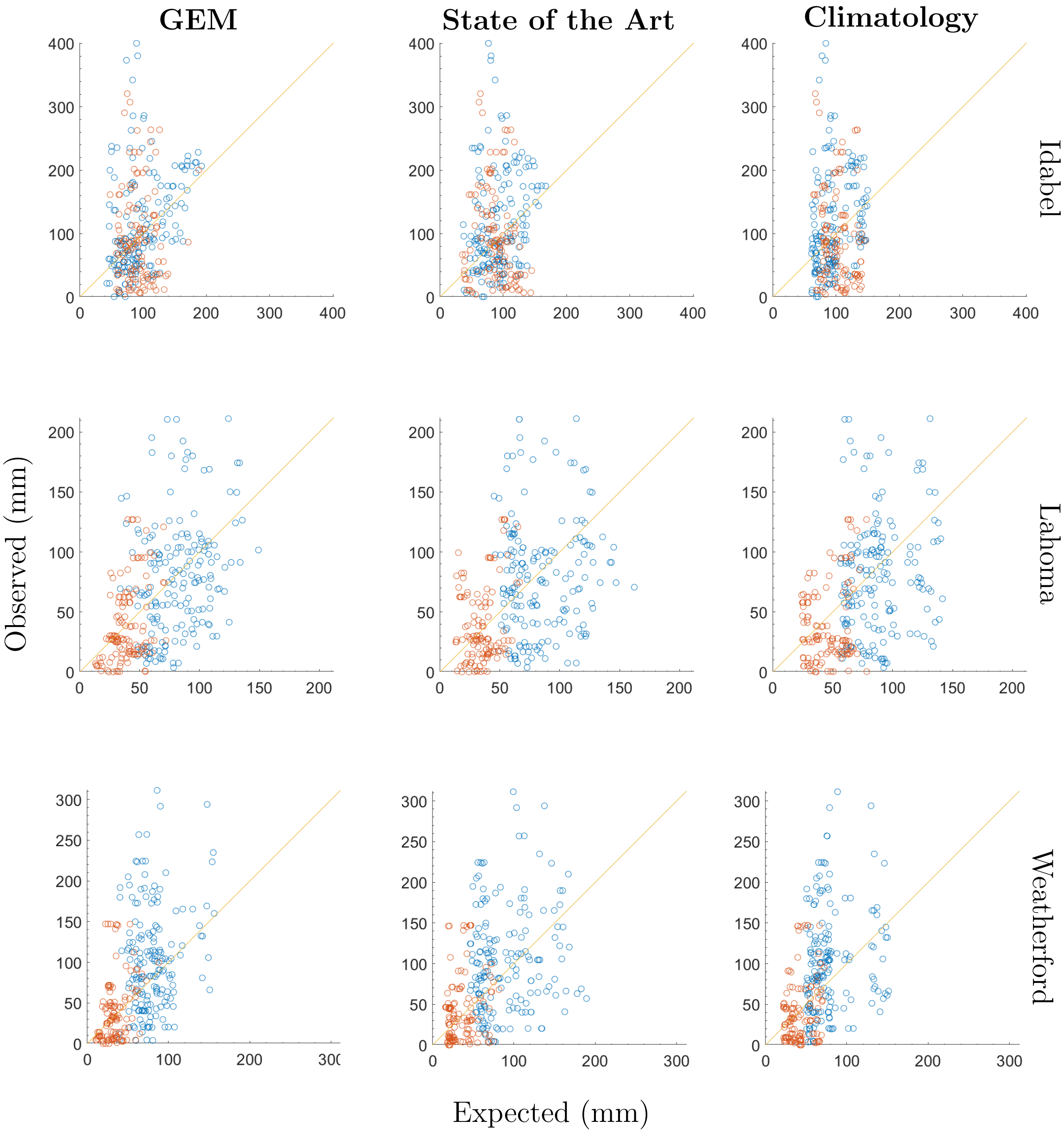
Figure 5.1: RMSE, MRE, NS data presented as a bar graph.

overestimation. Both of these problems are to be expected, seeing as the forecasts are built using weighted averages. As such, while all methods have instances of this, GEM's willingness to make forecasts that exceed the upper bounds of the others' predictions can result in forecasts that go further past lower observations as they do approach higher ones.



