Predictive flood mapping in cloud-obscured satellite imagery

Ian P. Davies

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Science

University of Washington

2020

Committee:

Phillip Levin

David Butman

Youngjun Choe

Program Authorized to Offer Degree:

School of Environmental and Forest Sciences

University of Washington

Abstract

Predictive flood mapping in cloud-obscured satellite imagery

Ian P. Davies

Chair of the Supervisory Committee:

Phillip Levin

School of Environmental and Forest Sciences

Maps of flood inundation derived from satellite imagery during and after a flood event are critical tools for disaster management. Their utility, however, is limited by optically thick cloud cover that obscures many spaceborne sensors. This study explores a data-driven method to predict flooding in cloud-obscured pixels. For an obscured image, models were trained on visible pixels using 30-meter flood conditioning features and then used to predict flooding on the cloud-covered pixels of that same image. Logistic regression, random forest, and neural networks were evaluated. To obtain prediction uncertainty estimates, a Bayesian neural network using Monte Carlo dropout was trained and compared to Logistic regression confidence intervals. Logistic regression and neural networks averaged 96% accuracy and 86% AUC, but poor recall of <35%. The Bayesian neural

network provided useful measures of uncertainty that tracked well with prediction errors. Finer resolution data and more input features may improve this method.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1. INTRODUCTION

In hydrological systems throughout the world, floods are natural and inevitable drivers of environmental change. Yearly flooding is responsible for recharging groundwater, increasing connectivity between aquatic habitats, transferring organic matter to terrestrial ecosystems, and is closely linked to the breeding and migration of many species (Associated Programme on Flood Management, 2006; Smardon et al., 1996; G. Zhang et al., 2017). Yet at the same time, floods are also the deadliest and most common environmental disaster in the world, each year displacing millions of people and causing billions in economic disruptions (Dottori et al., 2016; Jongman et al., 2012). Unfortunately, flood-vulnerable populations are expected to grow in the coming decades due to a confluence of trends. First, more extreme precipitation regimes in a changing climate are likely to increase the frequency of floods in certain parts of the world, particularly in the tropics but also coastal North America (Hirabayashi et al., 2013). These regions are projected to experience the greatest population growth in the coming decades, much of that concentrated around rivers and coasts (Neumann et al., 2015). To accommodate population growth, these regions will likely undergo significant land use changes, such as urbanization and the filling of wetlands, which increase runoff and remove the natural capacity of an area to absorb excess rainfall (Ireland et al., 2015; Loucks, 2019).

Flood events, like many environmental disasters, occur disproportionately in tropical countries with higher poverty rates and generally less infrastructure and institutional capacity for managing flood risk or events (Centre for Research on the Epidemiology of Disasters, 2015; Jha et al., 2012; Katsuhama & Grigg, 2010). But even within more prosperous countries, floods occur most disastrously in vulnerable communities without the resources necessary to mitigate, adapt, or

rebuild from them, such as during Hurricanes Katrina in 2005 and Sandy in 2012 (Brodie et al., 2006; Fussell, 2015; McGhee et al., 2020). These realities have led governments and international development organizations to prioritize integrated flood management to reduce the negative externalities of flooding while maintaining social and ecological benefits (WMO-UNESCO Joint-Task Team, 2007).

1.1 Flood Mapping

With global flood regimes shifting so quickly, there is a growing need to predict and monitor floods around the world before, during, and after an event. To predict floods, researchers and managers rely on flood hazard maps that depict the probability that a flood will occur in an area within a given time period. Flood hazard maps are derived from hydrological models and are consequential in determining where a city can build and, in the U.S., which landowners are required to purchase flood insurance.

On the prediction and monitoring side of flood management lie flood occurrence maps. Rather than probabilistic models of where floods may occur, flood occurrence maps (referred to here as flood maps) identify areas where inundation has occurred or is occurring in real time. Flood maps are empirical observations of flooding and are critical tools for governments, aid organizations, and disaster responders. Maps of flooded areas can assist in prioritizing aid resources and relocating displaced persons. When combined with socioeconomic data, these maps can help disaster responders triage relief to the most vulnerable populations. Additionally, flood maps can help assess affected assets like inundated cropland, blocked roads, and damaged infrastructure. In turn, maps of historical flooding can then be used in identifying high-hazard areas, flood

frequency, and serving as validation data for probabilistic flood hazard models. Some insurance companies are even exploring the use of flood maps for parametric insurance, where an agreed upon threshold (such as water volume or inundation in a specified area) triggers insurance payouts if met during a flood event (Johnson, 2017).

This is all complicated by the fact that extreme floods often occur in areas where broad and accurate data is difficult to come by, particularly in rural areas and in the lower income countries. Field-based mapping, while necessary at times for ground truth points, is laborious and expensive for any sizable area. In recent years, however, remote sensing of flooded areas using high and moderate resolution satellites has eclipsed field-based maps for flood monitoring. Passive remote sensing uses airborne or spaceborne imaging spectrometers to record solar radiation that is reflected from the Earth's surface. Differences in the magnitude and wavelength of reflected light can reveal properties of the reflecting objects. Not only is satellite imagery relatively inexpensive for individual consumers, the spatial coverage (roughly 33,000 km^2 for a single Landsat image, for example) and data delivery speed are incomparably better than field measurements. The growing array of operational satellites suitable for flood detection is leading not only to greater coverage of the earth in different wavelengths, but also a substantial corpus of historical flood imagery that can be used to validate models. Cloud-computing environments for analyzing public satellite imagery, like Google Earth Engine, are further reducing the computational requirements for satellite-based flood mapping.

Advantages notwithstanding, remotely sensed flood detection faces several significant challenges. Despite the increasing number of operational satellites, varying orbits and return times can result

in floods that are small or short in duration being completely missed. Floods in small rivers or highly heterogeneous landscapes, like cities, may require finer spatial resolution than public satellites can offer. Atmospheric contamination from haze, cirrus clouds, and especially from optically-thick clouds can obscure images and render them useless. Cloud contamination is of particular concern to flood mapping because the intense precipitation events that often cause flooding are accompanied by thick cloud cover (Fig. 1). At any given time, around 35% of the Earth is covered with clouds (Shen et al., 2015).

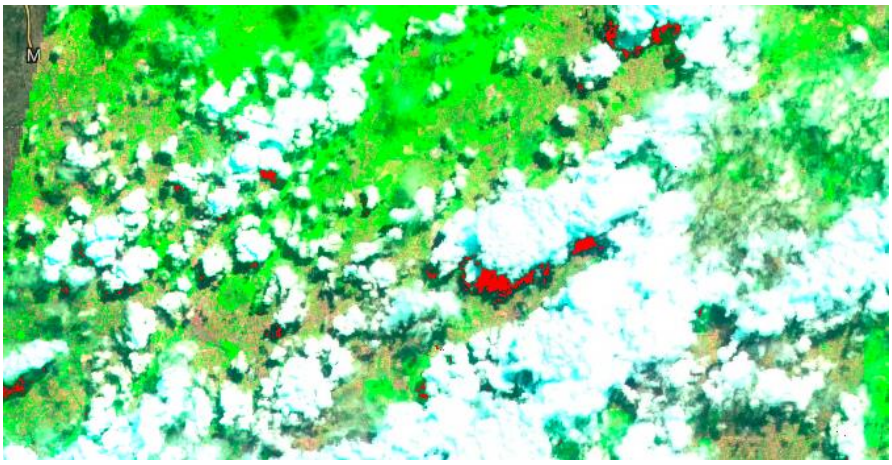

Figure 1.1. Flooding (red) captured by Sentinel-2 obscured by optically-thick clouds in Accra, Ghana (June 2018).

One promising solution to this problem is the use of synthetic aperture radar (SAR). SAR is an active remote sensing system; rather than passively measuring reflected solar radiation, SAR satellites produce and record microwaves in different polarizations. These signals allow operators to create images of the structure of the reflected surface, like roughness or height. Water surfaces are highly discernable in the resulting image, making this a useful method for detecting floods. Critically, the microwaves used by SAR are not absorbed by clouds and can be recorded day and night, effectively eliminating many of the challenges facing optical sensors. However, there are currently only two public and operational SAR satellites with a combined repeat time of 6 days

(the European Space Agency's Sentinel-1A and 1B). While more SAR satellites are coming online in the coming years (Erwin, 2019), most existing sensors at 30m or better resolution record in the visible to near-infrared. This means that the majority of timely flood maps will be built using imagery from these optical sensors.

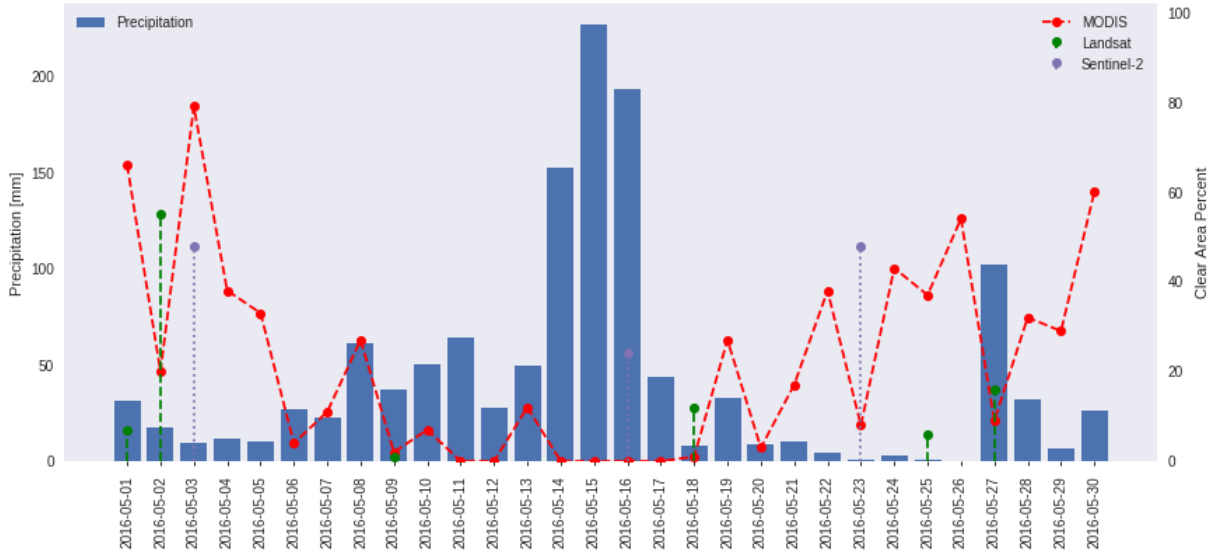

Figure 1.2. This chart illustrates the relationship between precipitation and the percent of pixels in available satellite images that are not obscured by clouds. Clear coverage drops in the middle of May 2016 in Sri Lanka during a storm event.

Currently, a flood map partially obscured by clouds can only offer information on flood extent for the unobscured portions. While this is still useful, the sheer quantity of incomplete flood maps is a massive unsolved problem. A flood map with filled gaps and associated uncertainty for those imputed values would be a serious improvement for end users who would otherwise receive no information from those gaps.

1.1 GAP-FILLING METHODS IN REMOTE SENSING

Gap-filling has been an active area of research in remote sensing for decades (Ng et al., 2017). Bad pixels or missing data in imagery can result from atmospheric contamination, random errors, and sensor malfunction, such as the scan-line corrector failure of the ETM+ sensor aboard Landsat 7.

Given the frequency of missing data in satellite imagery, many algorithms have been developed for gap-filling. These methods can be loosely divided into those based on spatial information (e.g. interpolation from neighboring pixels), those that rely on temporal information (e.g. using pixel values from a different date), those that employ spectral information (e.g. using values of spectra from other bands), and those that utilize a hybrid approach (e.g. interpolating neighboring pixels from a different date) (Gerber et al., 2018; Shen et al., 2015).

Although satisfactory in some scenarios, these methods all have their respective drawbacks. Spatial methods fail when gaps are too large which reduces the pool of similar neighboring pixels that can be used for interpolation (J. Chen et al., 2011; Shen et al., 2015). Spectral methods do not work for gaps where all useful spectra are missing, which is the case for optically-thick clouds. Temporal methods do work well for optically-thick clouds because they use pixels from imagery taken on clear days. However, the underlying assumption of temporal methods is that there is some continuity between images in time; they fail, therefore, when there are abrupt changes between images, such as the construction of a building or a flood (Shen et al., 2015).

Perhaps the most significant challenge to the existing gap-filling algorithms is that serious flooding is, generally, a rare event. This rarity means that algorithms which rely solely on spatial or spectral correlation within an image or temporal correlation between images will be unable to predict a

flood. Even if previous imagery captures regular flooding in an area, it is unlikely that a temporal algorithm would be able to generalize to floods of different magnitude than in the prediction set. This is not to say that spatial correlation is irrelevant – on the contrary, flooded pixels in an image are likely to share many characteristics with each other, such as low elevation or proximity to waterbodies that can overflow. But while viewers can understand these similarities, these characteristics are not encoded into remotely sensed data. An accurate and generalizable algorithm would need to take into account the spatial correlation between features that make an area more likely to flood, and this requires auxiliary data.

