Conditional mixture models for precipitation data quality control

Tadesse ZeMicheal
Oregon State University
Corvallis, OR
zemichet@oregonstate.edu

Thomas G. Dietterich
Oregon State University
Corvallis, OR
tgd@cs.orst.edu

ABSTRACT

Rainfall is a very important weather variable, especially for agriculture. Unfortunately, rain gauges fail frequently. This paper describes a conditional mixture model for predicting the presence and amount of rain at a weather station based on measurements at nearby stations. The model is evaluated on simulated faults (blocked rain gauges) inserted into observations from the Oklahoma Mesonet. Using the negative log-likelihood as an anomaly score, we evaluate the area under the ROC and precision-recall curves for detecting these faults. The results show very good performance.

CCS CONCEPTS

• **Computing methodologies → Anomaly detection**; *Mixture modeling*.

ACM Reference Format:
Tadesse ZeMicheal and Thomas G. Dietterich. 2020. Conditional mixture models for precipitation data quality control. In *ACM SIGCAS Conference on Computing and Sustainable Societies (COMPASS) (COMPASS '20), June 15–17, 2020, Ecuador*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3378393.3402233

1 INTRODUCTION

Rainfall is a vital weather variable for humanity. Many countries in the world depend primarily on rainfed agriculture for their staple foods. In sub-Saharan Africa, more than 95% of the farmed land is rainfed; for Latin America it is almost 90%; and for South Asia it is approximately 60% [15, 20]. Achieving high data quality for rainfall data is challenging for three reasons. First, in many regions, rainfall is zero on most days, but on days when rain does occur, the amount has a heavy-tailed distribution. This makes it difficult to create a probabilistic model of predicted rainfall that can be applied to flag measurement errors. Second, rainfall is often a spatially local phenomenon. The correlation between rainfall at nearby weather stations can be very low, especially for rain showers. Third, rain gauges fail frequently because they collect precipitation and funnel it through a small orifice before measuring it (often with a tipping bucket mechanism). The orifice can be blocked by dust and leaves, and the moving parts of the tipping bucket can break or jam.

Our work is motivated by the weather station network TAHMO, the Trans-Africa Hydro-Meteorological Observatory. TAHMO has the objective of covering all of sub-Saharan Africa with low cost weather stations with an inter-station spacing of 30 km and a total of roughly 20,000 stations. This network will revolutionize the collection of weather and water data in Africa and will be the largest (by way of scale and number) uniform-station sensor weather network in the world [18, 22]. However, the most prevalent method for quality control of precipitation data relies on simple range check rules combined with human expert analysis. It is not practical to employ human experts to perform quality control on such large weather networks.

In this work, we introduce a conditional mixture model to detect failures in rain gauges due to blockage. We design a two-stage model that assigns a predicted probability to the precipitation measured at one weather station based on nearby stations. The negative log of that probability provides an anomaly score that signals unusual precipitation readings, which are then flagged for human inspection. In the first stage of the model, we predict whether it rained or not via logistic regression. This handles the large number of days with zero precipitation. In the second stage, we predict the amount of rain via a log-normal distribution. This models the heavy tail of the non-zero precipitation. Both steps use the amount of rain at k neighboring stations as covariates.

In the following sections, we first describe the TAHMO weather network in more detail and survey related work. We then introduce the problem definition and the method in detail. In the subsequent sections, we present experiments to evaluate the effectiveness of the method on data from the Oklahoma weather Mesonet.

2 THE TAHMO WEATHER NETWORK

TAHMO currently has approximately 600 automated weather stations deployed in 22 countries in sub-Saharan Africa. The stations are the ATMOS 41 model manufactured by METER, Incánd include sensors for precipitation, temperature, atmospheric pressure, relative humidity, solar radiation, and wind speed and direction (https://www.metergroup.com/environment/products/atmos-41-weather-station/). Measurements are recorded every five minutes. Each station has a data logger that uploads data multiple times per day via cellular data services.

The rain gauge is based on an electrical drip counter. Precipitation enters a collecting funnel on the top of the station. At the spout of the funnel, the precipitation is formed into standard-sized droplets, which are then counted electrically. At the time of writing, the majority of the 180 open trouble tickets are for precipitation problems. The rain gauge has two primary failure modes: blockage and short circuit. Blockage is caused by dust, leaves, bird droppings,

COMPASS '20, June 15–17, 2020, , Ecuador

ACM ISBN 978-1-4503-7129-2/20/06…$15.00
https://doi.org/10.1145/3378393.3402233

and other materials blocking or constricting the funnel spout. Short circuits arise when something, typically an insect, lodges between the two electrodes and causes false drip counts. Short circuits are particularly easy to detect, because the measured rainfall rapidly exceeds normal levels. Blockages are more challenging, because they are difficult to distinguish from the many days that have no measurable rainfall.

Most TAHMO stations are located at schools, and one of the school teachers is recruited to be the station host and regularly clean the station. In return, the teacher (and the school) get access to the data for their school and for the TAHMO network. TAHMO provides lesson plans and educational materials to incorporate weather data into elementary and secondary school curricula in science and mathematics.

Data from TAHMO is analyzed by a set of quality control rules developed for the Oklahoma Mesonet by Fiebrich, et al. [6]. The precipitation rule is able to detect high readings caused by short circuits, but it is not able to detect blockages, because a reading of zero is very common. Data values are flagged as "ok", "inconsistent", "suspect", and "error" depending on the type of rule that is violated and the degree of the violation. A report summarizing the number of problem flags per station is consulted daily by the network manager. This person is responsible for creating trouble tickets and assigning them to field engineers. Depending on the nature of the fault, the field engineer may contact the host and ask them to clean the station, or they may travel to the station to replace the batteries, failed sensors, or the entire station as necessary.

Once errors are removed from the TAHMO data, it is provided to the governments of each of the countries where TAHMO operates. It is also incorporated into the CHIRPS global rainfall model that combines satellite and ground station data to provide quasi-global climatology from 1981 to the present (https://www.chc.ucsb.edu/data/chirps). CHIRPS is in turn a key component of the Famine Early Warning System Network (https://fews.net). TAHMO is seeking additional partners interested in applying the data to create or improve businesses in crop insurance, logistics, and transportation.