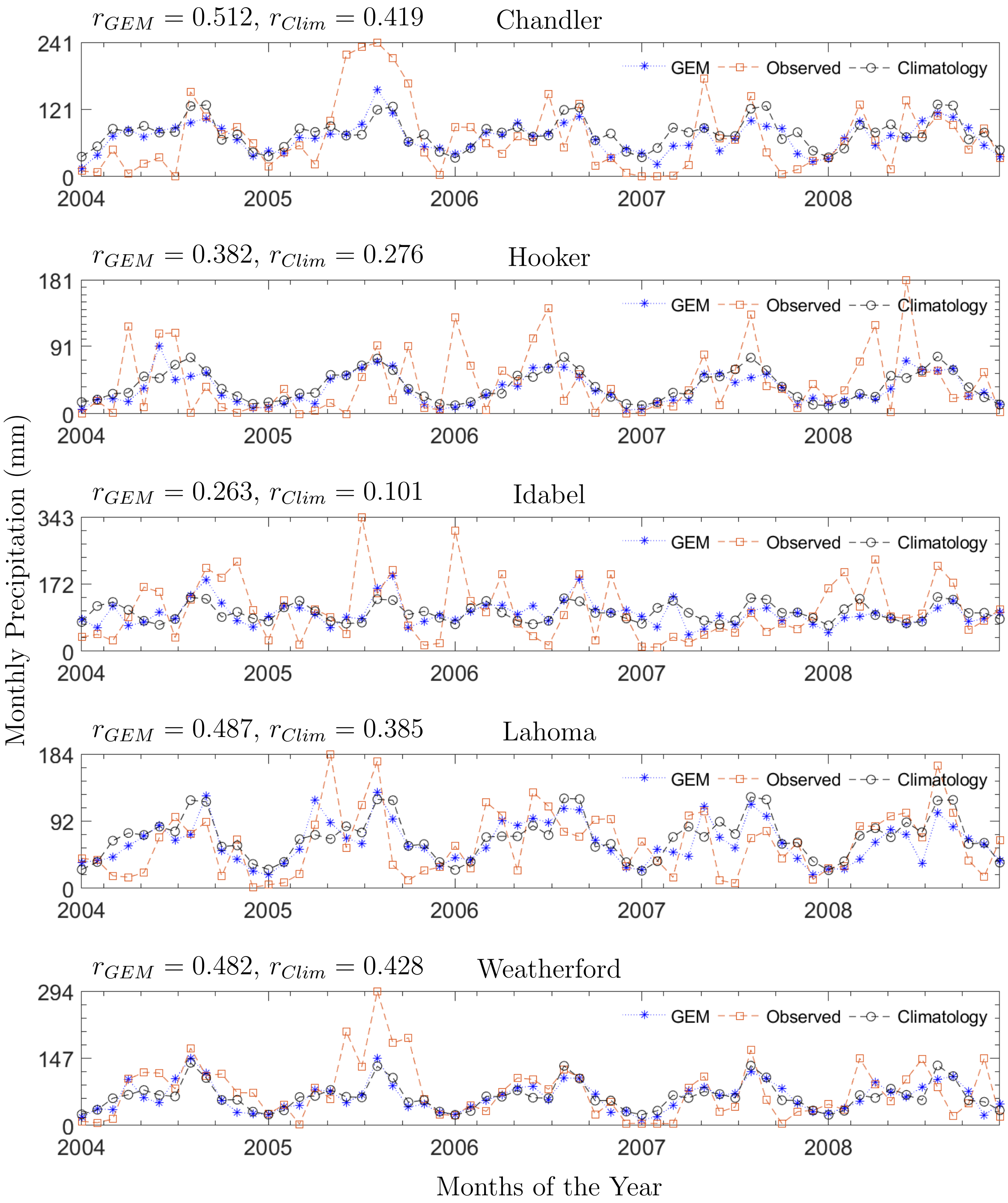
Scatter plots of mean monthly precipitation forecasts using the three methods mentioned in Chandler and Hooker. Red circles are forecasts for March through September (the warm season of the year), and blue circles are for all other months (the cold season).

Figure 5.2: Results Scatter Plots 1



Scatter plots of mean monthly precipitation forecasts using the three methods mentioned in Idabel, Lahoma, and Weatherford. Red circles are forecasts for March through September (the warm season of the year), and blue circles are for all other months (the cold season).

Figure 5.3: Results Scatter Plots 2.



A comparison of forecasts developed by climatology and GEM to the observed monthly precipitation from 1997 to 2006 at Chandler, Hooker, Idabel, Lahoma, and Weatherford. Also gives the correlation coefficients for GEM and climatology to the observed precipitation with $r = \frac{\sum(e_i-\bar{e})(o_i-\bar{o})}{\sqrt{\sum(e_i-\bar{e})^2\sum(o_i-\bar{o})^2}}$ where $\bar{e}$ and $\bar{o}$ are the respective means of the forecast being correlated and the observer precipitation.

Figure 5.4: Extended Results Comparisons.

One can also look to the aforementioned figures for the differences in performance at different times of the year, with red circles indicating the warm season of March through September, and blue being the cold, October through February. During the warm season, GEM is likely to make lower forecasts, regardless of observed precipitation. Interestingly, though, Idabel it is just as likely to over-forecast during the warm season, and in Chandler it rarely under-forecasts at all. For both SotA and climatology, forecasts skew into a mix of over and under-forecasting for both the warm and cold seasons, though much like with GEM, their seasonal forecasts at all stations sans Idabel can be clearly distinguished by the range of values each method was willing to put out.

Figure 5.4, meanwhile, is a demonstration of how GEM (the highest performing of the three kNN implementations) performed in monthly forecasts of the testing data. The purpose of this is to give a visual aid for not only how GEM is meeting individual forecasts, but how well it is able to match the trends of the observed data. For GEM, all stations had a correlation coefficient above 0.25, with the highest being Chandler at 0.512, and the lowest being Idabel at 0.263. Alphabetically, GEM has an RMSE of 54.465, 41.937, 75.829, 41.32, and 56.953 mm for these graphs. With climatology, the resulting correlation coefficients were universally lower, with its highest being Weatherford at 0.428 and its lowest being Idabel at 0.101. Alphabetically, climatology has an RMSE of 59.015, 43.280, 78.937, 44.552, and 58.634 mm for these graphs. By its nature as the 30-year average, the climatological forecast takes on a seasonal cycle, making any similarities to the observed trend coincidental. GEM, however, has a much more interesting relationship. Although it rarely predicts the heavier precipitation totals accurately, GEM's forecasts often will "peak" in comparison to the forecasts made a month prior and after, suggesting a level of sensitivity to the larger precipitation values. Examples of this trend can be seen in most larger precipitation totals, with exceptions for Lahoma. It should be noted that the GEM forecast shown in the above results is a weighted average from the k nearest neighbors identified by the GEM system, thus this represents not a single instance of a precipitation prediction but a smoothed average.