1.2 Data-Driven Models

Flood susceptibility models can be roughly characterized as hydraulic models or data-driven models. Hydraulic models simulate flow and inundation in a watershed by solving equations that are derived from physical processes (Solomatine et al., 2008). Although these physical models are useful for simulating floods under varying conditions, they are computationally intensive, need complete monitoring datasets that can be expensive to acquire, and require deep domain knowledge of hydrological parameters (Mosavi et al., 2018). With the advancement of computing power and global spatial datasets, data-driven models have emerged as viable complements or even alternatives to hydraulic models. These models find predictive relationships between features of a hydrological system without explicit knowledge of the underlying physical behavior of the system (Solomatine et al., 2008). Although data-driven models require large datasets at sufficient coverage and spatial resolution, they have the advantage of being flexible, computationally efficient, and can be deployed for rapid flood mapping in response to storm warnings.

One of the most common uses of data-driven models is the estimation of *a priori* flood hazard, or the probability of inundation in an area under certain circumstances as mentioned in Section 1.1 (Han, 2011). To this end, significant research has been conducted into factors most associated with flooding. Previous research has used combinations of topographical data derived from digital elevation models (DEMs), lithography, soil type, land cover, and rainfall measurements to estimate the flood susceptibility of an area (Chapi et al., 2017; Choubin et al., 2019; Janizadeh et al., 2019; Mojaddadi et al., 2017; Tehrany et al., 2013).

These methods report high performance (>85% accuracy and AUC) on test sets when predicting flooding. However, all of the methods reviewed used high resolution local data sets, such as 5m DEMs or local soil maps (Janizadeh et al., 2019; Mojaddadi et al., 2017). Currently the only DEM with global coverage is NASA's 2000 Shuttle Radar Topography Mission at 30m resolution. In fact, the unfulfilled need for a precise, high resolution, global DEM is seen as a critical obstacle to the development of accurate flood hazard maps in otherwise data-poor regions (Guy J-P. Schumann & Bates, 2018; Guy J.-P. Schumann et al., 2014). Likewise for other flood conditioning features. While it is possible to use interferometric SAR data to generate higher resolution DEMs, this is a complex process and has not been completed for the entire globe (Vassileva et al., 2017). As such, those in need of high resolution DEMs must procure commercial airborne LIDAR data which can be expensive.

With the release of Google Earth Engine, global remote sensing datasets have become much easier to access. Still, as of 2020, only moderate resolution (30m or coarser) datasets are available on

Earth Engine. Whether flood prediction and gap-filling methods for flood maps can be performed successfully using these data has not been explored in the literature.

1.3 Uncertainty estimation

Developing a model that can generate predictions is useful but often not sufficient. If the predictions are to form the basis of important decisions, such as medical diagnoses or autonomous vehicle driving, users need to understand how much confidence they can place in them. This is especially true for inverse ill-posed problems, like gap-filling, where verifying the predictions may be impossible. Yet confidence or uncertainty estimates are notably absent from most published gap-filling and flood prediction research. Gerber et al. (2018) reviewed a selection of twenty published gap-filling algorithms and found that only seven provided uncertainty quantification of the predicted values (Gerber et al., 2018). One reason for this absence is that uncertainty estimation is especially difficult for deep learning models, like neural networks, which often only provide point estimates of parameters or predictions (Gal, 2016). Inevitably, this leads to overly confident predictions even when the model is returning output no better than a guess. Unboxing these "black boxes" is an active area of research in deep learning with implications for myriad fields (Kendall & Gal, 2017; Kwon et al., 2018; Wang et al., 2019).

It is important to note here that there are multiple kinds of uncertainty. The two relevant types here are epistemic and aleatoric uncertainty. Epistemic (or structural) uncertainty is the uncertainty that comes with the modeling process or limited data, and it can be minimized with an infinitely large dataset. In flood mapping, epistemic uncertainty may arise if the gap-filling model is trained on an area of pluvial flooding but then asked to predict a flood caused by dam failure. Aleatoric (or statistical) uncertainty is the random noise inherent in every dataset, like varying brightness levels

in satellite images or unusually high or low feature values. Aleatoric uncertainty is intrinsic to the data – if two pixels share the same exact feature values but one is flooded and one is not, a model will not accurately discriminate between them no matter how well it is fitted.

Uncertainty can be estimated from machine learning models through the use of both frequentist and Bayesian methods. Rather than just using point estimates of weights, Bayesian neural networks (BNNs) place probability distributions over each weight, approximating a Gaussian process (Gal, 2016). Then, the model does multiple forward runs with values sampled from these distributions, resulting in a posterior predictive distribution of output values rather than just a point estimate. Sampling from this distribution allows one to estimate the mean and variance of the expected target feature values.

But placing entire probability distributions over the weights increases the number of parameters considerably and is computationally expensive. Gal and Ghahramani found that Monte Carlo dropout can be added to models to approximate a Gaussian process without the need to build weight distributions (Gal & Ghahramani, 2015b, 2015a). In this method, neurons (or connections, see section 2.6.3) in the network are randomly dropped during training and testing and the model is run many times to yield a distribution of outcomes. These outcomes can be interpreted as samples from a probabilistic distribution, and epistemic uncertainty can be calculated from this distribution. To estimate aleatoric uncertainty, Kwon et al. improves on Kendall and Gal's original method of estimating input mean and variance during the last layer of the Bayesian neural network and instead estimate the variability of the predictive probability directly after training using the Monte Carlo dropout predicted values (Kwon et al., 2018). These methods are relatively new, and

some have questioned the accuracy of Monte Carlo dropout uncertainty estimates, noting that in some tests the uncertainty does not decrease with the addition of more data (Rothmann, 2019)

1.4 OBJECTIVES

The overall objective of this thesis is to develop a method for filling the gaps in cloud-obscured flood maps derived from satellite imagery. Specifically, I pursue the following goals:

1. Develop modeling methods and a testing framework for predicting flood inundation beneath clouds in Landsat 8 imagery.
2. Compare the predictive performance of these methods.
3. Compare how the uncertainty measures from logistic regression and Bayesian neural network models track with prediction errors
4. Discuss model robustness and applications, implications for flood mapping, and avenues for future research.

Chapter 2. METHODOLOGY

The general approach I employed for this work is as follows: imagery and feature data were acquired for the study areas, then masked with procedurally generated clouds; the data are used to train logistic regression, random forest, and neural network models to predict flooding in the masked images; these models are evaluated along multiple performance metrics; finally, the best performing models are used to generate estimates of uncertainty for flood predictions.

It should be noted here that the models were trained and tested on the same image. Usually, models are trained on very large datasets consisting of hundreds or thousands of images so that they can

generalize well to a new image. However, training a generalized model here for predicting flooding in cloud gaps would simply be estimating *a priori* flood hazard. For instance, two images taken in the same location at different times would have identical flood predictions because the underlying features are generally static. When the model is trained and tested on the same image, it learns weights for the physical characteristics associated with the particular flood event in that image. The assumption is that the pixels in an image are spatially autocorrelated and share similar physical features that drive flooding. An overview of the methods for this study are illustrated in Figure 2.1.

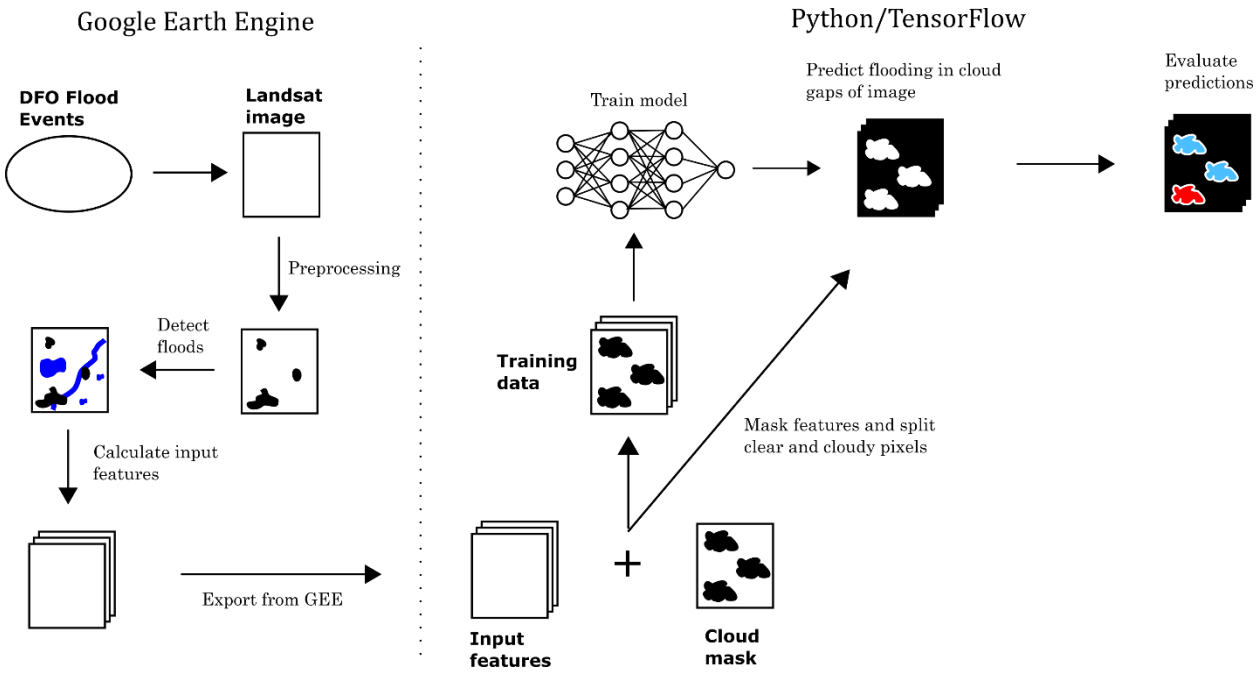

Figure 2.1. Workflow for preprocessing of data in Google Earth Engine and training, prediction, and evaluation of models in Python

2.1 IMAGERY SELECTION

This study used imagery from Landsat 8, which began in 2014 and is currently operational, mainly because the spatial resolution (30m) is equivalent to the input features and there are good snow,

cloud, and cloud shadow masks available with the Tier 1 Surface Reflectance product. Landsat 7, while still operational at the time of this study, suffers from a scan-line corrector failure and would have introduced unwanted gaps into the study imagery.

Although floods are a global phenomenon, this research focused on flooding in the U.S. This was done to maintain a manageable study area and utilize the National Land Cover Dataset for the U.S., as no 30m land cover dataset is available globally. The date and approximate geographic extent of flood events were identified using data from the Dartmouth Flood Observatory which records flood events around the world using information from news media, governments, and satellite imagery (Brakenridge, 2019). The flood extent boundaries recorded by the DFO are very rough polygons of the general area in which a flood occurred; while the DFO does produce more detailed flood maps, they suffer from cloud gaps as well. In Google Earth Engine, Landsat 8 imagery was selected that coincided with the approximate flood extent and event duration recorded by the Dartmouth Flood Observatory. Flood events were chosen based on the availability of clear (<15% cloud cover) Landsat 8 imagery, and were manually inspected for visible flooding. Snow, clouds, and cloud shadows were masked out in each chosen image using the quality assessment bands. Images were divided into smaller images to avoid large areas of missing data, increase the study set, and reduce computational overhead. All of these models were trained on CPU so they would be as accessible as possible. CPU, although inexpensive and more common than GPU, is also less capable of parallel computations. As such, image size and memory requirements had to be taken into account.

2.2 FLOOD CONDITIONING FEATURES

Input features in the auxiliary data were chosen based on those commonly used in the literature for flood susceptibility mapping (Kia et al., 2012; Mojaddadi et al., 2017; Tehrany et al., 2014). All datasets are downloaded and all derivative features are calculated in Earth Engine, which is a cloud-computing platform for remote sensing data written in JavaScript (Gorelick et al., 2017). All features are shown in Table 1.

Table 1: Input features

Feature	Source	Resolution, units, data type
Elevation	SRTM DEM	30m, meters, float
Slope	SRTM DEM	30m, degrees, Float
Curvature	SRTM DEM	30m, degrees, float
Aspect	SRTM DEM	30m, degrees, float
Height above nearest drainage	Donchyts et al., 2016	30m, float
Topographical wetness index	HRNHD Plus (based on SRTM DEM)	30m, unitless, float
Stream power index	HRNHD Plus (based on SRTM DEM)	30m, unitless, float
Distance from permanent/seasonal water	Joint Research Centre	30m, meters, float
Land cover (Forested, planted, wetlands, developed, other)	National Land Cover Dataset	30m, binary

The majority of features are derived from the 2000 Shuttle Radar Topography Mission (SRTM) 30m Digital Elevation Model (DEM). Topographical wetness index and stream power index were calculated using the 30m flow accumulation layers from the High Resolution National Hydrography Dataset Plus (HRNHD Plus).

Slope ranges from 0 to 90 degrees and controls runoff and the rate of infiltration, with flatter areas experiencing slower infiltration and higher susceptibility to flooding (Kia et al., 2012; Lawal et

al., 2014; Nobre et al., 2011). Flooding is impossible on steep surfaces, making slope a useful feature for ruling out flooding.

Curvature is the derivative of the slope and can be convex, concave, or flat. It can indicate areas where water flow accelerates or decelerates over the surface (Hallema et al., 2016; Pourghasemi et al., 2019).

Aspect is the direction a surface faces which determines direct sunlight exposure and correlates with soil moisture, plant root development, and weathering (Dai et al., 2001; Haghizadeh et al., 2017).

The height above nearest drainage is a normalized measure of elevation calculated from the nearest drainage point of a flow network rather than sea level. This results in a terrain model that better describes elevation with respect to hydrology (Nobre et al., 2011). The dataset was created by Donchyts, et al., 2016.