3 RELATED WORK

There have been many studies in weather data quality control (QC) [7, 9, 11, 16, 19]. Most approaches follow standard practice for industrial quality control. Measured values from each sensor are checked to see if they are within legal ranges or within ranges based on historical climate. Some QC rules check values to detect large step changes. Other QC rules verify that the variance in measured values over a time period exceeds a specified non-zero threshold in order to detect flat line readings caused by frozen or broken sensors [3, 5].

These approaches to quality control do not work well for precipitation [21]. For example, we expect long periods of zero precipitation as well as large step changes in precipitation, so step tests and variance tests fail. Recently, many approaches that use multiple stations for QC procedures have been proven effective [10, 12]. The spatial regression test (SRT) uses nearby stations data to predict observations at the target station [7, 11]. Gutman and Quayle [8] employ inverse distance weighting to predict the observation at the target station was the weighted average of observations at nearby stations. Eischeid, et al. [4] employ multiple regression to predict the value at the target station, and Hubbard, et al., [10] compute interpolation weights in proportion to the standard error estimate of the difference between the observation at the target station and the measurements taken at nearby stations within a certain radius.

You, et al. [21] partition the observations at the target station into 10 bins based on equal quantiles of the precipitation observed at the nearest neighboring station. Then a gamma distribution is fit to each bin based on 30 years of daily precipitation data. During QC, the relevant gamma distribution is selected depending on which bin the neighboring station's observed precipitation falls into. A flag is raised if the observed precipitation at the target station is in the tails of the selected gamma distribution.

Our work is most similar to the multiple regression approach of Scheid, et al. [4]. However, we employ a model that combines a binary prediction for the presence of precipitation with a log-normal regression to estimate the quantity of precipitation.

4 PROBLEM SETUP

Let $s_1, s_2, \ldots s_n$ denote a network of weather stations. Let $R(s,t)$ denote the total rainfall measured at station s during the 24-hour period starting at time t. Given daily total precipitation data $R(s,t)$ for all n weather stations, our goal is to assign a score to each station s on each day t such that small values indicate the station's rain gauge was working correctly and large values indicate that the rain gauge is likely to be blocked or broken.

When assigning a score to station s, we will use the observations from nearby stations. Specifically, we will compute a set of k neighboring stations, denoted $\eta(s)$, that are most useful for predicting the observations at station s. Let $r_{\eta(s)}(t)$ denote the vector of rainfall observations for the 24-hour period beginning at time t. When the station s is clear, we will drop the (s) and write this as $r_\eta(t)$. For example, figure 1 shows readings for a target station (named FITT) and five nearby stations (SULP, TISH, ADAX, VANO, and CENT).

5 METHODS

We now define a series of progressively more sophisticated models for scoring the precipitation at station s as a function of its neighbors η. Each historical data point is a pair $(r_{\eta(s)}(t), R(s,t)))$. To simplify the notation, we will adopt the standard machine learning notation (x_i, y_i), where i indexes the data points, $x_i = r_{\eta(S)}(t)$ gives the observations at the neighboring stations, and $y_i = R(s,t)$ specifies the desired target value to be predicted.

5.1 Logistic regression model

Our first model simply predicts the probability $p_1(s,t)$ that it is raining at station s at time t by fitting a logistic regression:

$$p(y|x;\alpha) = \frac{1}{1 + e^{-(\alpha_0 + \alpha_1^\top x)}},$$

where $y = 0$ indicates no rain and $y = 1$ indicates rain. The parameters of the model are denoted by the vector $\alpha = (\alpha_0, \alpha_1)$, where α_0 is a scalar and α_1 is a k-dimensional vector with one value for each neighboring station. The notation $\alpha_1^\top x$ denotes the dot product between the vector α_1 and the vector of precipitation observations

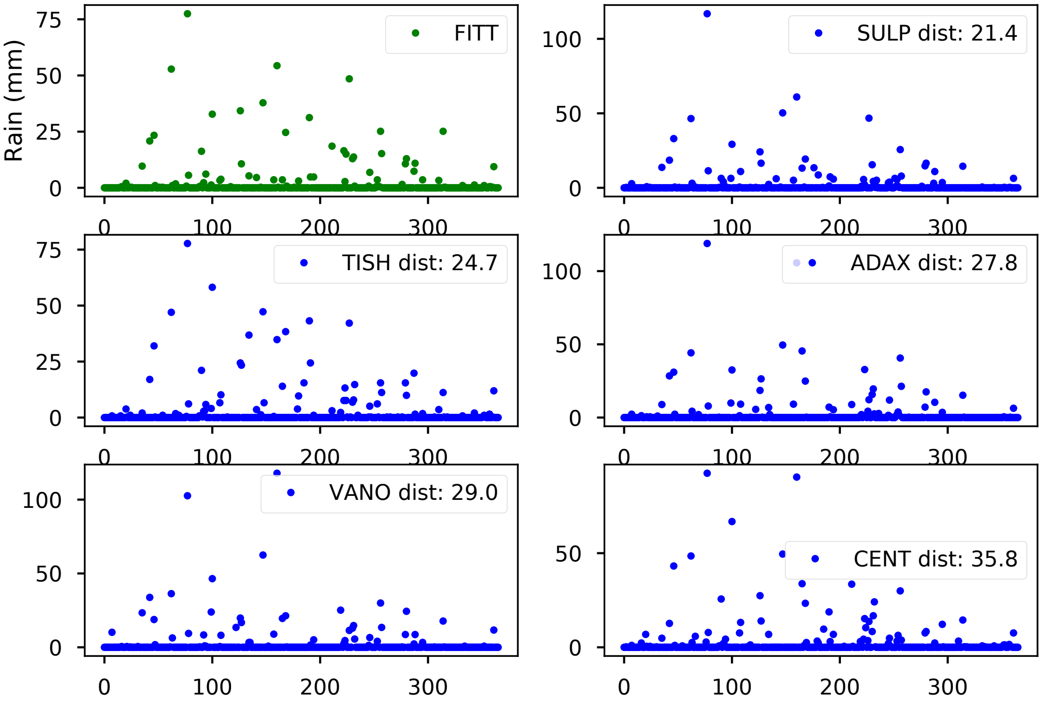


Figure 1: Daily precipitation of target station (FITT), and its most similar stations with distance in KM separation from the Oklahoma Mesonet network

$x = r_\eta$. The model is fit to observations of the form $(r_{\eta(s)(t)}, y)$, where $y = 1$ if $R(s,t) > 0$ and 0 otherwise.