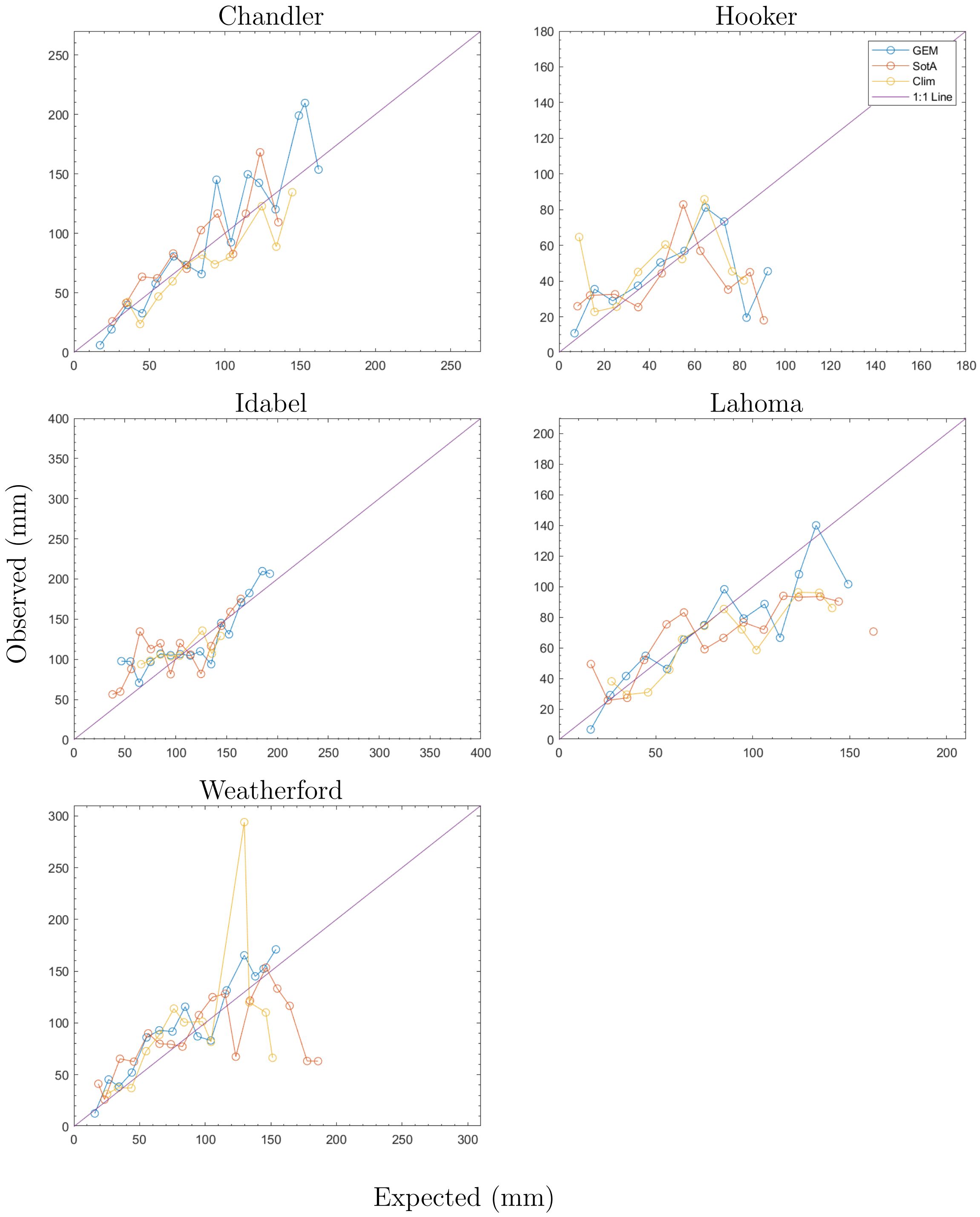


Figure 5.5: Reliability graphs for each of the five stations.

The spread of the nearest neighbor forecast values extends beyond the observed values in most cases (outside of extreme values).

Of the 5 stations, Hooker, Lahoma, and Weatherford's GEM forecasts were found to be significantly different from those of SotA, with p-values of 0.029, 0.019, and 0.002, respectively. Chandler and Idabel, meanwhile, had p-values of 0.800 and 0.144 respectively. When compared with climatology, the differences are more apparent both in Figures and in these statistics, with Chandler, Hooker, Lahoma, and Weatherford having p-values of less than 0.001 while only Idabel had an extreme p-value of 0.947.

Figure 5.5 shows the reliability graphs for each station. For lower precipitation totals, the averaged forecasts and observed precipitations are relatively close. The primary exception to this is Idabel at 30-50mm. Once the expected forecasts pass a certain threshold, however, they are found to be generally lacking. In the cases of Chandler and Weatherford, they tend to underestimate, while Hooker and Lahoma overestimate. Once again, the exception seems to be Idabel, however this should not necessarily be taken as evidence of quality. Looking back to Figure 5.3, while many of the extreme forecasts are accurate, moderate forecasts have several examples of over and under-forecasting to give the results seen in Figure 5.5.

For SotA, some similarities in shape to GEM can be observed in Chandler and Hooker, however it is otherwise quite different, in the cases of Chandler, Idabel, and Lahoma being unable to produce forecasts greater than those of GEM. While it is able to do so in Weatherford, it is only able to make over-forecasts, whereas in Hooker, where both GEM and SotA tend to over-forecast, SotA's are slightly more extreme.

Perhaps unsurprisingly, climatology has the fewest extremes of these three forecasting methods. It will have the lowest range of forecasted precipitation, particularly noticeable in Chandler, Lahoma, and Weatherford, and lacking extreme deviation from the 1:1 line, or at least rarely to the extreme of its kNN counterparts. The main exceptions to this are in Hooker where its lowest forecasts correlate to observed precipitation nearly seven times greater, as well as Weatherford where forecasts around 140mm correlate to observed precipitation nearly double that on average.

CHAPTER 6

DISCUSSION

GEM-kNN, of course, did not come to this point as soon as this research began, and empirical evidence alone is not enough to create a case for its use in an academic context. In this chapter previous iterations of the method shall be discussed briefly, along with the logic behind the conclusion drawn.