Stream power index is a measure of the erosive power of overland flow. Stream power index increases with catchment size and slope as a greater water volume flowing at a higher speed can lead to flooding (Mojaddadi et al., 2017). Topographic wetness index is a measure of flow accumulation at any given location within a river catchment and is widely used in flood modeling (Kelleher & McPhillips, 2020; Ma et al., 2019). They are calculated using Eqs. 1 and 2, respectively,

$$SPI = A\tan\beta \tag{2. 1}$$

$$TWI = \ln\left(\frac{A}{\tan\beta}\right) \quad (2.\ 2)$$

where A is flow accumulation and β is slope.

The SRTM-derived features are calculated for the U.S. Geological Survey HUC-6 sub-watershed underneath each image. The features are calculated for sub-watershed, rather than just the image area, because some features such as curvature use a kernel which would leave edge pixels with fewer neighbors and could yield inaccurate results. Using sub-watersheds that are larger than the image area removes this problem. All features are added to the original image as additional bands.

The permanent water and distance from permanent/seasonal water features were derived from the Global Surface Water dataset released by the European Commission's Joint Research Centre (Pekel et al., 2016). This dataset maps the spatial and temporal occurrence of surface water across the globe from a composite of 32 years of Landsat 5, 7, and 8 imagery. The definition of permanent water is important because it allows one to distinguish water that is a flood from water that is not a flood. In this study, areas that are underwater throughout the entire year are defined as permanent waterbodies. Intuitively, flood hazard decreases with distance from water features. Although rainfall was used in some of the literature, the spatial resolution of the available global rainfall datasets was too coarse for this study.

Finally, land cover classes come from the National Land Cover Dataset a 30m classification of the U.S. into various land cover classes based on Landsat imagery (Jin et al., 2013). Different types of overlying land cover can have a significant effect on flood susceptibility – vegetation tends to reduce the magnitude and velocity of runoff while impervious surfaces and farmland accelerate it

(Tehrany et al., 2013). Wickham et al. carried out a thematic accuracy assessment of the 2011 National Land Cover Dataset and found agreement between the dataset and reference data of between 82-89%, depending on land cover class (Wickham et al., 2017). But even with reasonable accuracy levels there is the potential for change over time between the National Land Cover Dataset and a flood event after 2011. A sample of all features in one image are shown in Figure 2.2.

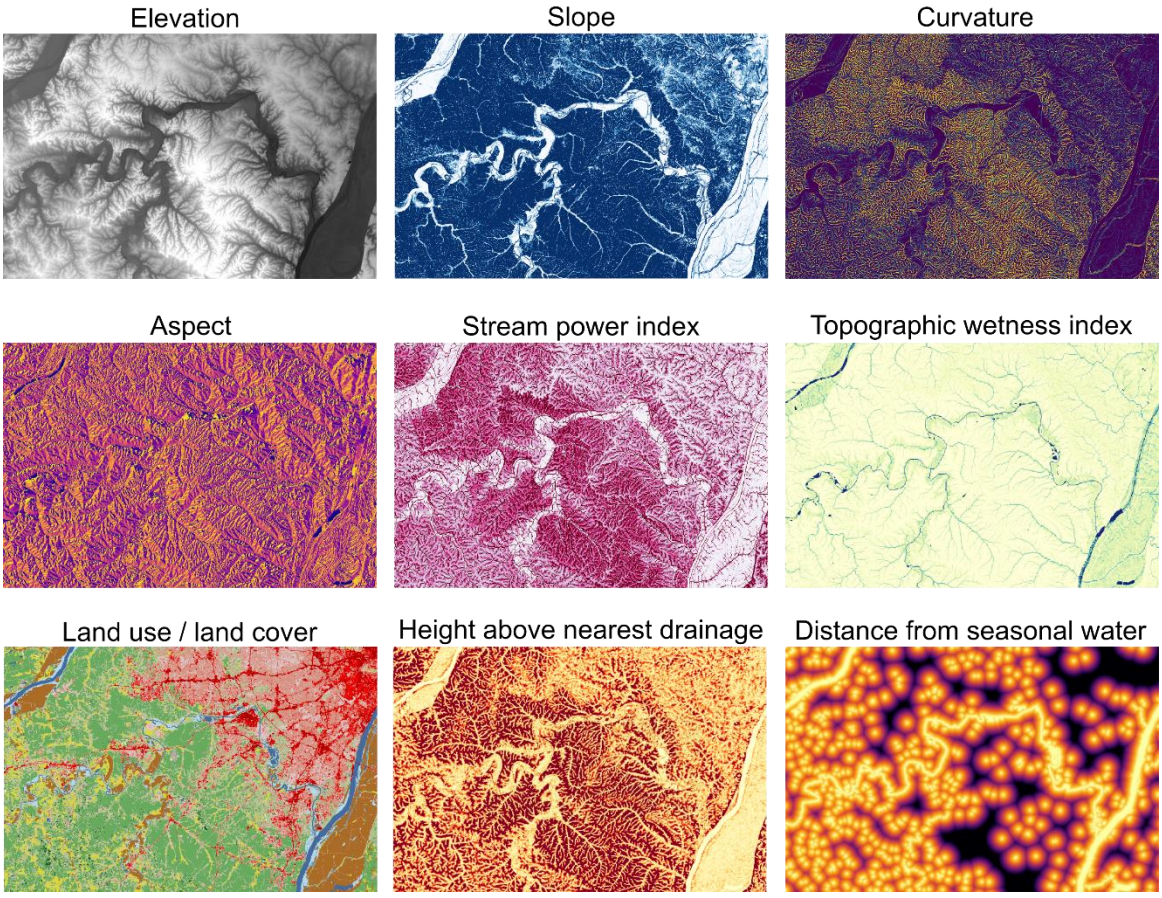

Figure 2.2. Input features from a sample region of an image

2.3 FLOOD DETECTION

The most important feature is the target feature, a binary layer indicating whether a pixel is flooded or not. There are numerous methods to detect flooding in multispectral satellite imagery, the most common being band ratios like the Normalized Difference Water Index (NDWI) and its improvement, the modified NDWI, although the application of machine learning algorithms to flood detection is becoming common as well. These ratios take advantage of the fact that water absorbs light strongly in the near infrared. The modified NDWI (Eq. 3) generates values between -1 and 1 and the resulting image is thresholded at zero, with values < 0 indicating dry land and values > 0 indicating water (Xu, 2006).

$$MNDWI = \left(\frac{\rho_{green} + \rho_{NIR}}{\rho_{green} - \rho_{NIR}} \right) \tag{2. 3}$$

where ρ is the reflectance in the green and near infrared wavelengths, respectively. However, this is simply an heuristic, and choosing more appropriate threshold values (by using Otsu's method to minimize in-class variance, for example) with high accuracy is time consuming and must be done for each individual image (Otsu, 1979).

A more recent method is the Automated Water Extraction Index (AWEI) (Feyisa et al., 2014). AWEI is a linear equation with coefficients derived empirically under different levels of environmental noise (Eq. 4).

$$AWEI = 4 \times \left(\rho_{green} - \rho_{SWIR1} \right) - \left(0.25 \times \rho_{NIR} + 2.75 \times \rho_{SWIR2} \right) \tag{2. 4}$$

where ρ is the reflectance value of the green, shortwave infrared (SWIR1 and SWIR 2), and near infrared (NIR) bands, respectively. This study uses AWEI for flood detection because it has improved accuracy over the modified NDWI and maximum likelihood classifiers while conveniently offering a stable threshold at zero (Feyisa et al., 2014).

2.4 CLOUD SIMULATION

Although the ultimate goal for this method is to predict inundation for cloud gaps, using images with naturally occurring cloud cover to train the classifier would make validating the method impossible. One potential solution might be to use optical imagery that has complementary SAR imagery in the same area and time period. However, this greatly reduces the pool of available imagery, and would be subject to additional errors from flood expansion/retreat between image acquisition times. The solution used in this study is to generate artificial clouds to mask out clear imagery. In this way, reflectance values beneath the clouds are known and cloud cover can be experimentally controlled.

Cloud cover was created using the Perlin noise algorithm which was developed for use in movie and video game graphics (Perlin, 2002) and has also been used to simulate clouds in satellite imagery before (Enomoto et al., 2017). Perlin noise is useful because it procedurally generates random yet structured gradient noise and can be easily thresholded at create more or less cloud cover. In this study, cloud masks were generated at varying levels of cloud cover for each image (10, 30, 50, 70, and 90%) (Figure 2.3). Each image retained the same cloud mask for all models, except for the random cloud trials which were undertaken to test the robustness of the models and

importance of randomness in pixel masking. During these trials, new cloud masks were randomly generated for each image during each trial.

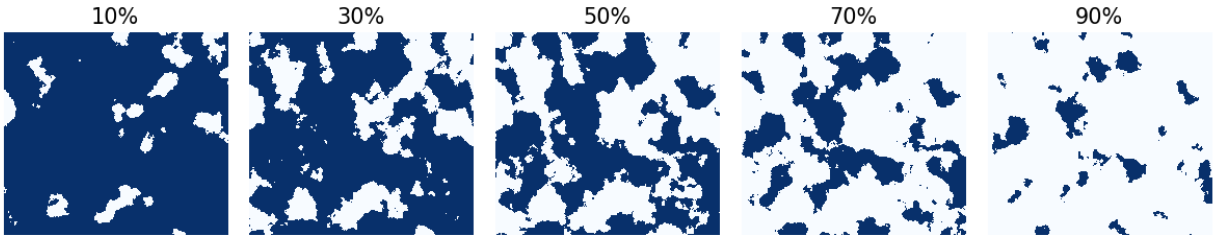

Figure 2.3. Cloud masks generated with Perlin noise and thresholded at different cloud cover percentages.

2.5 PREPROCESSING

Feature data for each image were exported from Google Earth Engine and stacked into 3-dimensional matrices $\boldsymbol{M}_{m\times n\times k}$ for k features and $m\times n$ pixels. In Python, the data were masked by clouds, reshaped into 2-dimensional matrices $\boldsymbol{N}_{(m\times n)\times k}$, and standardized to have mean 0 and variance 1. Any feature that had a standard deviation of 0 (such as a land cover class that was not represented in a given image after cloud masking) was removed from the matrix $\boldsymbol{N}$. During training, the target feature was composed of all detected water, but during prediction and evaluation any water pixel that coincided with a permanent water pixel was revalued as a non-flooded pixel. In other words, permanent water is included in training but did not count towards model performance. This led to the highest performance for the baseline logistic regression model when compared to training with only flood pixels (Figures 1B and 2B).

Because the dataset was comprised of the entire image (except pixels masked out due to snow, ice, or clouds) data sparsity was a concern. Sparsity – where one class is greatly underrepresented in

the dataset, in this case the number of flood pixels ≪ number of dry pixels – can lead to very conservative or no predictions because the model fails to learn. On average, 3-12% of the images were covered by floods. To test if the dataset should instead be a sample of the image, the baseline linear regression model was trained on datasets consisting of pixels within a buffer around all detected water and a number of randomly sampled pixels. For buffers and samples of varying sizes, the linear regression model performed very poorly with a high false positive prediction rate compared to training on the entire image, likely due to the oversampling of flood pixels in the buffer (Table 1B).

2.6 Model Selection

Logistic regression, random forest, and neural network models were trained with the preprocessed images at varying cloud coverages. Logistic regression was chosen for speed and the interpretability of its weights and was used as the "baseline model" when designing the preprocessing framework. Random forest is used extensively in pixel-based classification problems, including flood detection and land cover classification (Mosavi et al., 2018; Woznicki et al., 2019). Neural networks are also used extensively in the literature and tend to outperform other machine learning algorithms for classification and regression tasks in remote sensing problems. Detailed description of model specifications can be found in Appendix A.

Notably missing from this selection of models is a convolutional neural network, or CNN. Convolutional neural networks are commonly used for image processing and classification problems and perform very well because they learn the high-level visual and spatial semantics of an image, like edges and shapes and other contextual elements, rather than just the pixel-based

values. Despite the high performance in other problems, however, flood prediction in cloud-obscured imagery is a particularly vexing problem for the currently developed CNN architectures because of missing data. In neural networks and most other models, observations with missing data can simply be removed from the dataset if desired. In traditional Convolutional neural networks, however, missing pixels must be imputed prior to training (with, for example. the mean or K-nearest neighbors interpolation; see section 1.1) because they comprise the shape of the image. Using Convolutional neural networks themselves to impute missing data is an active area of research. Previous efforts to reconstruct satellite imagery have relied on multi-temporal data (M. Chen et al., 2019; Wu et al., 2019) or a mixture of spatiotemporal and spectral data (Q. Zhang et al., 2018). Unfortunately, these methods require either a large number of training images, or a reference image obtained from the same area from which high-level structures can be learned. For Landsat 8, there are only an estimated 20-35 relatively cloud-free images available for flood events in the U.S. (based on the Dartmouth Flood Observatory), which is a very small training set for a generalizable CNN. Multi-temporal imagery that also contains flooding over the same area is likewise unavailable given the (by definition) low frequency of moderate to extreme flooding events. Training a CNN on the same image it predicts is also problematic because the image dimensions must be identical.