Unfortunately, this model performs poorly because the large number of zero-precipitation days overwhelms the smaller number of rainy days, which causes the model to predict zero rain on most days and to assign it a high probability. The performance is so poor that we do not include this model in our experiments. However, we will use it as a component in the mixture models described below.

5.2 Linear regression model

Our second model is simple linear regression, similar to You, et al. [21]. We fit a model of the form

$$\hat{y} = \beta_0 + \beta_1^\top x,$$

where $\hat{y}$ is the predicted value of $y = R(s,t)$ given the observed precipitation $x = r_{\eta(s)}(t)$ of the neighboring stations. We fit this model with the objective of minimizing the squared error between $\hat{y}$ and the observed value $y = R(s,t)$.

The score, $LRR(s,t)$ for $R(s,t)$ is defined as the squared error

$$LRR(s,t) = (\hat{y} - R(s,t))^2.$$

The linear regression model suffers from the same drawback as logistic regression: It is unable to handle the large number of zero-rain days. However, because it is the most fundamental method for predictive modeling, We will include it as our baseline method.

5.3 Random forest regression model

Random forest regression fits a set of randomized regression trees to the historical precipitation data. The primary advantage of random forests is that they can fit complex non-linear relationships unlike the two previous models which both assumed a linear relationship between the neighboring stations and the target station. In particular, random forests can easily deal with the large number of zero-rain days. We fit a forest of 100 trees. To score the observed rain $R(s,t)$ at target station s, we pass the observed precipitation at the neighbors x through the random forest to obtain a predicted precipitation $\hat{y}$. We denote the resulting score as $RFR(s,t)$ and compute it as the squared difference between the predicted and observed precipitation:

$$RFR(s,t) = (\hat{y} - R(s,t))^2.$$

5.4 Random forest conditional density (RFCD)

The LRR and RFR scores assume that squared error is a good measure. An alternative is to fit a full probability density model $p(r|r_\eta)$

and then define the score as $-\log p(R(s,t)|r_{\eta(s)}(t))$. This will assign high scores to observed precipitation $R(s,t)$ that has low probability according to the fitted model.

For this purpose, we employ the Random Forest Conditional Density (RFCD) method of Pospisil and Lee [14]. This method constructs a random forest using a slight modification of the random forest method. Then to compute the conditional density $p(y'|x')$, it sends the observations x' through each tree of the random forest. When the observations reach the leaf ℓ of tree τ, the RFCD method collects up all of the training data points $F_\ell^\tau(x') = \{(x,y)\}$ that were assigned to that leaf during the tree-building process. It takes the multiset union $Z(x') = \bigcup_{\tau=1}^{M} F_\ell^\tau(x')$, where $M = 100$ is the number of trees in the forest, and the multiset union keeps all duplicates. We will denote the elements of Z by $z_1, \ldots, z_{|Z|}$, where each z_i has the form $z_i = (x_i, y_i)$.

RFCD then computes the conditional density via a kernel density estimate. Let $K_h(u)$ be a kernel function with width parameter h. The conditional density is defined as

$$p(y'|x') = \frac{1}{|Z(x')|} \sum_{i=1}^{|Z(x')|} K_h(y_i - y).$$

Plugging in $x' = r_{\eta(s)}(t)$, we obtain the score

$$-\log p(R(s,t)|r_{\eta(s)}(t)).$$

5.5 Conditional Mixture Model

All of the models discussed so far attempt to fit a single model structure to all of the data. An alternative is to model the rainy and non-rainy days with separate components. The result is a mixture model—a weighted combination of two models.

Similar models have been applied previously to both count data and continuous data [13]. A model similar to ours was developed by Duan [1] to assess health care policies, but it is not a predictive model and cannot be applied to score the probability of an observation.

5.5.1 Single station unconditional mixture model. Let us start by defining a model for the precipitation at the target station considering no neighbor stations. We will model the probability $P(R(s,t) = r)$ as

$$P(R(s,t) = r) = \begin{cases} 1 - p_1, & r = 0 \\ p_1 \cdot \mathcal{N}\big(\log(r); \mu, \sigma^2\big) & r > 0. \end{cases} \tag{1}$$

In this model, p_1 is the probability that rain will occur at station s and time t. Hence, $1 - p_1$ is the probability of no rain. The quantity of rain is modeled as a log-normal distribution $\mathcal{N}$ with mean μ and variance σ^2.

This is a mixture model with two components. In general, a two-component mixture model has the form $P(x) = (1-p_1)P_0(x) + p_1 P_1(x)$, where $1 - p_1$ and p_1 are called the mixing proportions, and P_0 and P_1 are called the mixture components. In our case, the first component P_0 is an "impulse" at 0. We can write this as $\delta_0(r)$; it is also known as the Dirac delta function. An impulse is a probability density at a single point that integrates to 1. The second component is the lognormal distribution. With this notation, we can also write (1) as

$$P\big(R(s,t) = r\big) = (1-p_1)\delta_0(r) + p_1 \cdot \mathcal{N}(\log(r); \mu, \sigma^2)$$