Due to its higher volatility, precipitation is especially susceptible to the problems this produces, with extreme errors being far more likely than moderate errors. While GEM certainly shows improvement over SotA and climatological forecasts, it is far from being exceptional regardless of time and location, and past tests could be even less exemplary. A less thorough archive of values from past tests are provided in the appendix. It is important to note several differences from these past tests, namely $NMNSC = \frac{1}{2-NS}$ for the NS established in Section 5.2, and that K_{t^*} was instead defined as

$$K_{t^*} = \sum_{i=1}^{k} \frac{k_i}{i \sum_{j=1}^{k} \frac{1}{j}}.$$

As mentioned, empirical evidence was given in Section 5.6 as to the improvements of GEM over SotA. Here we shall go into the more conjectural nature of our experiment. To begin with, GEM is predicated on the assumption that any (a, b) that could accurately and consistently forecast a weather variable in the past is capable of doing so again with currently available data. According to the “no free lunch” theorem (NFL), (Wolpert and Macready 1997) algorithms are going to perform equally well when their results are compared across all different problems, with the only difference being on which problems they performed well, and which poorly. This theorem is our justification of the results presented in Section 5.6. Looking to climatology and SotA for comparison, climatology is consistent in both the number of potential analogues used, and where these potential analogues are found. It is also the worst performing of the three. Along with that, we have SotA, which has the unusual criteria for what is included in the feature vector of “how many days are there between now and January 1st of this year.” While this is technically variation, it is arbitrary variation. GEM falls in line with the “no free lunch” theorem by acknowledging that a single iteration won’t always work for a forecast, and instead tries to identify which iteration is most likely to work in the present. This claim cannot be presented as anything more than conjecture, however the evidence should be sufficient to not call it baseless.

CHAPTER 7

CONCLUSION AND FUTURE RESEARCH

In this paper, a novel method of performing kNN was introduced and tested for five different stations across Oklahoma, being compared to the climatological average as well as the typical and state of the art methodologies for kNN. This comparison was performed by noting the RMSE, MRE, and NS of all methodologies' forecasts of the total precipitation 30 days after several target dates. As well, statistical significance of the results were measured, and reliability graph were made to determine whether the novel GEM method was not only superior, but distinct to those others.

The GEM method was demonstrated as superior to all other methods in this study. While far from perfect, GEM universally had a lower RMSE and MRE than all other methods, and was the only method to maintain a positive NS across all five stations. The conclusions drawn by GEM were found to be statistically significantly different from those of SotA, which implies that with refinement the program could continue to improve on those results.

In the future, applying GEM to machine learning algorithms aside from kNN would be most pressing. While the efficacy of the method in contrast to SotA and climatology has been demonstrated, other methods of supervised machine learning such as long short-term memory (Hochreiter and Schmidhuber 1997), support vector machine (Cortes and Vapnik 1995), or Random Forests (Ho 1995) have yet to be contrasted with GEM-kNN, or have GEM used in conjunction with them where possible.

Furthermore, climate change, one of the most extreme, controversial, and prevalent climate phenomena of the day runs the risk of introducing problems in this program's efficacy. Were it not clear from 2.6, this algorithm is incapable of forecasting precipitation, or any predictand, outside the range of its potential analogues. Furthermore, it is dependant on a consistent correlation between precipitation and temperature, something global warming has been shown to harm (Butler 2018). A more in-depth look into these changes is necessary to determine the severity of climate change's impacts on this algorithm's forecasting efficacy.

And finally, a look into implementing downscaling weather data is necessary. As mentioned in the introduction, the most certain solution to the absence of data on the station level is to start collecting data now and wait for enough to be available later. However, before that, there is the potential option of taking data recorded or modeled at higher-scale resolution (in this instance, meaning simply to be applied to a larger geographical region than what precipitation could realistically be applied to) and performing statistical downscaling to bring those models down to the resolution necessary to perform those forecasts (Raje and Mujumdar 2011).