2.6.1 *Logistic Regression*

Logistic regression is a statistical method for predicting binary classes and is one of the most commonly used machine learning algorithms for classification.

$$\ell = \beta_0 + \beta_1 x_1$$
$$\ell = \log \frac{p}{1-p} \qquad (2.5)$$

Input data are mapped onto a scale from 0 to 1 and combined linearly with weights β and coefficient values x that are learned through training. The trained model is a linear equation between the input features and the log-odds ℓ that can be used to predict the probabilities p of the binary target feature. The models were trained in Python using the *scikit-learn* package (Pedregosa et al., 2011). To train the models, the negative log-likelihood of the parameters was minimized using a stochastic average gradient descent solver, which allowed for multi-thread processing and thus faster training in Python.

2.6.2 *Random Forest*

Random forest (RF) is a supervised ensemble machine-learning algorithm, meaning it is a composite of multiple algorithms that perform better in combination than alone. The constituent algorithms are called decision trees and they operate in a similar way to flowcharts – each observation of the input data reaches a "leaf" of the tree which contains a discrete decision rule (e.g. elevation=0-150 feet, elevation=151-500 feet, etc.) that determines the next leaf, and so on until reaching a probability for each of the target classes. The random forest constructs many decision trees from random subsets of the dataset and then these trees vote on a final outcome based on their individual output probabilities. By introducing randomness into model training and growing complex decision trees, random forest can outperform individual trees and prevent overfitting on the training data. Hyperparameters (parameters which cannot be learned and must be set prior to training) were optimized through a random grid search using the Python package *scikit-optimize*; the model was trained using *scikit-learn.*

2.6.3 *Neural Network*

Loosely based on the organization of the human nervous system, neural networks (NN) are comprised of a fundamental unit called a neuron. This neuron takes input values, performs some operation on them (usually multiplying them by learned weights and biases), then passes the resulting value through an activation function to return a normalized result (usually between 0-1). Neural networks can be composed of multiple layers, each containing many neurons that represent abstract features of the dataset. During training, the network will compare the final output to the actual target value in the training data and adjust the weights and biases accordingly in a process called backpropagation – the "learning" part of machine learning. Multiple layers allows a neural network to classify linearly non-separable data, and additional layers allow for more refined feature abstraction in the network and contribute to far greater performance on tasks like image classification (Goodfellow et al., 2016). Neural networks with many layers are often called "deep neural networks."

The neural network in this study was built using the Keras API in TensorFlow and used two dense layers with Rectified Linear Unit (ReLU) activation functions and batch normalization. ReLU has been shown to offer some computation and convergence improvements over other activation functions, like sigmoid (Nair & Hinton, 2010). Batch normalization improves speed and reduces overfitting by normalizing the outputs of each activation function so that the model can generalize to datasets that may have different distributions (Ioffe & Szegedy, 2015).

2.7 EVALUATION

For each image (n = 28) and cloud cover mask ($\theta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$), cloud-free pixels trained a model which was then used to predict flooding in the cloud-covered pixels. This resulted in 140 models and predictions for each of the three model types.

Models were evaluated based on their accuracy, precision, recall, and F1 and AUC scores. Given that the datasets were sparse (3-12% flooded pixels) accuracy and AUC, or area under the receiver-operator curve, will likely be biased towards higher values (Table 2.2).

Table 2.2. Evaluation metrics calculated from true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN)

Metric	Formula	Description
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	Fraction of correctly identified pixels out of total pixels
Recall	$\frac{TP}{TP + FN}$	Fraction of correctly identified flood pixels out of all true flood pixels
Precision	$\frac{TP}{TP + FP}$	Fraction of correctly identified flood pixels out of all pixels identified as floods
F1 Score	$2 \times \frac{precision \times \text{recall}}{precision + recall}$	Harmonic mean of precision and recall

A model with high recall means that most of the flood pixels in the image were correctly identified. A model with high precision means that the pixels identified as flooded were precisely identified as such, with few false positives. Precision and recall are inversely related – a model that identified every pixel as flooded would achieve perfect recall but very low precision. The F1 score is a compromise between recall and precision, and is based on the harmonic rather than arithmetic mean of the two metrics in order to punish low values for either metric. For this reason, F1 score was the single most important performance metric in this study.

2.8 UNCERTAINTY

Two methods of estimating prediction uncertainty are compared. In the first method, a two-layer Bayesian neural network (BNN) was constructed with batch normalization and dropout after the dense layers. Dropout rates of 0.2 – 0.4 are commonly used in neural networks for regularization purposes; 0.2 was used here after preliminary testing, meaning 20% of the input data was randomly omitted during each training epoch. To estimate uncertainty, a Monte Carlo distribution of predicted probabilities $\hat{p}$ was calculated by predicting on test data T times. Aleatoric and epistemic uncertainties are calculated separately and combined to estimate total uncertainty U,

$$U = \underbrace{\frac{1}{T}\sum_{t=1}^{T} diag\left(p_t\right) - p_t^{\otimes 2}}_{aleatoric} + \underbrace{\frac{1}{T}\sum_{t=1}^{T}\left(p_t - \bar{p}\right)^{\otimes 2}}_{epistemic} \quad (2.6)$$

The mean prediction of all the Monte Carlo passes, $\bar{p}$, was used as the model prediction for a given pixel.

In the second method, a frequentist approach was used that computed standard errors from a logistic regression model. For each pixel, a confidence interval was estimated as the difference between the lower and upper bounds of the prediction.

Chapter 3. RESULTS AND DISCUSSION

Across all images, there was no strong linear correlation between flooding and any one feature (Figure 3.1).

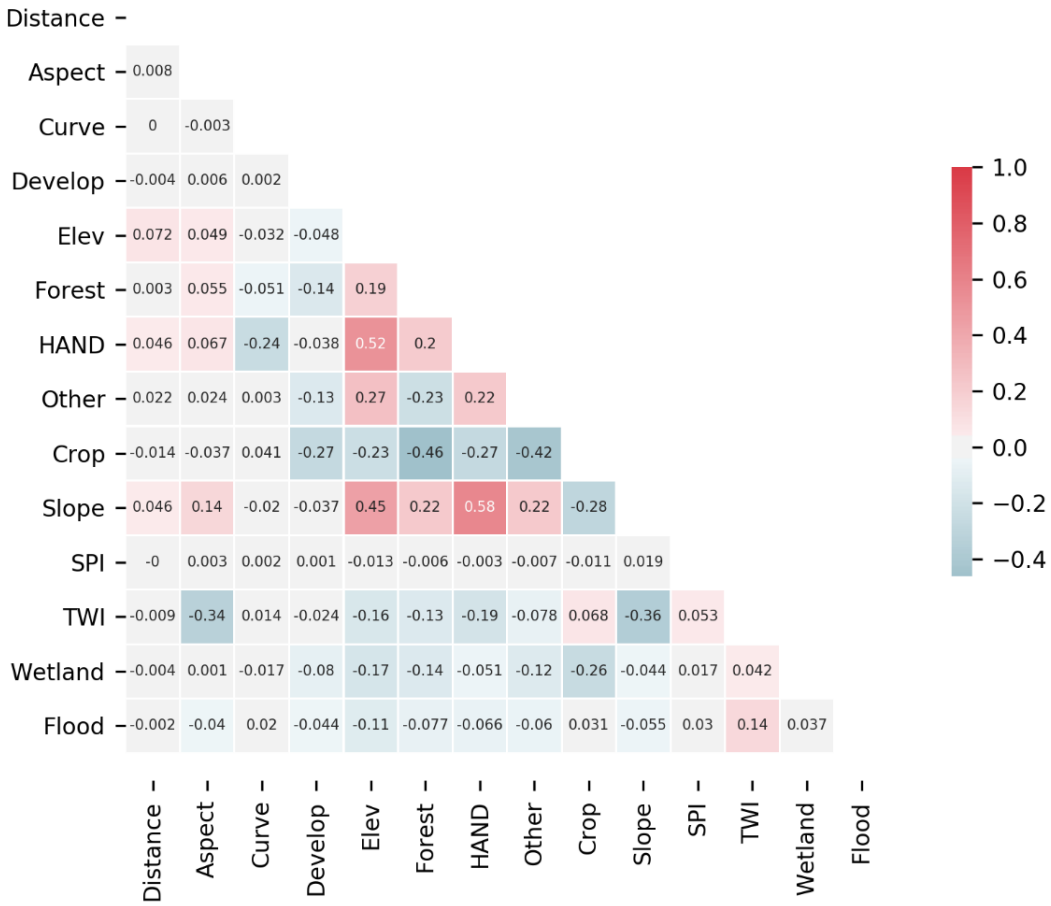

Figure 3.1. Correlation of input variables for all 28 images. Correlation between continuous variables was calculated with Pearson's coefficient, binary variables with point biserial coefficient.

Overall, the models had consistently high accuracy and AUC scores. These correct predictions were inflated from the large number of true negatives though. Most models had difficulty correctly

identifying the relatively few flooded pixels in the images, as reflected in their low recall scores. Precision was generally higher on account of these conservative predictions.

The neural network and logistic regression had nearly identical performance, with more positive predictions from the neural network resulting in higher recall and lower precision (Figure 3.2). The predictions from the neural network were less consistent than logistic regression, with over 70% of flood pixels identified in some images and nearly 0% in others. Even after hyperparameter tuning, random forest performed worse than the other models and with much greater variability from image to image.

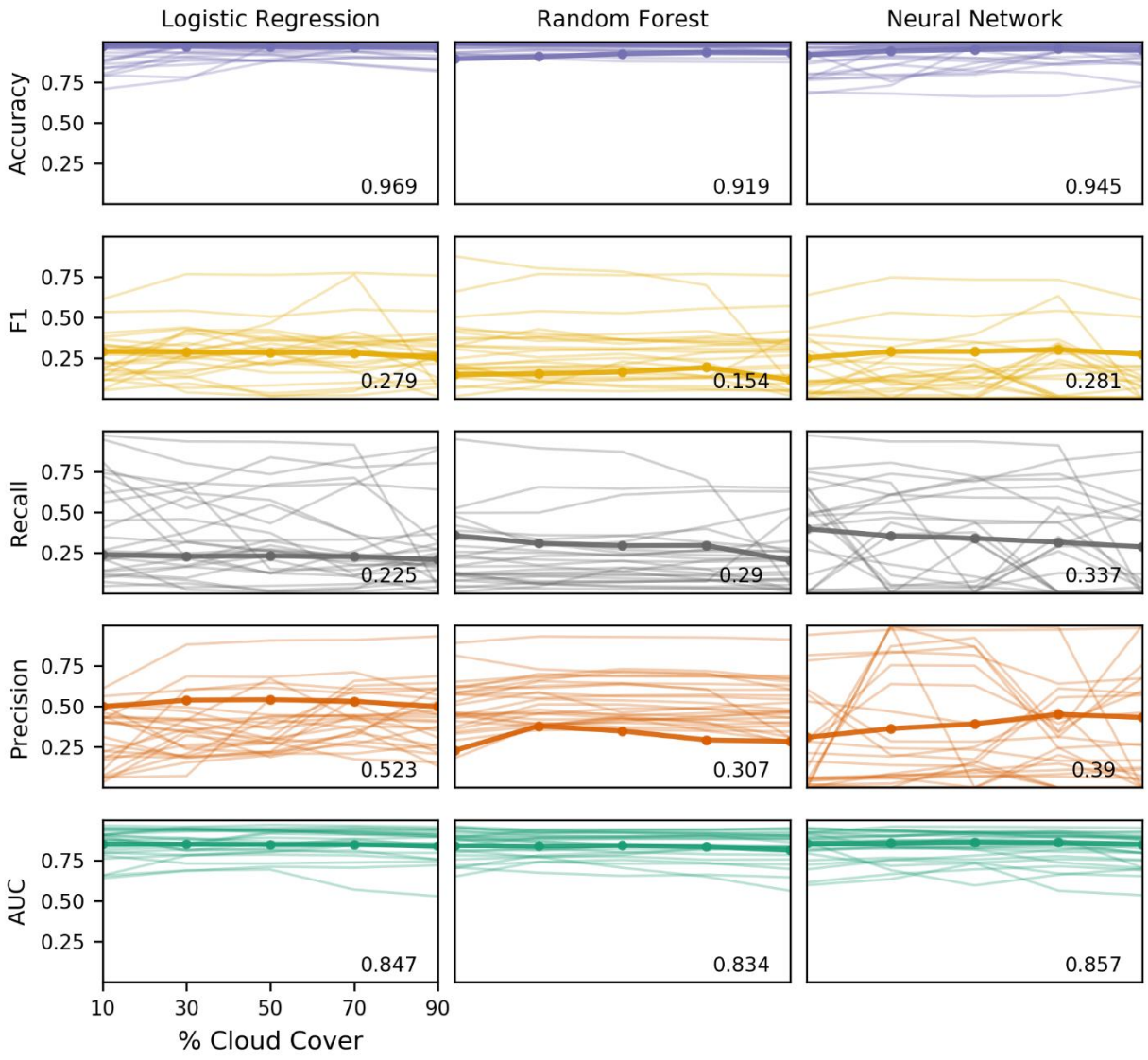

Figure 3.2. Mean performance metrics of logistic regression, random forest, and neural network models. Each point represents the average score for that model at a given percent of cloud cover. Average across all cloud covers is noted in the corner.

Surprisingly, there was no clear drop in performance with increasing cloud cover. On average though, recall tended to decrease slightly with cloud cover while precision increased slightly or remained stable, indicating that the models were predicting less flooding when there were fewer training pixels to learn from.