It is easy to fit this model. Let $r_1, \ldots, r_N$ be a set of rainfall observations for station s. Let N_1 be the number of observations that are nonzero. Then

$$\hat{p}_1 = \frac{N_1}{N}$$

Let $r^1, \ldots, r^N$ be the non-zero observations. Then we can estimate the mean and variance of the lognormal distribution as

$$\hat{\mu} = \frac{1}{N_1} \sum_{i=1}^{N_1} \log r^i$$

and

$$\hat{\sigma}^2 = \frac{1}{N_1} \sum_{i=1}^{N_1} (\log r^i - \hat{\mu})^2.$$

5.5.2 A conditional mixture model. Now let us consider the case where we wish to predict the probability of rain at station s using observations from neighboring stations $\eta(s)$: $P\big(R(s,t) = r|r_\eta(t)\big)$. Generalizing from our single-station mixture model, we can capture this as a conditional mixture model:

$$P(R(s,t) = r|r_\eta(t)) = \\ \begin{cases} (1 - p_1(r_\eta(t); \alpha)) & r = 0 \\ p_1(r_\eta(t); \alpha) \cdot \mathcal{N}\big(\log(r); \beta_0 + \beta_1^\top \log(r_\eta(t) + \vec{\epsilon}), \sigma^2\big) & r > 0. \end{cases} \tag{2}$$

The first component is a Dirac delta function at zero, and the second component is a lognormal regression model, with covariates $log(r_\eta(t) + \vec{\epsilon})$ and parameters $\beta = (\beta_0, \beta_1)$ and σ^2. The purpose of the $\vec{\epsilon}$ is to avoid taking the logarithm of zero when there is no rain. We set $\epsilon = 0.1$ and $\vec{\epsilon} = (0.1, \ldots, 0.1)$. The mixing proportion is computed by the logistic regression model $p_1(r_\eta; \alpha)$ as defined in Section 5.1.

To estimate β, we could restrict our attention to only the cases where $R(s,t) > 0$ and predict $\log(R(s,t))$ as a function of the log-transformed covariates $\log(R(s',t))$ for $s' \in \eta(s)$. However, this assumes that our logistic regression model makes perfect 0-1 predictions, whereas because of the properties of the logistic function, it always predicts a probability in $(0,1)$. Consequently, we chose to fit the lognormal regression to *all* of the observations.

To fit the mixture model, we first fit the logistic regression model to obtain α. Then we plug the values of $\hat{p}_1(s,t) = p(y = 1|x; \alpha)$ into the following likelihood function and then find the values of β to maximize the likelihood over the historical data:

$$l(\beta) = \sum_t \hat{p}_1(s,t) \Big[\log(R(s,t) + \epsilon) \\ - \Big(\beta_0 + \sum_{s' \in \eta(s)} \beta_{s'} \log(R(s',t) + \epsilon)\Big)\Big]^2.$$

This is different from the estimation done in [2], where the second model component is fit only to the days with non-zero precipitation and there is no need for ϵ and $\vec{\epsilon}$.

Once we have fit the model, we compute the residuals on the log scale.

$$\rho(R(s,t)) = \log(R(s,t) + \epsilon) - \Big(\beta_0 + \sum_{s' \in \eta(s)} \beta_{s'} \log(R(s',t) + \epsilon)\Big) \tag{3}$$

Finally, we fit the variance parameterσ^2 of the lognormal distribution by computing the variance of these residuals.

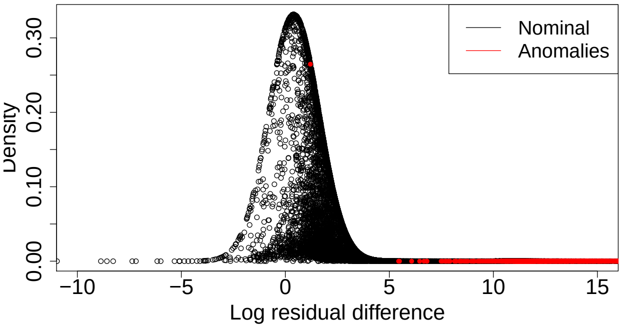


Figure 2: The distribution of of residuals showing nominal (black) and anomalous (red) sensor readings. The vertical axis is the computed density $P(\rho(R(s,t))|r_\eta(t))$.

Figure 2 shows the distribution of the residuals. The residuals corresponding to sensor failures mostly fall into the right tail, where they are easily detected by the proposed model.

To assign a score to the observed precipitation $R(s,t)$, we compute its p-value according to the mixture distribution. We do this by computing the cumulative distribution function (CDF) $F(\rho)$ of the conditional mixture model:

$$F(\rho) = (1 - p_1) + p_1\Phi(\rho; 0, \sigma^2), \qquad (4)$$

where $\rho = \rho(R(s,t)|r_\eta(t))$ is the residual computed according to equation (3). The mean in Φ is zero, because we are modeling the CDF of the residuals rather than of the precipitation. Note that all of the parameters depend on the neighboring stations $\eta(s)$. We write the score of $y = R(s,t)$ as

$$MNORM.CDF(y) = -\log[\min\{F(\rho(y)), 1 - F(\rho(y))\}], \qquad (5)$$

where $\rho(y)$ is the residual computed by equation 3.

5.5.3 *An Extended Mixture Model.*

The above model works fairly well, but the impulse at zero prevents the model from computing useful p-values when there is heavy rainfall at the neighboring stations and a blocked rain gauge at the target station. To address this, we extend the mixture model to incorporate both mixture components when $r = 0$. Let

$$f(r|r_\eta(t)) = \mathcal{N}\big(\log(r); \beta_0 + \beta_1^\top \log(r_\eta(t) + \vec{\epsilon}, \sigma^2)\big)$$

be the lognormal probability density from Equation (2). With this shorthand, we can write the modified model as Equation (6):

$$P(R(s,t) = r|r_\eta(t)) = \begin{cases} \min\{1 - p_1(r_\eta(t);\alpha)),\ p_1(r_\eta(t);\alpha)) \cdot f(r|r_\eta(t))\} & r = 0 \\ p_1(r_\eta(t);\alpha) \cdot f(r|r_\eta(t)) & r > 0. \end{cases} \qquad (6)$$

The change is in the first term, where we take the minimum of $(1 - p_1)$ and the density predicted by the log linear regression. This is no longer a proper density when $r = 0$. However, it gives better results when the lognormal model predicts a large value but the observed rainfall is zero.