APPENDIX A

APPENDIX

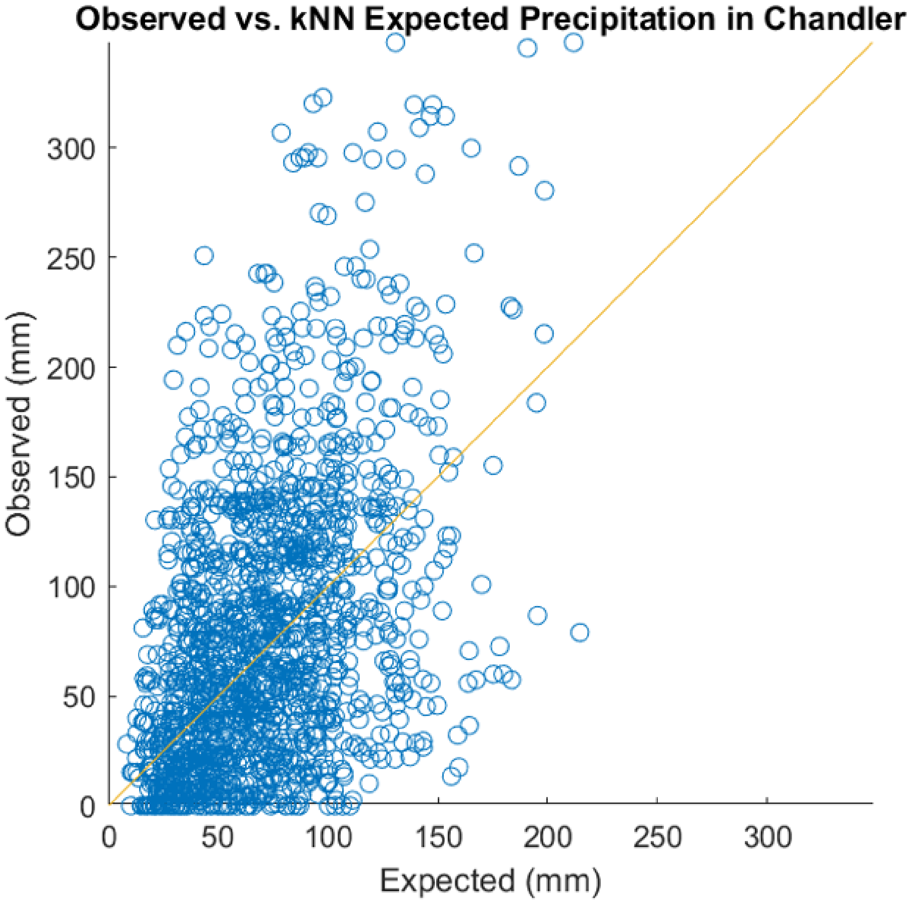


Figure A.1: Older Chandler Forecasts

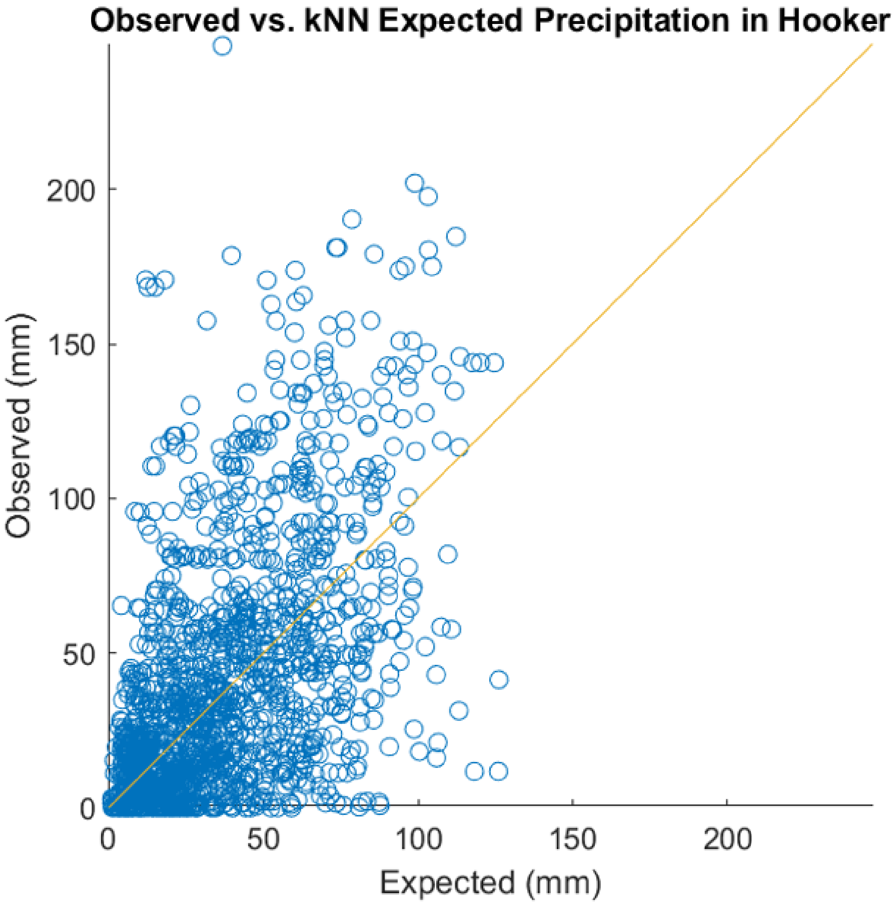


Figure A.2: Older Hooker Forecasts

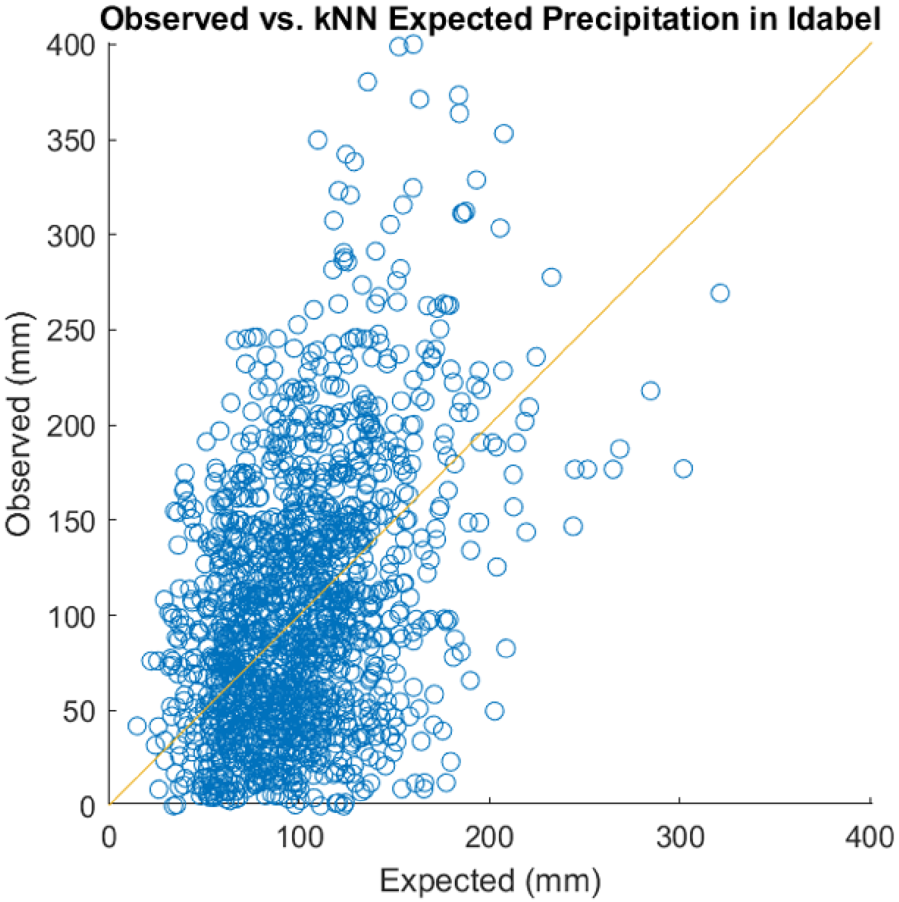