Overall, the models performed better in some images and not as well in others. Performance metrics were comparable to those reported in the flood susceptibility literature, though many of the studies only tested on sample points rather than every pixel available in an image (Bui et al., 2018; Janizadeh et al., 2019; Mojaddadi et al., 2017). The availability of many pixels with low flood hazard inflated the accuracy and AUC scores considerably here.

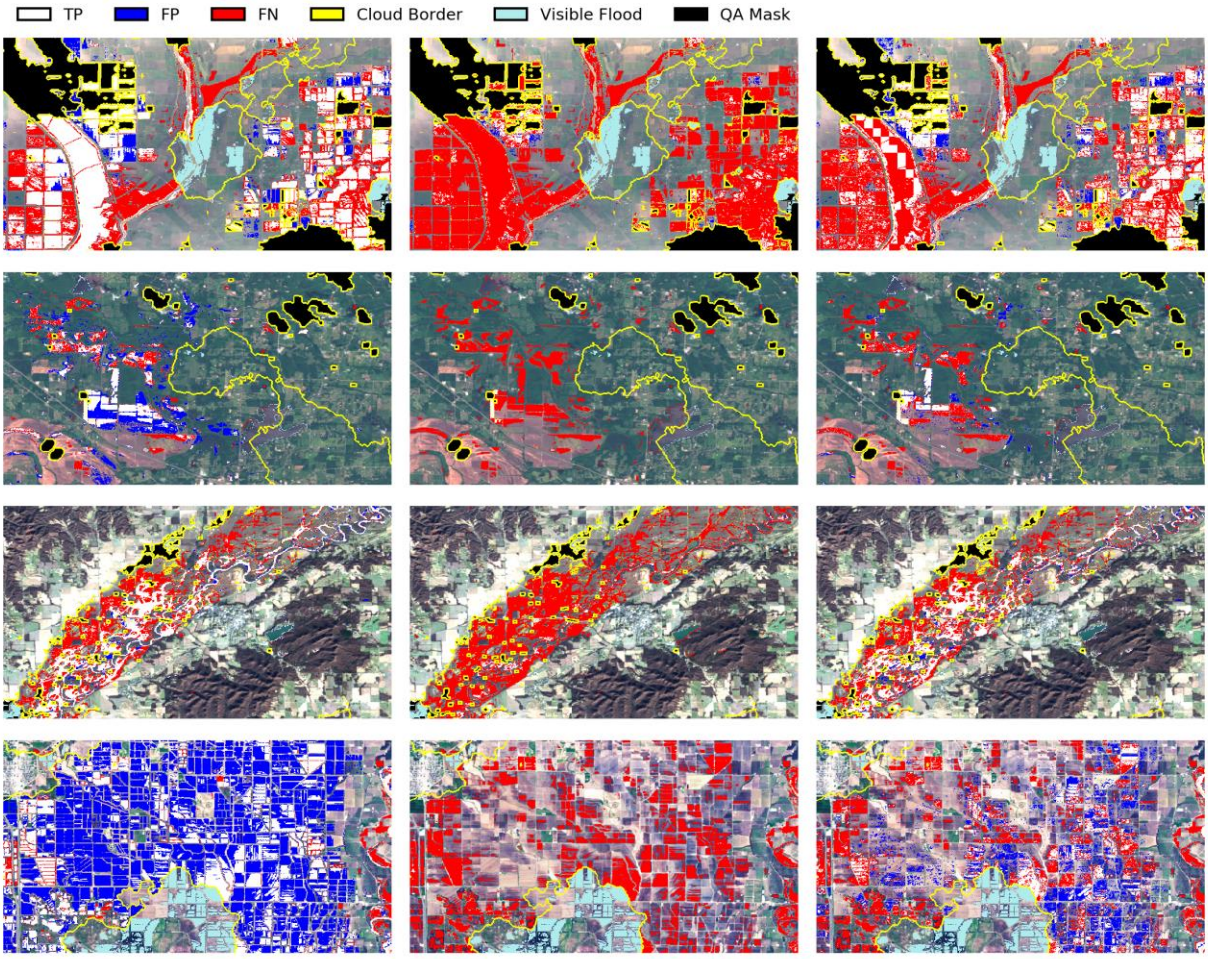

Figure 3.3. Comparison of model predictions in segments from four images. Outline of cloud borders depicted in yellow; actual clouds were removed using the Landsat QA band and are shown in black. Model predictions are true positive (TP), false positive (FP), or false negative (FN).

Although the models were generally robust to cloud coverage, the placement of those clouds had a significant impact on performance. The random cloud trials with five different cloud masks yielded wildly different results for many images (Figure 3.4).

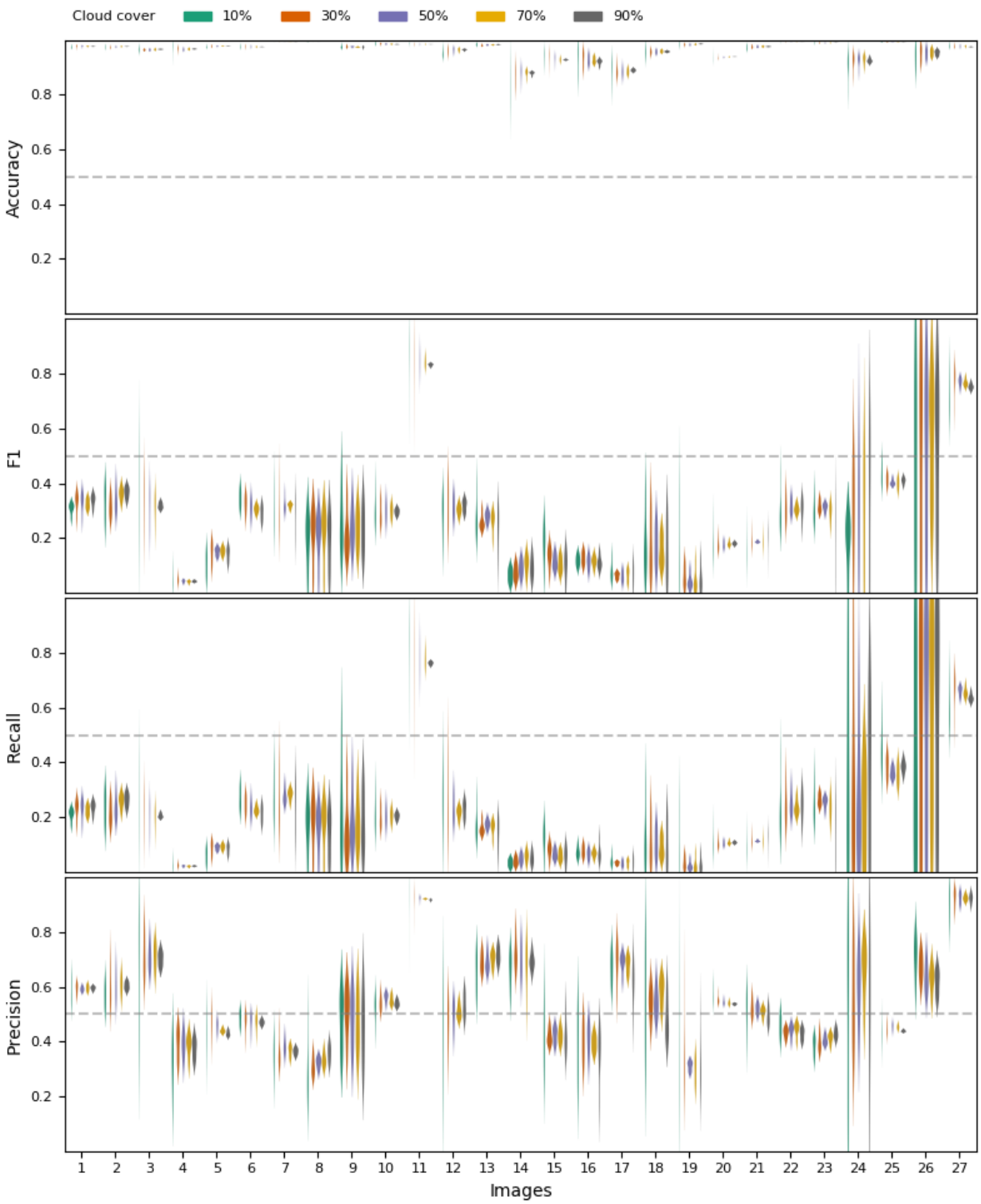

Figure 3.4. Performance metrics for each image across five runs of the logistic regression model with different randomly generated cloud masks. Metrics for each image at a given cloud

cover are represented by a rotated kernel density plot. Images with long density plots have highly varying performance between random cloud trials.

While there was little variation in accuracy, shuffling the placement of clouds caused the other metrics in about half of the images to vary substantially from trial to trial (Figure 3.5). A number of tests were conducted to explain the variance. Train and test data were compared for each image across trials but there was no discernable relationship between performance and differences in feature means, variances, or entropies (Figures 3A-5A). This rules out a distributional shift between train and test sets as the cause of performance variability across trials. Nor was there any relationship between performance and the number of flooded pixels available to each training set (Figure 6A). Given the sparsity of flooding in any image, obscuring just a few dozen flooded pixels could have an outsized effect on model learning that would not be substantial enough to alter the distribution.

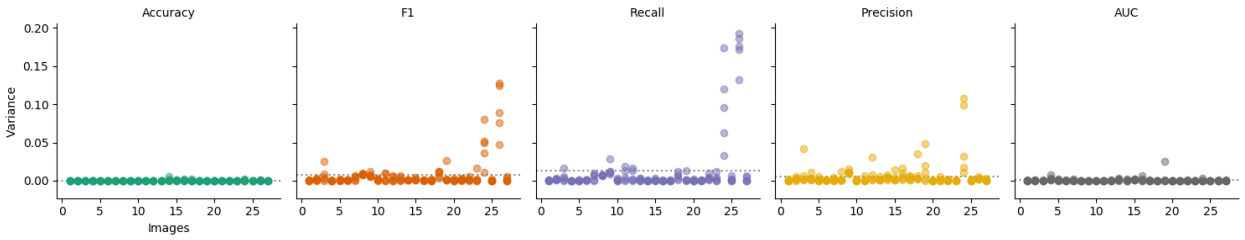

Figure 3.5. Variance of performance metrics for each image across five runs of the logistic regression model with different randomly generated cloud masks. Mean variance is noted with a dotted line.

Notably, some images had consistently high or low metrics across trials. This suggests that for some images, regardless of cloud cover, the flood conditioning factors simply did not have a strong relationship to flooding.

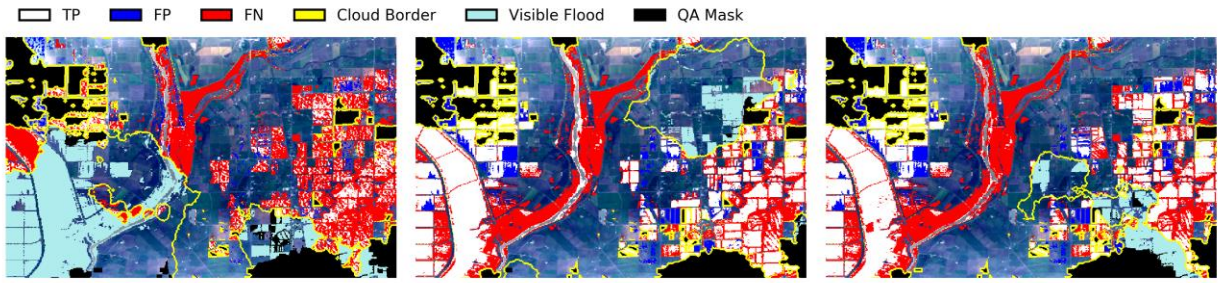

Figure 3.6. Predictions varied greatly with the placement of cloud cover. In this image, flooding in farmland is predicted differently based on which pixels are obscured. Model predictions are true positive (TP), false positive (FP), or false negative (FN).

With such variability in predictions, it is crucial that the model provide some measure of its uncertainty. Uncertainty estimates from the Bayesian neural network and logistic regression were compared alongside predictions to see how well they tracked with errors. As depicted in the histograms of Figure 3.7, the two models appeared to have nearly opposite prediction and uncertainty relationships.

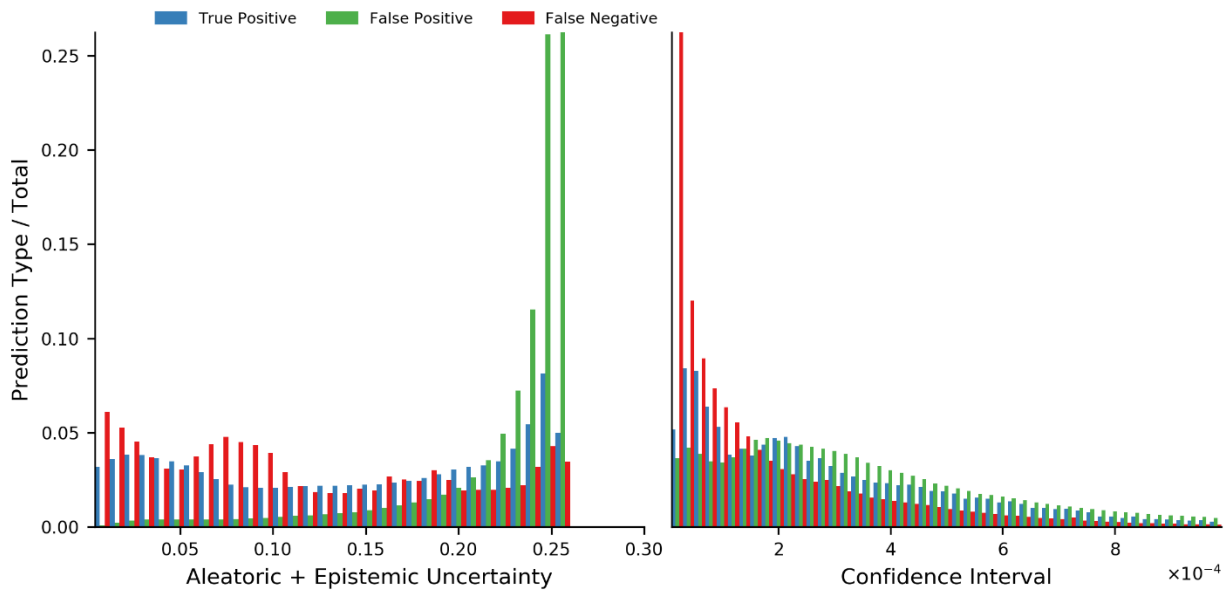

Figure 3.7. Histograms of relative prediction type binned by uncertainty of Bayesian neural network (left) and confidence interval of logistic regression (right).