We compare two methods for using this model to assign a score to $R(s,t)$. The first method computes the negative log of the fitted probability density:

$$MNORM.NLL(r) = -\log P(R(s,t) = r|r_\eta(t)).$$

This assumes that the residuals have a Gaussian distribution with mean zero and variance σ^2. The second method fits a non-parametric kernel density estimator (KDE) to the residuals using a Gaussian kernel and automatic kernel width selection via grid search [17]. Let $KDE(\rho)$ denote the kernel density estimator and re-define f to be

$$f(r|r_\eta(t)) = KDE(\rho(r)).$$

Then the score is computed from Equation (6) using this version of f:

$$MNORM.KDE(r) = -\log P(R(s,t) = r|r_\eta(t)).$$

6 EXPERIMENTAL EVALUATION

We designed an experiment to address the following research questions.

RQ1: What is the best method for selecting the set $\eta(s)$ of neighboring stations?
RQ2: Which model gives the best accuracy?
RQ3: What is the best way to model the residual?

6.1 Experiment Design

We obtained two years of weather observation data from Oklahoma (OK) Mesonet [6], which operates 120 stations distributed across the state of Oklahoma. We extracted precipitation data and aggregated it to a daily time step. All compared models were fit to data from 2008 and then tested on data from 2009. The data contained no actual occurrences of blocked rain gauges, but it did contain several instances of gauge failure coded as large negative values by the data logger. We discarded these cases, as they are easily detected by simple rules.

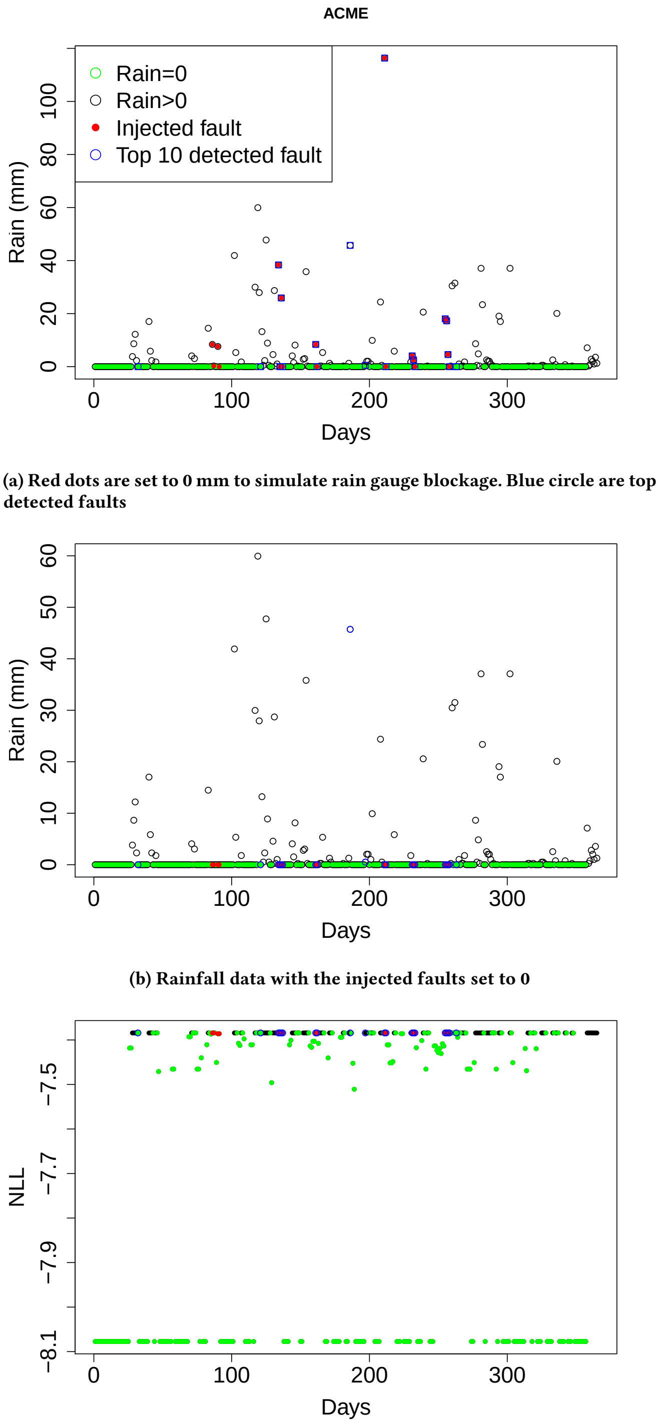
To assess the accuracy of the models for each target station s, we insert simulated blocked sensor faults into the 2009 Oklahoma data as follows. The precipitation time series for s was segmented into rainfall episodes of more than one day in duration. Five percent of these episodes were then selected uniformly at random, and the observed precipitation values were replaced with zero readings. Figure 3 shows an example of the faults injected for the ACME station.

6.1.1 *Metrics.*

We use the following metrics to evaluate the performance of the models.

Precision @ 80% recall (PREC@80): Precision is the fraction of all detected failures that are real sensor failures, and *Recall* is the fraction of all real sensor failures detected. Ideally, we want to achieve 100% precision (everything we detect is correct) and 100% recall (we detect everything that is broken). The *Precision@80%* recall metric measures the precision achieved by a model that detects 80% of the true (simulated) faults. We measure this by setting a decision threshold θ such that all observations with scores greater than θ are declared to be faults. We choose θ to achieve 80% recall and then measure the precision.

Average precision (AP): *Average precision* is defined as the average precision at a set of decision thresholds. This captures how a models detect failures without considering any specific decision



(a) Red dots are set to 0 mm to simulate rain gauge blockage. Blue circle are top detected faults

(b) Rainfall data with the injected faults set to 0

(c) Negative likelihood of the points after anomaly scoring. Blue points incircled are succesffully detected of top 10 faults

Figure 3: Example of rain gauge blockage fault injection and detection for the ACME station (2008 data).