Figure A.3: Older Idabel Forecasts

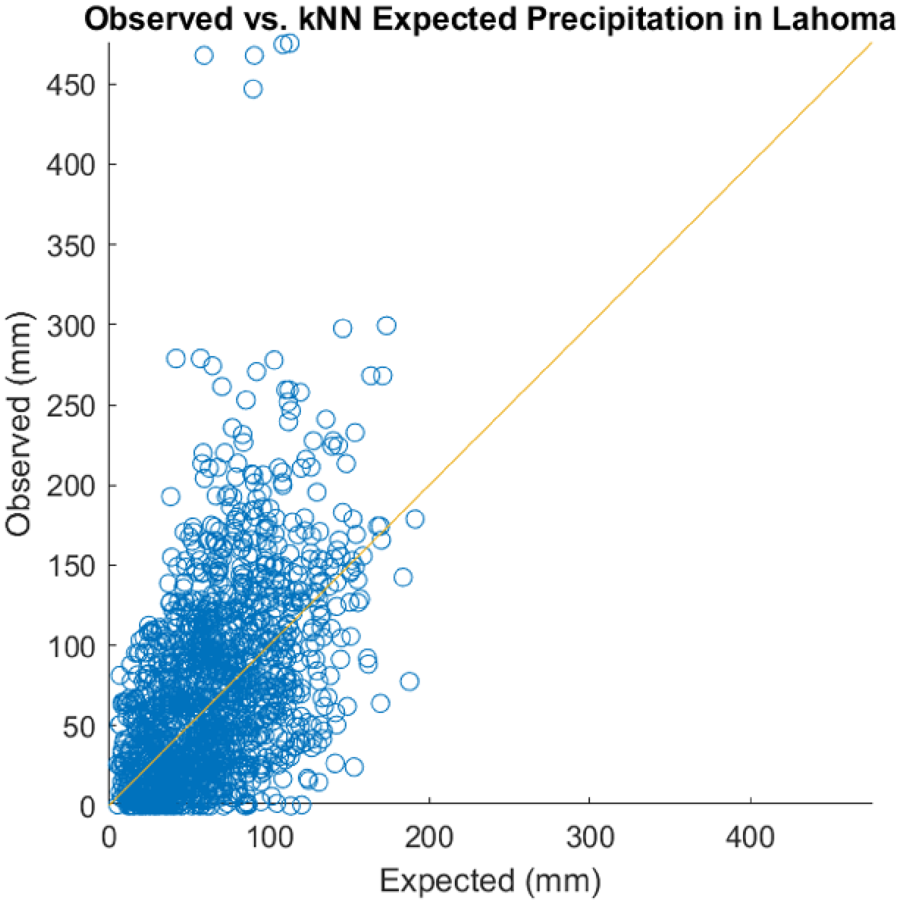


Figure A.4: Older Lahoma Forecasts

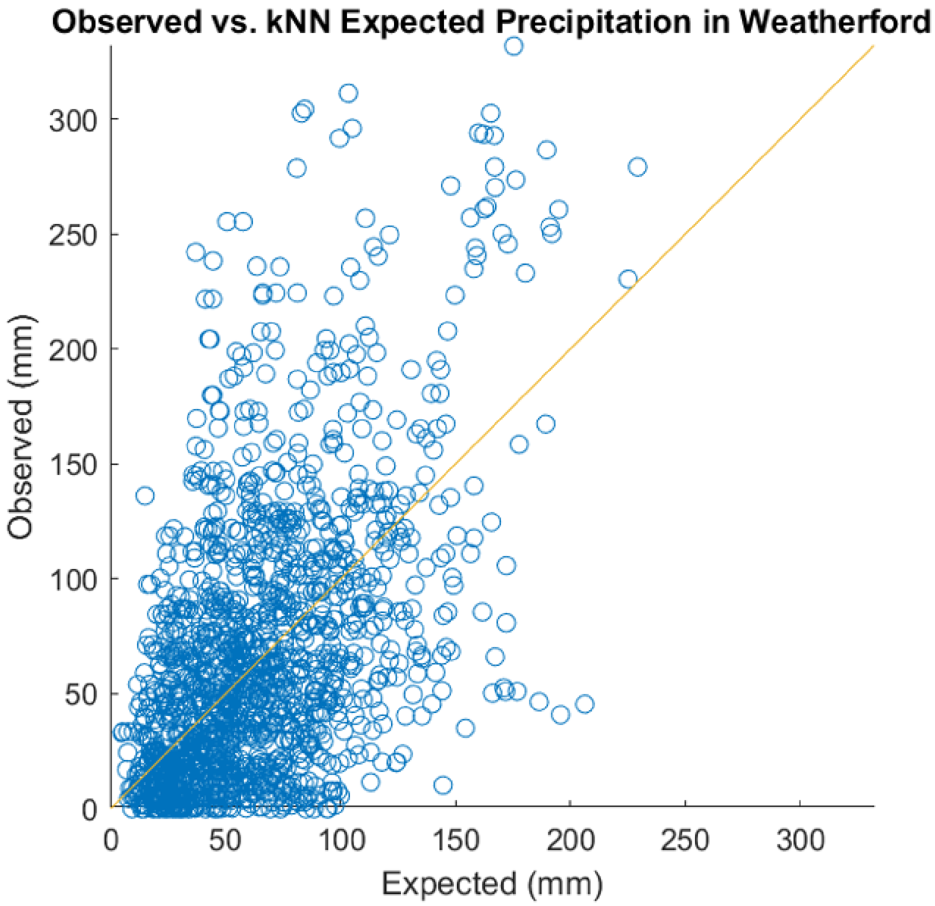
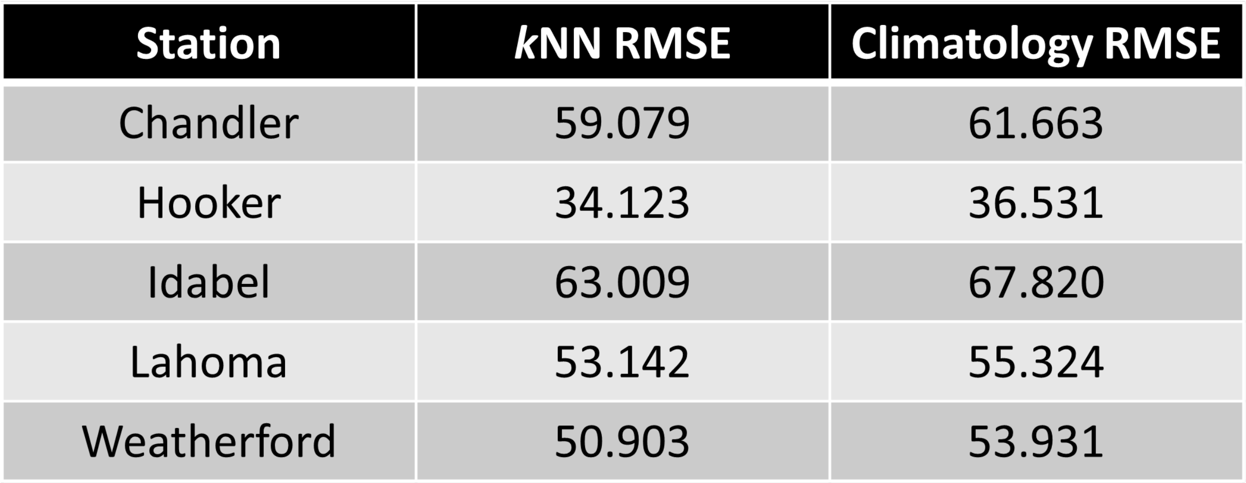