In the Bayesian neural network, false positives were overwhelmingly accompanied by higher aleatoric and epistemic uncertainty, though uncertainty did not track as well with false negatives. The Bayesian neural network also had high uncertainty around true positives, meaning that when the model predicted any flooding it generally did so with a high degree of uncertainty. The uncertainty estimates from logistic regression were not as informative. While the confidence intervals did tend to be higher for false positives, they were also high for true positives and remained low for nearly all false negatives. Visualizing the uncertainty measures alongside predictions helps illustrate these differences (Figure 3.8).

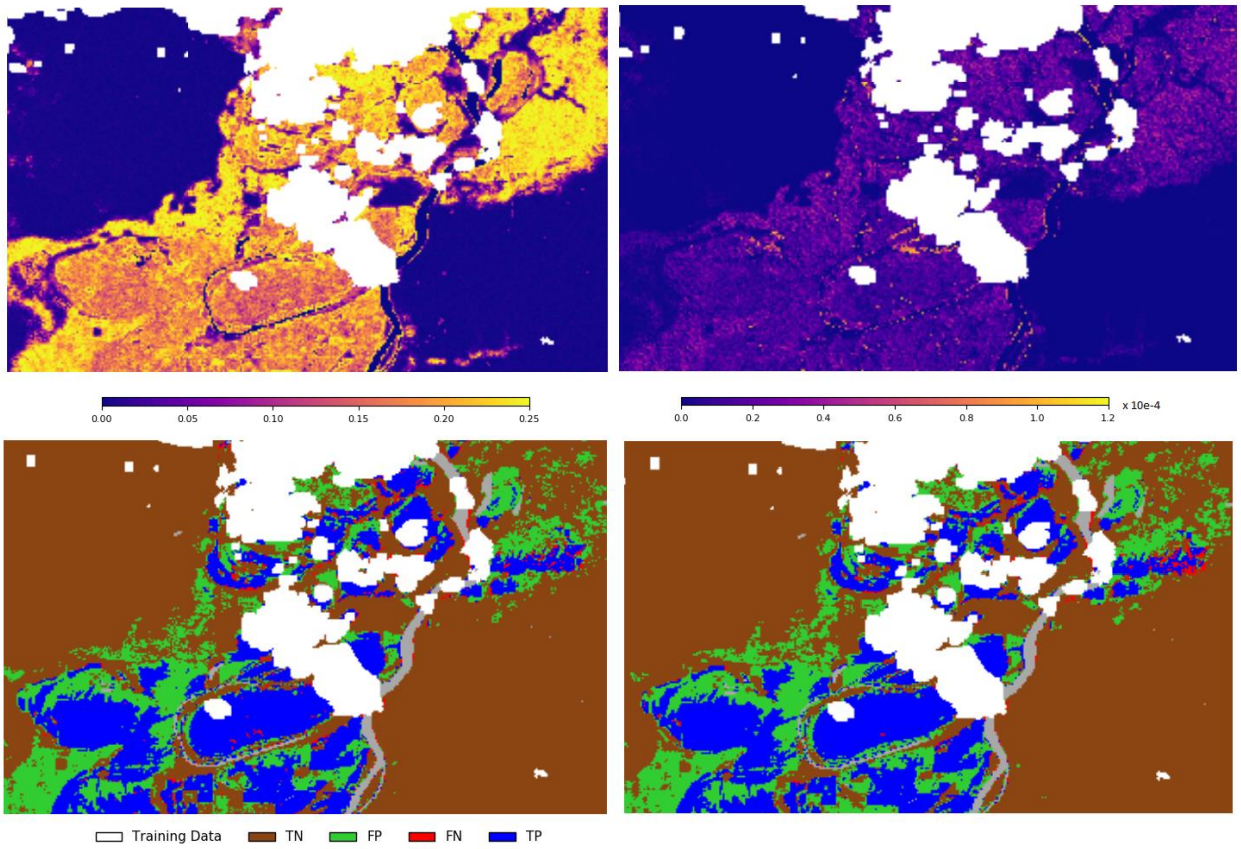

Figure 3.8. Uncertainty measures (top) and predictions (bottom) for the Bayesian neural network (left) and logistic regression (right) models in a segment of a sample image. While uncertainty is high for all predictions of flooding in the Bayesian neural network, it is highest for false positives and false negatives. The logistic regression confidence intervals were not able to discriminate error types as well.

While the Bayesian neural network displayed high uncertainty for all predicted flooding, it displayed higher relative uncertainty for false positives, in particular in the western farmland in Figure 3.8. The logistic regression confidence intervals, on the other hand, did not discriminate as finely. Interestingly, the spatial coverage of uncertainty is similar for both models, meaning at the very least they both give a rough indication of areas where predictions should be treated with skepticism.

There are a few important caveats to consider. First, the Bayesian neural network did not predict as accurately as the NN. The addition of dropout layers – which randomly removed 20% of the image pixels during training and during the Monte Carlo predictions – may have had an outsized effect on model learning due to the sparsity of flooding. Increasing the number of Monte Carlo simulations might average out this effect, though at the cost of computational time. Second, some of the false positive predictions appeared to have saturated soil which indicates previous inundation. Since a satellite image is just a snapshot in time, it is possible that those pixels may have been flooded during the same event but the water receded before other areas in the image.

The methods explored in this study demonstrate the difficulty of flood prediction using moderate resolution data. Although the flood conditioning features led to accurate predictions in the literature, they seem to become decoupled from flooding when averaged across a larger resolution. In that case, this method might yield better results with higher resolution data.

While all models could rule out the presence of flooding in less susceptible areas with high accuracy, on average they could only correctly identify <35% of flooded pixels in an image. Additionally, prediction performance varied greatly with the placement of clouds, though the underlying reasons are unclear. Despite this variability, estimating uncertainty with the Bayesian neural network could help end-users interpret predictions more critically. Uncertainty estimates are glaringly absent from most flood prediction research, and the use of a Bayesian neural network with Monte Carlo dropout could be a useful tool to bolster confidence in the results.

There are a number of future research paths that could be explored to further improve this method. As mentioned in the methodology, Convolutional neural networks may be a promising method, but there currently is no existing architecture for this particular problem. While a generalized model – that is, a model trained once on many cloud free images then used to predict flooding in other, cloudy images – showed no improvements over the method here in preliminary tests, transfer learning may be beneficial. Transfer learning is a method where some of the weights of a previously trained model are updated to accommodate new training data. One method might be to train a model on many cloud-free images then update the weights using the visible pixels of a cloud-obscured image. These weights could be given larger values to boost prediction on the cloud gaps since they occur in the same image and are spatially autocorrelated. No matter what method is pursued, the low and varying performance is unlikely to improve significantly without improvements to the underlying input features.

REFERENCES

Associated Programme on Flood Management. (2006). *Environmental Aspects of Integrated Flood Management*. Geneva, Switzerland. Retrieved from http://www.apfm.info

Brakenridge, G. R. (2019). *Dartmouth Flood Observatory*. Retrieved from http://floodobservatory.colorado.edu/Archives/index.html

Brodie, M., Weltzien, E., Altman, D., Blendon, R. J., & Benson, J. M. (2006). Experiences of Hurricane Katrina Evacuees in Houston Shelters: Implications for Future Planning. *American Journal of Public Health*, *96*(8), 1402–1408. https://doi.org/10.2105/AJPH.2005.084475

Bui, D. T., Panahi, M., Shahabi, H., Singh, V. P., Shirzadi, A., Chapi, K., et al. (2018). Novel Hybrid Evolutionary Algorithms for Spatial Prediction of Floods. *Scientific Reports*, *8*(1). https://doi.org/10.1038/s41598-018-33755-7

Centre for Research on the Epidemiology of Disasters. (2015). *The human cost of natural disasters: A global perspective*.

Chapi, K., Singh, V. P., Shirzadi, A., Shahabi, H., Bui, D. T., Pham, B. T., & Khosravi, K. (2017). A novel hybrid artificial intelligence approach for flood susceptibility assessment. *Environmental Modelling and Software*, *95*, 229–245. https://doi.org/10.1016/j.envsoft.2017.06.012

Chen, J., Zhu, X., Vogelmann, J. E., Gao, F., & Jin, S. (2011). A simple and effective method for filling gaps in Landsat ETM+ SLC-off images. *Remote Sensing of Environment*, *115*(4), 1053–1064. https://doi.org/10.1016/J.RSE.2010.12.010

Chen, M., Newell, B. H., Sun, Z., Corr, C., & Gao, W. (2019). Reconstruct missing pixels of Landsat land surface temperature product using a CNN with partial convolution. In M. E. Zelinski, T. M. Taha, J. Howe, A. A. Awwal, & K. M. Iftekharuddin (Eds.), *Applications of Machine Learning* (Vol. 11139, p. 11). SPIE. https://doi.org/10.1117/12.2529462

Choubin, B., Moradi, E., Golshan, M., Adamowski, J., Sajedi-Hosseini, F., & Mosavi, A. (2019). An ensemble prediction of flood susceptibility using multivariate discriminant analysis, classification and regression trees, and support vector machines. *Science of the Total Environment*, *651*, 2087–2096. https://doi.org/10.1016/j.scitotenv.2018.10.064

Dai, F. C., Lee, C. F., Li, J., & Xu, Z. W. (2001). *Assessment of landslide susceptibility on the natural terrain of Lantau Island, Hong Kong*. *Cases and solutions Environmental Geology* (Vol. 40). Springer-Verlag.

Donchyts, G., Schellekens, J., Winsemius, H., Eisemann, E., van de Giesen, N., Donchyts, G., et al. (2016). A 30 m Resolution Surface Water Mask Including Estimation of Positional and Thematic Differences Using Landsat 8, SRTM and OpenStreetMap: A Case Study in the Murray-Darling Basin, Australia. *Remote Sensing*, *8*(5), 386. https://doi.org/10.3390/rs8050386

Donchyts, G., Winsemius, H., Schellekens, J., Erickson, T., Gao, H., Savenije, H., & van de Giesen, N. (2016). Global 30m Height Above the Nearest Drainage. In *Proceedings of the European Geosciences Union General Assembly*. Retrieved from https://www.researchgate.net/profile/Gennadiy_Donchyts/publication/301559649_Global_30m_Height_Above_the_Nearest_Drainage/links/5719ed8f08ae30c3f9f2cc88/Global-30m-Height-Above-the-Nearest-Drainage.pdf

Dottori, F., Salamon, P., Bianchi, A., Alfieri, L., Hirpa, F. A., & Feyen, L. (2016). Development and evaluation of a framework for global flood hazard mapping. *Advances in Water Resources*, *94*, 87–102. https://doi.org/10.1016/J.ADVWATRES.2016.05.002

Enomoto, K., Sakurada, K., Wang, W., Fukui, H., Matsuoka, M., Nakamura, R., & Kawaguchi, N. (2017). Filmy Cloud Removal on Satellite Imagery with Multispectral Conditional Generative Adversarial Nets. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 1533–1541). IEEE. https://doi.org/10.1109/CVPRW.2017.197