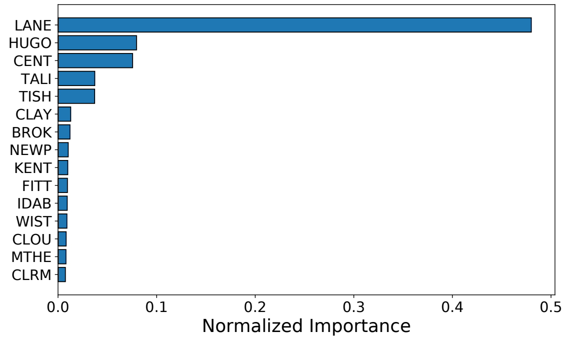


Figure 4: Feature importance ranked using random forest fitted across all stations for target station DURA

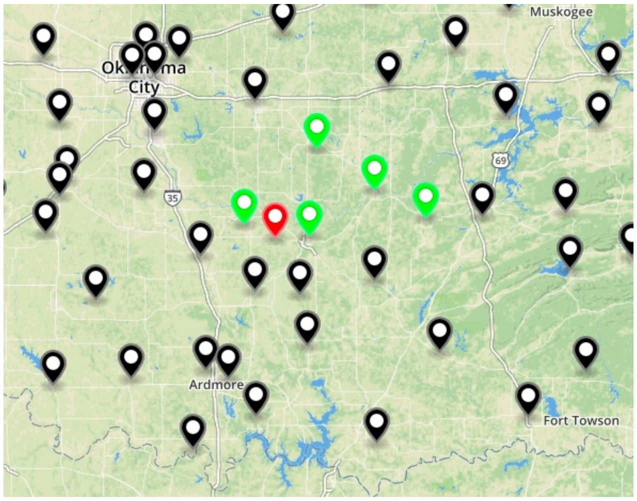


Figure 5: A target station marked red (named VANO) and the green stations are selected as the most predictive stations to it.

threshold. It is equivalent to measuring the area under the precision-recall curve, which plots the precision as a function of recall (by varying θ).

Area under ROC curve (AUC): The ROC curve plots the tradeoff between the false alarm rate and the true alarm rate. The area under this curve (AUC) can be interpreted as the probability that the model will assign a higher score to a randomly-selected faulty measurement than to a randomly-selected good reading. A good model will have an AUC close to 1.0, whereas, a model that assigns scores at random will have an AUC of 0.5.

6.2 Selecting nearby stations

To address RQ1 studied two methods for selecting the neighboring stations $\eta(s)$. The first is to use the k geographically-nearest stations (as measured by Haversine distance). The second is to fit a random forest to predict whether $P(R(s,t) > 0)$ at station s based on the

Table 1: Mean accuracy ±1 standard error on 2009 Oklahoma Mesonet with inserted faults detected by the MNORM.NLL model.

	$\eta(s)$		
Aggregation	Method	AUC	AP
Day	RF	**0.85** ± 0.10	**0.55** ± 0.28
	Distance	0.81± 0.11	0.47±0.27
Episode	RF	**0.88 ± 0.11**	**0.65±0.31**
	Distance	0.86± 0.11	0.58± 0.30

observed precipitation at *all* of the weather stations in the region (excluding s) and then use the random forest variable importance score to select the k most informative stations within 100 km.

This is similar to Eischeid, et al., [4], who compute the pairwise correlation between monthly precipitation time series of the target station and nearby stations and select the four neighboring stations with the highest correlation coefficients. Similarly, Hubbard, et al., [10] select neighboring stations based on the standard error of a simple linear regression that predicts the value at the target station from the value of the candidate neighbor station. The 5 neighbors with smallest standard error are selected and employed in their weighted regression model.

Figure 4 shows a typical example of for the random forest importance scores, and Figure 5 shows the stations selected for the VANO target station. Note that in this case, the most predictive stations are all located on the north side of VANO.

We evaluated these two neighbor selection algorithms by selecting neighbors, fitting the model of Equation (6), and measuring the resulting AUC and AP for detecting the inserted rain gauge blockages using the MNORM.NLL score. We measure this effectiveness on two time scales. First, we measure separately for each day that the gauge is blocked. Second, we compute the metrics for entire precipitation episodes. We consider that an episode has been detected if at least one day within the episode was scored as an anomaly. The rationale for scoring episodes is that in field application, if at least one day is detected, then the network manager will notice the problem and create a trouble ticket.

Table 1 shows the results. To answer RQ1, we see that the random forest method for choosing $\eta(s)$ is superior to the distance method according to both metrics and at both the daily and episode level ($p < 0.05$ with a paired differences test; not shown). The performance on episodes is better than the performance on individual days, which is unsurprising given that episodes are longer than single days, so they provide more opportunities to detect the problem (also significant at $p < 0.05$ with a paired differences test).

6.3 Model comparison

To address RQ2 and RQ3, we ran a set of experiments to detect blockages of rain gauges during rain events. We compare all of the models to the baseline models for failure detection. We evaluated 120 stations on individual day detection performance. We compute the metrics for each station separately, and finally we report the metrics averaged across the 120 stations along with a 95% confidence interval. Table 2 and Figure 6 show the results.

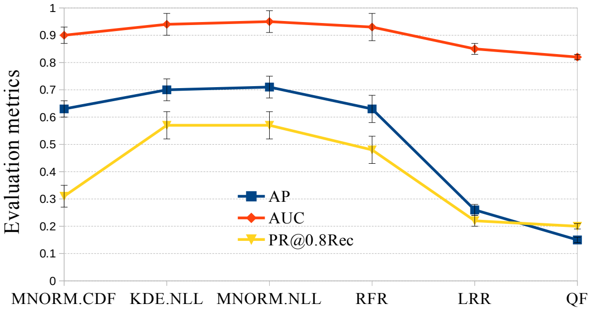


Figure 6: Average AUC, AP and PREC@80 recall across all weather station of OK in 2009

Table 2: Summary of model comparisons ± standard error on 2009 Oklahoma Mesonet with inserted faults.