Figure A.5: Older Weatherford Forecasts



Station	*k*NN RMSE	Climatology RMSE
Chandler	59.079	61.663
Hooker	34.123	36.531
Idabel	63.009	67.820
Lahoma	53.142	55.324
Weatherford	50.903	53.931

Figure A.6: Older Results

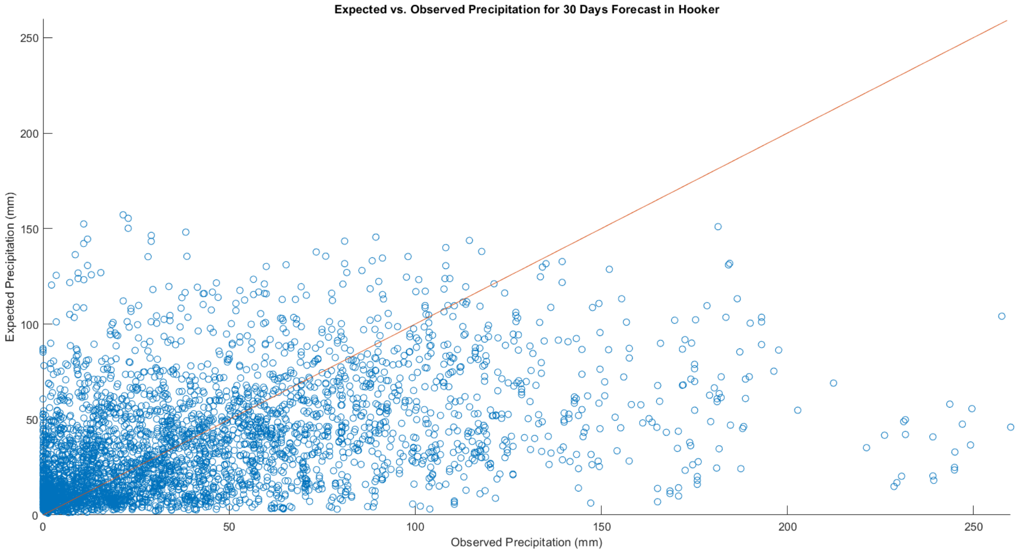


Figure A.7: Expected Precipitation vs. Observed Precipitation in Hooker Oklahoma using older model.

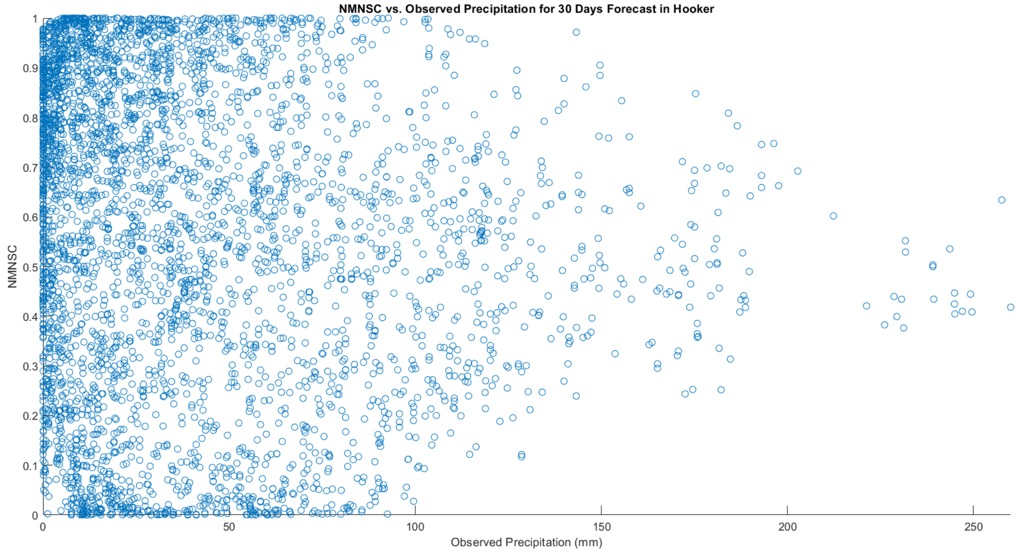


Figure A.8: Individual NMNSC vs. Observed Precipitation in Hooker Oklahoma using older model.

BIBLIOGRAPHY

Bannayan, M., and G. Hoogenboom, 2007: Predicting realizations of daily weather data for climate forecasts using the non-parametric nearest-neighbour re-sampling technique. *International Journal of Climatology*, **28**, 1357–1368, https://doi.org/10.1002/joc.1637.

Bannayan, M., and G. Hoogenboom, 2008: Weather analogue: A tool for real-time prediction of daily weather data realizations based on a modified k-nearest neighbor approach. *Environmental Modelling & Software*, **23 (6)**, 703–713, https://doi.org/https://doi.org/10.1016/j.envsoft.2007.09.011.

Bruno Soares, M., and S. Dessai, 2016: Barriers and enablers to the use of seasonal climate forecasts amongst organisations in europe. *Climatic Change*, **137**, 89–103.

Butler, C., 2018: Climate change, health and existential risks to civilization: A comprehensive review (1989–2013). *International Journal of Environmental Research and Public Health*, **15**, 2266, https://doi.org/10.3390/ijerph15102266.