Erwin, S. (2019). Capella Space to launch seven radar satellites in 2020 as it prepares for commercial operations. Retrieved February 27, 2020, from https://spacenews.com/capella-space-to-launch-seven-radar-satellites-in-2020-as-it-prepares-for-commercial-operations/
Feyisa, G. L., Meilby, H., Fensholt, R., & Proud, S. R. (2014). Automated Water Extraction Index: A new technique for surface water mapping using Landsat imagery. *Remote Sensing of Environment*, *140*, 23–35. https://doi.org/10.1016/J.RSE.2013.08.029
Fussell, E. (2015). The Long-Term Recovery of New Orleans' Population After Hurricane Katrina. *American Behavioral Scientist*, *59*(10), 1231–1245. https://doi.org/10.1177/0002764215591181
Gal, Y. (2016). *Uncertainty in Deep Learning*. Retrieved from http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf
Gal, Y., & Ghahramani, Z. (2015a). Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference. Retrieved from http://arxiv.org/abs/1506.02158
Gal, Y., & Ghahramani, Z. (2015b). Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. Retrieved from http://arxiv.org/abs/1506.02142
Gerber, F., de Jong, R., Schaepman, M. E., Schaepman-Strub, G., & Furrer, R. (2018). Predicting Missing Values in Spatio-Temporal Remote Sensing Data. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(5), 2841–2853. https://doi.org/10.1109/TGRS.2017.2785240
Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, *202*, 18–27. https://doi.org/10.1016/j.rse.2017.06.031
Haghizadeh, A., Siahkamari, S., Haghiabi, A. H., & Rahmati, O. (2017). Forecasting flood-prone areas using Shannon's entropy model. *J. Earth Syst. Sci*, *126*, 39. https://doi.org/10.1007/s12040-017-0819-x
Hallema, D. W., Moussa, R., Sun, G., & Mcnulty, S. G. (2016). Surface storm flow prediction on hillslopes based on topography and hydrologic connectivity. https://doi.org/10.1186/s13717-016-0057-1
Han, D. (2011). *Flood Risk Assessment and Management*. Sharjah, UNITED ARAB EMIRATES: Bentham Science Publishers. Retrieved from http://ebookcentral.proquest.com/lib/washington/detail.action?docID=864287
Hirabayashi, Y., Mahendran, R., Koirala, S., Konoshima, L., Yamazaki, D., Watanabe, S., et al. (2013). Global flood risk under climate change. *Nature Climate Change*, *3*(9), 816–821. https://doi.org/10.1038/nclimate1911
Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning, ICML 2015* (Vol. 1, pp. 448–456). International Machine Learning Society (IMLS).
Ireland, G., Volpi, M., Petropoulos, G. P., Gloaguen, R., & Thenkabail, P. S. (2015). Examining the Capability of Supervised Machine Learning Classifiers in Extracting Flooded Areas from Landsat TM Imagery: A Case Study from a Mediterranean Flood. *Remote Sensing*, *7*, 3372–3399. https://doi.org/10.3390/rs70303372
Janizadeh, S., Avand, M., Jaafari, A., Phong, T. Van, Bayat, M., Ahmadisharaf, E., et al. (2019). Prediction Success of Machine Learning Methods for Flash Flood Susceptibility Mapping in the Tafresh Watershed, Iran. *Sustainability*, *11*(19), 5426. https://doi.org/10.3390/su11195426
Jha, A. K., Bloch, R., & Lamond, J. (2012). *Cities and Flooding: A Guide to Integrated Urban Flood Risk Management for the 21st Century*. The World Bank. https://doi.org/doi:10.1596/978-0-8213-8866-2
Jin, S., Yang, L., Danielson, P., Homer, C., Fry, J., & Xian, G. (2013). A comprehensive change detection method for updating the National Land Cover Database to circa 2011. *Remote Sensing of Environment*, *132*, 159–175. https://doi.org/10.1016/j.rse.2013.01.012
Johnson, D. (2017, November). Experts See Expanding Role for Parametric Insurance, Including for U.S. Disasters. *Insurance Journal*. Retrieved from

https://www.insurancejournal.com/news/national/2017/11/22/472010.htm
Jongman, B., Ward, P. J., & Aerts, J. C. J. H. (2012). Global exposure to river and coastal flooding: Long term trends and changes. *Global Environmental Change*, *22*(4), 823–835. https://doi.org/10.1016/J.GLOENVCHA.2012.07.004
Kalimuthu, H., Tan, W. N., Sin Liang, L., & Fauzi, M. F. A. (2015). Assessing frequency ratio method for landslide susceptibility mapping in Cameron Highlands, Malaysia. In *2015 IEEE Student Conference on Research and Development, SCOReD 2015* (pp. 93–99). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/SCORED.2015.7449440
Katsuhama, Y., & Grigg, N. S. (2010). Capacity building for flood management systems: A conceptual model and case studies. *Water International*, *35*(6), 763–778. https://doi.org/10.1080/02508060.2010.533348
Kelleher, C., & McPhillips, L. (2020). Exploring the application of topographic indices in urban areas as indicators of pluvial flooding locations. *Hydrological Processes*, *34*(3), 780–794. https://doi.org/10.1002/hyp.13628
Kendall, A., & Gal, Y. (2017). What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? Retrieved from http://arxiv.org/abs/1703.04977
Kia, M. B., Pirasteh, S., Pradhan, B., Mahmud, A. R., Sulaiman, W. N. A., & Moradi, A. (2012). An artificial neural network model for flood simulation using GIS: Johor River Basin, Malaysia. *Environmental Earth Sciences*, *67*(1), 251–264. https://doi.org/10.1007/s12665-011-1504-z
Kingma, D. P., & Lei Ba, J. (n.d.). Adam: A Method for Stochastic Optimization. Retrieved from https://arxiv.org/pdf/1412.6980.pdf
Kwon, Y., Won, J.-H., Joon Kim, B., & Cho Paik, M. (2018). *Uncertainty quantification using Bayesian neural networks in classification: Application to ischemic stroke lesion segmentation*.
Lawal, D. U., Matori, A. N., Yusuf, K. W., Hashim, A. M., & Balogun, A. L. (2014). Analysis of the flood extent extraction model and the natural flood influencing factors: A GIS-based and remote sensing analysis. *IOP Conference Series: Earth and Environmental Science*, *18*, 012059. https://doi.org/10.1088/1755-1315/18/1/012059
Loucks, D. P. (2019). Environmental Research Letters Developed river deltas: are they sustainable? Developed river deltas: are they sustainable? *Environ. Res. Lett*, *14*, 113004. https://doi.org/10.1088/1748-9326/ab4165
Ma, M., Liu, C., Zhao, G., Xie, H., Jia, P., Wang, D., et al. (2019). Flash Flood Risk Analysis Based on Machine Learning Techniques in the Yunnan Province, China. *Remote Sensing*, *11*(2), 170. https://doi.org/10.3390/rs11020170
McGhee, D. J., Binder, S. B., & Albright, E. A. (2020). First, Do No Harm: Evaluating the Vulnerability Reduction of Post-Disaster Home Buyout Programs. *Natural Hazards Review*, *21*(1), 05019002. https://doi.org/10.1061/(ASCE)NH.1527-6996.0000337
Mojaddadi, H., Pradhan, B., Nampak, H., Ahmad, N., & Ghazali, A. H. bin. (2017). Ensemble machine-learning-based geospatial approach for flood risk assessment using multi-sensor remote-sensing data and GIS. *Geomatics, Natural Hazards and Risk*, *8*(2), 1080–1102. https://doi.org/10.1080/19475705.2017.1294113
Mosavi, A., Ozturk, P., Chau, K., Mosavi, A., Ozturk, P., & Chau, K. (2018). Flood Prediction Using Machine Learning Models: Literature Review. *Water*, *10*(11), 1536. https://doi.org/10.3390/w10111536
Nair, V., & Hinton, G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. In *International Conference on Machine Learning* (pp. 807–814).
Neumann, B., Vafeidis, A. T., Zimmermann, J., & Nicholls, R. J. (2015). Future Coastal Population Growth and Exposure to Sea-Level Rise and Coastal Flooding - A Global Assessment. *PLOS ONE*, *10*(3), e0118571. https://doi.org/10.1371/journal.pone.0118571
Ng, M. K.-P., Yuan, Q., Yan, L., & Sun, J. (2017). An Adaptive Weighted Tensor Completion Method for the Recovery of Remote Sensing Images With Missing Data. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(6), 3367–3381.

https://doi.org/10.1109/TGRS.2017.2670021
Nobre, A. D., Cuartas, L. A., Hodnett, M., Rennó, C. D., Rodrigues, G., Silveira, A., et al. (2011). Height Above the Nearest Drainage – a hydrologically relevant new terrain model. *Journal of Hydrology*, *404*(1–2), 13–29. https://doi.org/10.1016/J.JHYDROL.2011.03.051
Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66. https://doi.org/10.1109/TSMC.1979.4310076
Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.
Pekel, J.-F., Cottam, A., Gorelick, N., & Belward, A. S. (2016). High-resolution mapping of global surface water and its long-term changes. *Nature*, *540*(7633), 418–422. https://doi.org/10.1038/nature20584
Perlin, K. (2002). Improving noise. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques - SIGGRAPH '02* (Vol. 21, p. 681). New York, New York, USA: ACM Press. https://doi.org/10.1145/566570.566636
Pourghasemi, H. R., Gayen, A., Edalat, M., Zarafshar, M., & Tiefenbacher, J. P. (2019). Is multi-hazard mapping effective in assessing natural hazards and integrated watershed management? *Geoscience Frontiers*. https://doi.org/10.1016/j.gsf.2019.10.008
Rothmann, D. (2019). Is your algorithm confident enough? Retrieved June 5, 2019, from https://towardsdatascience.com/is-your-algorithm-confident-enough-1b20dfe2db08
Schmidt, M., Roux, N. Le, & Bach, F. (2013). Minimizing Finite Sums with the Stochastic Average Gradient. *Mathematical Programming*, *162*(1–2), 83–112. Retrieved from http://arxiv.org/abs/1309.2388
Schumann, Guy J-P., & Bates, P. D. (2018). The Need for a High-Accuracy, Open-Access Global DEM. *Frontiers in Earth Science*, *6*, 225. https://doi.org/10.3389/feart.2018.00225
Schumann, Guy J.-P., Bates, P. D., Neal, J. C., & Andreadis, K. M. (2014). Technology: Fight floods on a global scale. *Nature*, *507*(7491), 169–169. https://doi.org/10.1038/507169e
Shen, H., Li, X., Cheng, Q., Zeng, C., Yang, G., Li, H., & Zhang, L. (2015). Missing Information Reconstruction of Remote Sensing Data: A Technical Review. *IEEE Geoscience and Remote Sensing Magazine*, *3*(3), 61–85. https://doi.org/10.1109/MGRS.2015.2441912
Smardon, R., Felleman, J., Shannon, S., Giacobbe, C., Wesley, J., & Mcshane, J. (1996). *Protecting Floodplain Resources: A Guidebook for Communities*.
Solomatine, D., See, L. M., & Abrahart, R. J. (2008). Data-Driven Modelling: Concepts, Approaches and Experiences. In *Practical Hydroinformatics* (pp. 17–30). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-79881-1_2
Tehrany, M. S., Pradhan, B., & Jebur, M. N. (2013). Spatial prediction of flood susceptible areas using rule based decision tree (DT) and a novel ensemble bivariate and multivariate statistical models in GIS. *Journal of Hydrology*, *504*, 69–79. https://doi.org/10.1016/J.JHYDROL.2013.09.034
Tehrany, M. S., Pradhan, B., & Jebur, M. N. (2014). Flood susceptibility mapping using a novel ensemble weights-of-evidence and support vector machine models in GIS. *Journal of Hydrology*, *512*, 332–343. https://doi.org/10.1016/J.JHYDROL.2014.03.008
Tehrany, M. S., Shabani, F., Jebur, M. N., Hong, H., Chen, W., & Xie, X. (2017). GIS-based spatial prediction of flood prone areas using standalone frequency ratio, logistic regression, weight of evidence and their ensemble techniques. *Geomatics, Natural Hazards and Risk*, *8*(2), 1538–1561. https://doi.org/10.1080/19475705.2017.1362038
Vassileva, M., Giulio Tonolo, F., Riccardi, P., Lecci, D., Boccardo, P., & Chiesa, G. (2017). Satellite SAR interferometric techniques in support to emergency mapping. *European Journal of Remote Sensing*, *50*(1), 464–477. https://doi.org/10.1080/22797254.2017.1360155
Wang, G., Li, W., Aertsen, M., Deprest, J., Ourselin, S., & Vercauteren, T. (2019). Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. *Neurocomputing*, *338*, 34–45.

https://doi.org/10.1016/j.neucom.2019.01.103
Wickham, J., Stehman, S. V., Gass, L., Dewitz, J. A., Sorenson, D. G., Granneman, B. J., et al. (2017). Thematic accuracy assessment of the 2011 National Land Cover Database (NLCD). *Remote Sensing of Environment*, *191*, 328–341. https://doi.org/10.1016/j.rse.2016.12.026
WMO-UNESCO Joint-Task Team. (2007). *International Flood Initiative*. Retrieved from http://www.ifi-home.info/IFI_Concept_Paper.pdf
Woznicki, S. A., Baynes, J., Panlasigui, S., Mehaffey, M., & Neale, A. (2019). Development of a spatially complete floodplain map of the conterminous United States using random forest. *Science of The Total Environment*, *647*, 942–953. https://doi.org/10.1016/J.SCITOTENV.2018.07.353
Wu, P., Yin, Z., Yang, H., Wu, Y., & Ma, X. (2019). Reconstructing geostationary satellite land surface temperature imagery based on a multiscale feature connected convolutional neural network. *Remote Sensing*, *11*(3). https://doi.org/10.3390/rs11030300
Xu, H. (2006). Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery. *International Journal of Remote Sensing*, *27*(14), 3025–3033. https://doi.org/10.1080/01431160600589179
Zhang, G., Feng, G., Li, X., Xie, C., & Pi, X. (2017). Flood Effect on Groundwater Recharge on a Typical Silt Loam Soil. *Water*, *9*(7), 523. https://doi.org/10.3390/w9070523
Zhang, Q., Yuan, Q., Zeng, C., Li, X., & Wei, Y. (2018). Missing Data Reconstruction in Remote Sensing image with a Unified Spatial-Temporal-Spectral Deep Convolutional Neural Network. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(8), 4274–4288. https://doi.org/10.1109/TGRS.2018.2810208

APPENDIX A

MODEL SPECIFICATIONS

Logistic regression:

Training and prediction carried out with the Python package *scikit-learn*. Coefficients of the logistic regression models were found by minimizing their negative log-likelihood using a stochastic average gradient solver. This solver finds the global minima in a similar way to stochastic gradient descent, but incorporates previous gradient values to achieve convergence more quickly (Schmidt et al., 2013)

Random forest:

Training and prediction carried out with the Python package *scikit-learn*. The Python package *scikit-optimize* was used to tune Hyperparameters. The parameter optimized the number of estimators over the space [2, 200], the maximum tree depth over [2, 3000], and the maximum number of leaf nodes over [2, 1000]. To find the optimal parameters, models were repeatedly trained through five-fold cross-validation with the objective of maximizing the balanced accuracy. Rather than measure the absolute number of correct predictions, balanced accuracy uses the proportion of correct predictions relative to each class size, making it useful for highly imbalanced classes.