Metric	AP	AUC	PREC@80 Recall
LRR	0.26±0.02	0.85±0.02	0.22±0.02
RFR	0.39±0.03	0.90±0.01	0.28±0.02
QF	0.15±0.01	0.82±0.01	0.20±0.01
MNORM.CDF	0.63±0.03	0.90±0.02	0.31±0.04
MNORM.NLL	**0.71±0.04**	**0.95±0.01**	**0.57±0.05**
KDE.NLL	0.70±0.04	0.94±0.01	**0.57±0.05**

For RQ2, the results show that all variants of the proposed models (MNORM.CDF, MNORM.NLL, KDE.NLL) have significantly better performance than the baseline models as measured by AP and PREC@80 and slightly better performance on AUC.

The AUC values above 0.90 show that the models work well in ranking the failures above the correctly measured values. However, further looking into the precision-recall tradeoff, we see significant differences in performance across the models. The baseline LRR and conditional quantile forest density (QF) perform significantly worse than all other models. This aligns with our intuition that the inflated zero value degrades the performance of both models. We expected the QF to achieve comparable results to the random forest regression (RFR), as both are less sensitive to the inflated zeros. However, on both AP and PREC@80 recall QF is worse than RFR. We suspect the reason could be related to the extra modeling of density under QF. The trees fit for QF are shallower because they need to have "large" leaves (i.e., that contain many data points) in order to give stable density estimates. This smoothness may lead to biases, especially near zero.

In contrast, the proposed conditional mixture models are consistently better in AP and PREC@80 than all other baseline models. The best model (MNORM.NLL) is able to achieve 57% precision while detecting 80% of the faulty days. This means, the network manager will need to deal with 43% false alarms. This is significantly better than the best baseline model (RFR), which generates around 72% false alarms to achieve the same level of recall.

Let us now turn to RQ3, which asks which of the three methods (CDF, NLL, or KDE) is the best method for computing a failure score

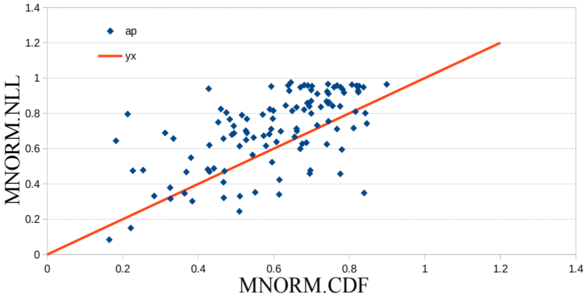


Figure 7: Average precision (AP) of 120 weather stations OK for 2009 precipitation of MNORM.CDF & MNORM.NLL

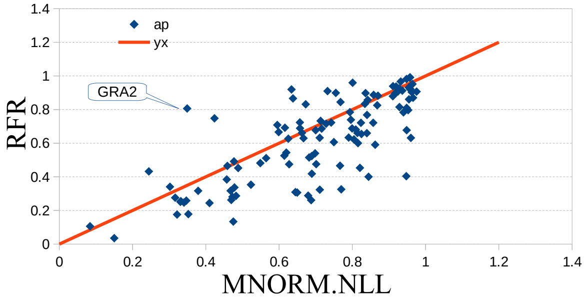


Figure 8: Average precision (AP) of 120 weather stations OK for 2009 precipitation of MNORM.NLL & RFR

from the fitted mixture model. The results clearly show that the CDF method performs the worst. This is somewhat surprising, because from a theoretical standpoint, the CDF method is the only way to combine the impulse δ_0 with the continuous lognormal density. The problem with the CDF method is that the cumulative distribution function jumps immediately up to $1 - p_1$ at zero precipitation, so it often assigns a high probability to zero values, even when the rain gauge is blocked.

The other two methods, MNORM.NLL and KDE.NLL, are statistically indistinguishable, with MNORM.NLL having a slight advantage.

To study how performance varies across stations, we show two scatter plots. Figure 7 plots the AP of MNORM.CDF on the horizontal axis and MNORM.NLL on the vertical axis. We observe that the majority of points lie above the diagonal line, which matches the overall result that MNORM.NLL is generally better. But we do observe that MNORM.CDF is better for several stations.

Figure 8 plots the AP of MNORM.NLL on the horizontal axis and the AP of RFR (random forest regression) on the vertical axis. RFR was the best of the baseline (non-mixture) models. Most of the points lie below the diagonal line, which shows that MNORM.NLL is the better method for most stations. However, again we observe that some stations lie substantially above the diagonal.

To understand this better, we looked at the point indicated as GRA2, which is the one furthest above the diagonal line. Figure 9 shows that this is a weather station near the border between Oklahoma and Texas. The green stations indicate the chosen neighbors, and they are all fairly far away from GRA2 and all lie on one side of it. Perhaps in such situations, the linearity assumptions of the lognormal regression model lead to problems, and the random forest method is able to do a better job. In this case, RFR achieves an AP near 0.8 whereas the AP of MNORM.NLL is below 0.5.

The overall lesson is that none of the methods we studied was the best on every station. Instead, the best scoring method may need to be chosen separately for each station.

7 CONCLUDING REMARKS

In this work, we have shown that a two-stage conditional mixture model can detect (simulated) blocked rain gauge events despite the challenges of precipitation quality control. We implemented three baseline methods (linear regression, random forest regression, and quantile forest regression) and three variations of a conditional mixture model (MNORM.CDF, MNORM.NLL, and KDE.NLL). All methods were assessed for their ability to detect injected faults in the 120 weather stations of the Oklahoma Mesonet. The statistical models were fit to (clean) data from 2008 and then evaluated on data from 2009 with simulated faults.

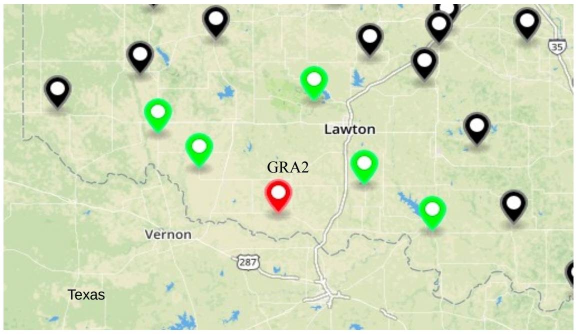


Figure 9: GRA2 location at the border of Oklahoma state

The conditional mixture models give much better Average Precision and Precision@80% Recall than the baseline methods, with the MNORM.NLL giving the best overall performance. However, detailed examination showed that the best model can vary from one station to another, and sometimes the random forest method gives the highest fault detection performance.