Carberry, P., G. Hammer, H. Meinke, and M. Bange, 2000: The potential value of seasonal climate forecasting in managing cropping systems. *Applications of Seasonal Climate Forecasting in Agricultural and Natural Ecosystems*, 167–181, https://doi.org/10.1007/978-94-015-9351-$9_1$2.

Chakraborty, A., 2010: The skill of ecmwf medium-range forecasts during the year of tropical convection 2008. *Monthly Weather Review*, **138 (10)**, 3787 – 3805, https://doi.org/10.1175/2010MWR3217.1.

Choubin, B., G. Zehtabian, A. Azareh, E. Rafiei Sardooi, F. Sajedi Hosseini, and O. Kisi, 2018: Precipitation forecasting using classification and regression trees (cart) model: a comparative study of different approaches. *Environmental Earth Sciences*, **77**, https://doi.org/10.1007/s12665-018-7498-z.

Cortes, C., and V. Vapnik, 1995: Support-vector networks. *Mach. Learn.*, **20 (3)**, 273–297, https://doi.org/10.1023/A:1022627411411.

Cover, T., 1968: Estimation by the nearest neighbor rule. *IEEE Transactions on Information Theory*, **14 (1)**, 50–55, https://doi.org/10.1109/TIT.1968.1054098.

Cover, T., and P. Hart, 1967: Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, **13 (1)**, 21–27, https://doi.org/10.1109/TIT.1967.1053964.

Drucker, H., C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, 1996: Support vector regression machines. *Proceedings of the 9th International Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, USA, 155–161, NIPS'96.

Dudani, S. A., 1976: The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-6 (4)**, 325–327, https://doi.org/10.1109/TSMC.1976.5408784.

El Mrabet, M. A., K. El Makkaoui, and A. Faize, 2021: Supervised machine learning: A survey. *2021 4th International Conference on Advanced Communication Technologies and Networking (CommNet)*, 1–10, https://doi.org/10.1109/CommNet52204.2021.9641998.

Fix, E., and J. L. Hodges, 1952: Discriminatory analysis-nonparametric discrimination: Small sample performance. Tech. rep., California Univ Berkeley.

Fix, E., and J. L. Hodges, 1989: Discriminatory analysis. nonparametric discrimination: Consistency properties. *International Statistical Review / Revue Internationale de Statistique*, **57 (3)**, 238–247.

Garbrecht, J., and J. Schneider, 2007: Climate forecast and prediction product dissemination for agriculture in the united states. *Australian Journal of Agricultural Research - AUST J AGR RES*, **58**, 966–974, https://doi.org/10.1071/AR06191.

Hellman, M. E., 1970: The nearest neighbor classification rule with a reject option. *IEEE Transactions on Systems Science and Cybernetics*, **6 (3)**, 179–185, https://doi.org/10.1109/TSSC.1970.300339.

Ho, T. K., 1995: Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, Vol. 1, 278–282 vol.1, https://doi.org/10.1109/ICDAR.1995.598994.

Hochreiter, S., and J. Schmidhuber, 1997: Long short-term memory. *Neural computation*, **9**, 1735–80, https://doi.org/10.1162/neco.1997.9.8.1735.

Huang, M., R. Lin, S. Huang, and T. Xing, 2017: A novel approach for precipitation forecast via improved k-nearest neighbor algorithm. *Advanced Engineering Informatics*, **33**, 89–95, https://doi.org/https://doi.org/10.1016/j.aei.2017.05.003.

Jones, J., J. Hansen, F. Royce, and C. Messina, 2000: Potential benefits of climate forecasting to agriculture. *Agriculture, Ecosystems Environment*, **82 (1)**, 169–184, https://doi.org/https://doi.org/10.1016/S0167-8809(00)00225-5.

Klemm, T., and R. McPherson, 2018: Assessing decision timing and seasonal climate forecast needs of winter wheat producers in the south-central united states. *Journal of Applied Meteorology and Climatology*, **57**, 2129–2140, https://doi.org/10.1175/JAMC-D-17-0246.1.

Mann, H. B., and D. R. Whitney, 1947: On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, **18 (1)**, 50 – 60, https://doi.org/10.1214/aoms/1177730491.

Meinke, H., and R. Stone, 2005: *Seasonal and Inter-Annual Climate Forecasting: The New Tool for Increasing Preparedness to Climate Variability and Change In Agricultural Planning And Operations*, Vol. 70, 221–253. Springer Nature, https://doi.org/10.1007/1-4020-4166-7$_1$1.

Nicholls, J., 1996: Economic and social benefits of climatological information and services: A review of existing assessments. *World Climate Applications and Services Programme.*

Raje, D., and P. P. Mujumdar, 2011: A comparison of three methods for downscaling daily precipitation in the punjab region. *Hydrological Processes*, **25 (23)**, 3575–3589, https://doi.org/https://doi.org/10.1002/hyp.8083, https://onlinelibrary.wiley.com/doi/pdf/10.1002/hyp.8083.

Roberts, A., 2023: Calibration curve: What you need to know. URL https://arize.com/blog-course/what-is-calibration-reliability-curve/.

Schneider, J., and J. Wiener, 2009: Progress toward filling the weather and climate forecast need of agricultural and natural resource management. *Journal of Soil and Water Conservation*, **64**, 100A–106A, https://doi.org/10.2489/jswc.64.3.100A.

Silverman, B., and Jones, 1989: E. fix and j.l. hodges (1951): An important contribution to nonparametric discriminant analysis and density estimation: Commentary on fix and hodges (1951). *International Statistical Review*, **57**, 233.

Wolpert, D., and W. Macready, 1997: No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, **1 (1)**, 67–82, https://doi.org/10.1109/4235.585893.

Yates, D., S. Gangopadhyay, B. Rajagopalan, and K. Strzepek, 2003: A technique for generating regional climate scenarios using a nearest-neighbor algorithm. *Water Resources Research*, **39 (7)**, 1199, https://doi.org/https://doi.org/10.1029/2002WR001769, https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2002WR001769.

Zhang, X., 2004: Calibration, refinement, and application of the wepp model for simulating climatic impact on wheat production. *Transactions of the American Society of Agricultural Engineers*, **47**, https://doi.org/10.13031/2013.16580.