One potential direction for improvement is to fit the lognormal regression using a robust loss function such as the Huber loss. This might further improve the ability of the mixture models to handle the large number of zero values.

The methods in this paper are currently being deployed on the TAHMO weather network, and will be operational by August, 2020.

8 ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant Nos. 1514550 and 1521687 and a gift from

Huawei, Inc. We thank John Selker, the head of TAHMO, for his advice and guidance.

REFERENCES

[1] Naihua Duan, Willard G Manning, Carl N Morris, Joseph P Newhouse, Naihua Duan, Willard G Manning, Carl N Morris, and Joseph P Newhouse. 1983. A Comparison of Alternative Models for the Demand for Medical Care. *Journal of Business & Economic Statistics* 1, 2 (1983), 115–126.

[2] Naihua Duan, Joseph P Newhouse, Carl N Morris, and Willard G Manning. 1981. *A Comparison of Alternative Models for the Demand for Medical Care.* RAND Corporation.

[3] Thomas Einfalt and Silas Michaelides. 2008. Quality control of precipitation data. In *Precipitation: Advances in measurement, estimation and prediction.* Springer, 101–126.

[4] Jon K Eischeid, C Bruce Baker, Thomas R Karl, and Henry F Diaz. 1995. The quality control of long-term climatological data using objective data analysis. *Journal of applied meteorology* 34, 12 (1995), 2787–2795.

[5] Song Feng, Qi Hu, and Weihong Qian. 2004. Quality control of daily meteorological data in China, 1951–2000: a new dataset. *International Journal of Climatology* 24, 7 (2004), 853–870.

[6] Christopher A Fiebrich, Cynthia R Morgan, Alexandria G Mccombs, Peter K Hall, Jr, Renee A Mcpherson, Ynthia R Morgan, Alexandria G Mccombs, Peter K Hall, and Renee A Mcpherson. 2010. Quality assurance procedures for mesoscale meteorological data. *J. Atmos. Ocean. Technol.* 27, 10 (2010), 1565–1582.

[7] Lev S Gandin. 1988. Complex quality control of meteorological observations. *Monthly Weather Review* 116, 5 (1988), 1137–1156.

[8] Nathaniel B Guttman and Robert G Quayle. 1990. A review of cooperative temperature data validation. *Journal of Atmospheric and Oceanic Technology* 7, 2 (1990), 334–339.

[9] Vesa Hasu and Heikki Koivo. 2008. Automatic Rain and Wind Measurement Fault Identification in Mesoscale Weather Station Networks. In *2008 IEEE Instrumentation and Measurement Technology Conference.* IEEE, 489–494.

[10] KG Hubbard, S Goddard, WD Sorensen, N Wells, and TT Osugi. 2005. Performance of quality assurance procedures for an applied climate information system. *Journal of Atmospheric and Oceanic Technology* 22, 1 (2005), 105–112.

[11] Kenneth G Hubbard and MVK Siva Kumar. 2001. Automated weather stations for applications in agriculture and water resources management. (2001).

[12] Alba Llabrés-Brustenga, Anna Rius, Raúl Rodríguez-Solà, M Carmen Casas-Castillo, and Angel Redaño. 2019. Quality control process of the daily rainfall series available in Catalonia from 1855 to the present. *Theoretical and Applied Climatology* (2019), 1–15.

[13] Yongyi Min and Alan Agresti. 2002. Modeling Nonnegative Data with Clumping at Zero: A Survey Models for Semicontinuous Data. *Journal of the Iranian Statistical Society* 1, 1-2 (2002), 7–33.

[14] Taylor Pospisil and Ann B Lee. 2018. RFCDE: Random Forests for Conditional Density Estimation. (April 2018). arXiv:1804.05753 [stat.ML]

[15] Johan Rockström and Malin Falkenmark. 2015. Agriculture: increase water harvesting in Africa. *Nature News* 519, 7543 (2015), 283.

[16] Mark A Shafer, Christopher A Fiebrich, Derek S Arndt, Sherman E Fredrickson, and Timothy W Hughes. 2000. Quality assurance procedures in the Oklahoma Mesonetwork. *Journal of Atmospheric and Oceanic Technology* 17, 4 (2000), 474–494.

[17] Berwin A Turlach. 1993. Bandwidth selection in kernel density estimation: A review. In *CORE and Institut de Statistique.* Citeseer.

[18] Nick van de Giesen, Rolf Hut, and John Selker. 2014. The Trans-African Hydro-Meteorological Observatory (TAHMO). *WIREs Water* 1, 4 (July 2014), 341–348.

[19] Charles G Wade. 1987. A quality control program for surface mesometeorological data. *Journal of Atmospheric and Oceanic Technology* 4, 3 (1987), 435–453.

[20] Suhas Pralhad Wani, Johan Rockström, Theib Yousef Oweis, et al. 2009. *Rainfed agriculture: unlocking the potential.* Vol. 7. CABI.

[21] Jinsheng You, Kenneth G Hubbard, Saralees Nadarajah, and Kenneth E Kunkel. 2007. Performance of quality assurance procedures on daily precipitation. *Journal of Atmospheric and Oceanic Technology* 24, 5 (2007), 821–834.

[22] Tadesse Zemicheal and Thomas G. Dietterich. 2019. Anomaly Detection in the Presence of Missing Values for Weather Data Quality Control. In *Proceedings of the 2nd ACM SIGCAS Conference on Computing and Sustainable Societies* (Accra, Ghana) *(COMPASS '19)*. Association for Computing Machinery, New York, NY, USA, 65–73. https://doi.org/10.1145/3314344.3332490