Neural network:

Training and prediction carried out with the using the Keras API in in the TensorFlow Python library. The neural network architecture was chosen after preliminary tests, adjusting number of

neurons, number of layers, regularization, activation functions. The neural network used here contained two dense layers (with 24 and 12 neurons, respectively), each followed ReLU activation function, and then batch normalization, and finally a softmax activation function yields the prediction. The model objective was to minimize the sparse categorical cross-entropy using the Adam algorithm (Kingma & Lei Ba, n.d.).

APPENDIX B

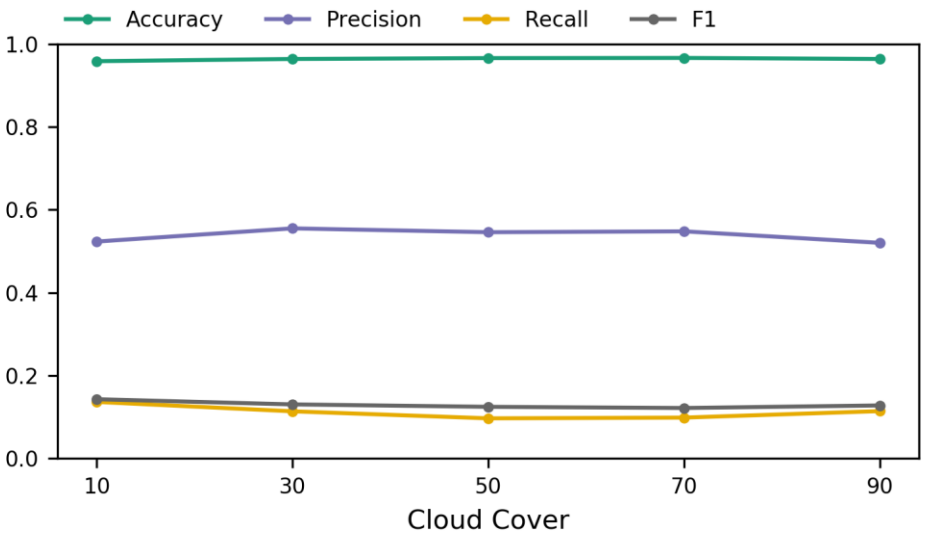

Figure 1A: Performance metrics vs. cloud cover using only flood pixels in training set and evaluation.

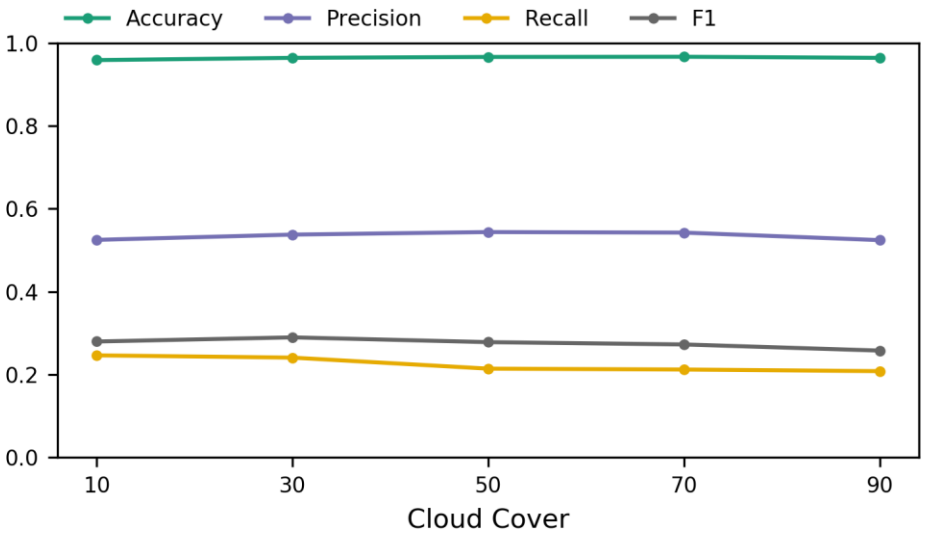

Figure 2A: Performance metrics vs. cloud cover using all detected water pixels in training set and only flood pixels during evaluation.

Sampling Regime	Cloud cover	Accuracy	Precision	Recall	F1	AUC
A	30%	0.670	0.08	0.480	0.057	0.631
B	30%	0.661	0.079	0.484	0.056	0.628
C	30%	0.647	0.078	0.486	0.053	0.625

Table 1A: Mean performance metrics at 30% cloud cover of different training data sampling regimes. Training sets consist of: (A) All pixels within 150m buffer around flood water; equal number of random pixels from outside buffer; (B) All pixels within 150m buffer around detected water (including permanent); equal number of random pixels from outside buffer; (C) All pixels within 150m buffer around detected water (including permanent); a number of random pixels from outside buffer equal to # flood pixels x 4

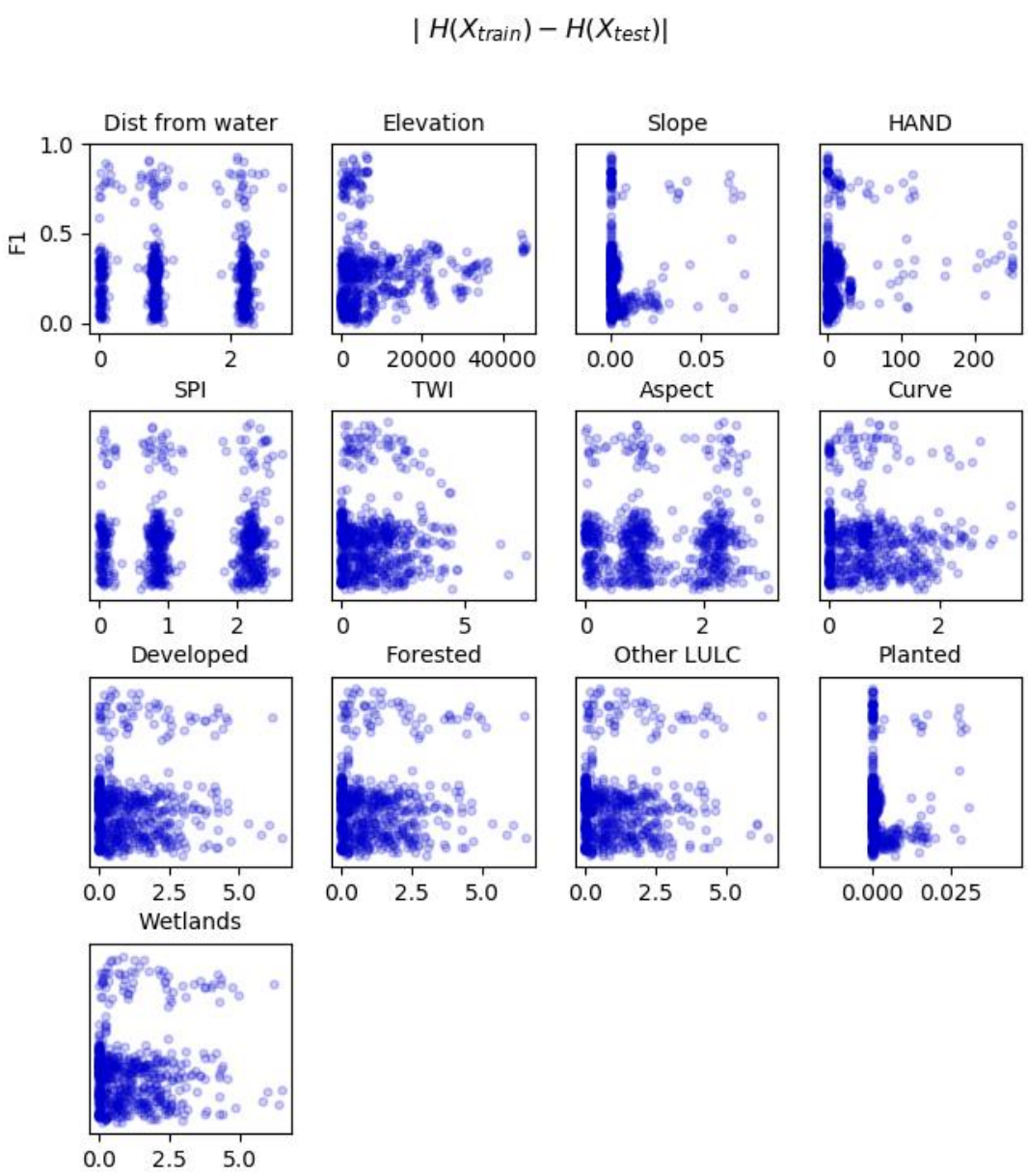

Figure 3A: scores vs. absolute difference in entropies of train and prediction datasets in random cloud trials. Entropy is the amount of information contained in a dataset, with an entropy of 1 meaning the data are disordered and contain a high-level of discriminative information.

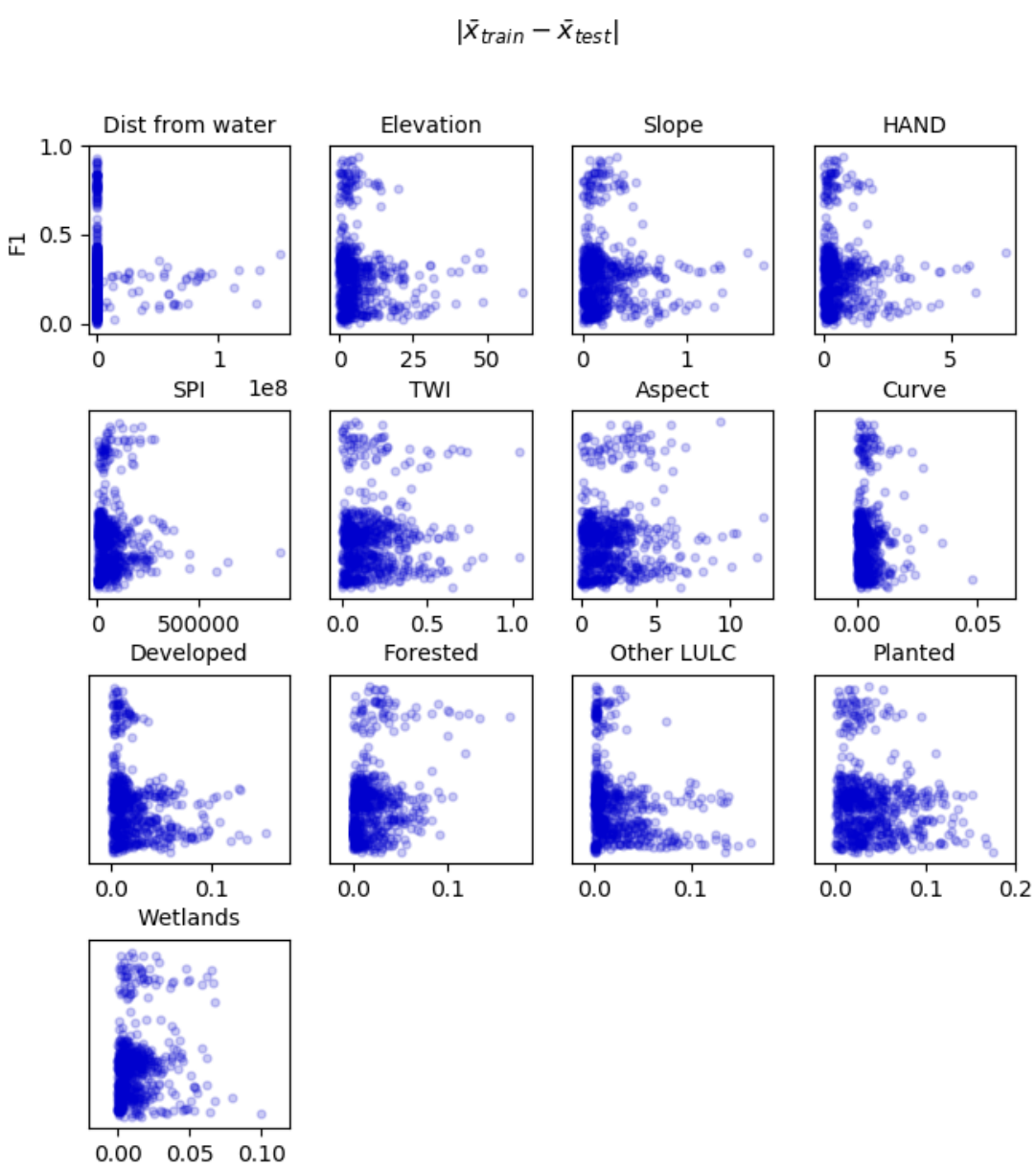

Figure 4A: F1 scores vs. absolute difference in means of train and prediction datasets in random cloud trials.

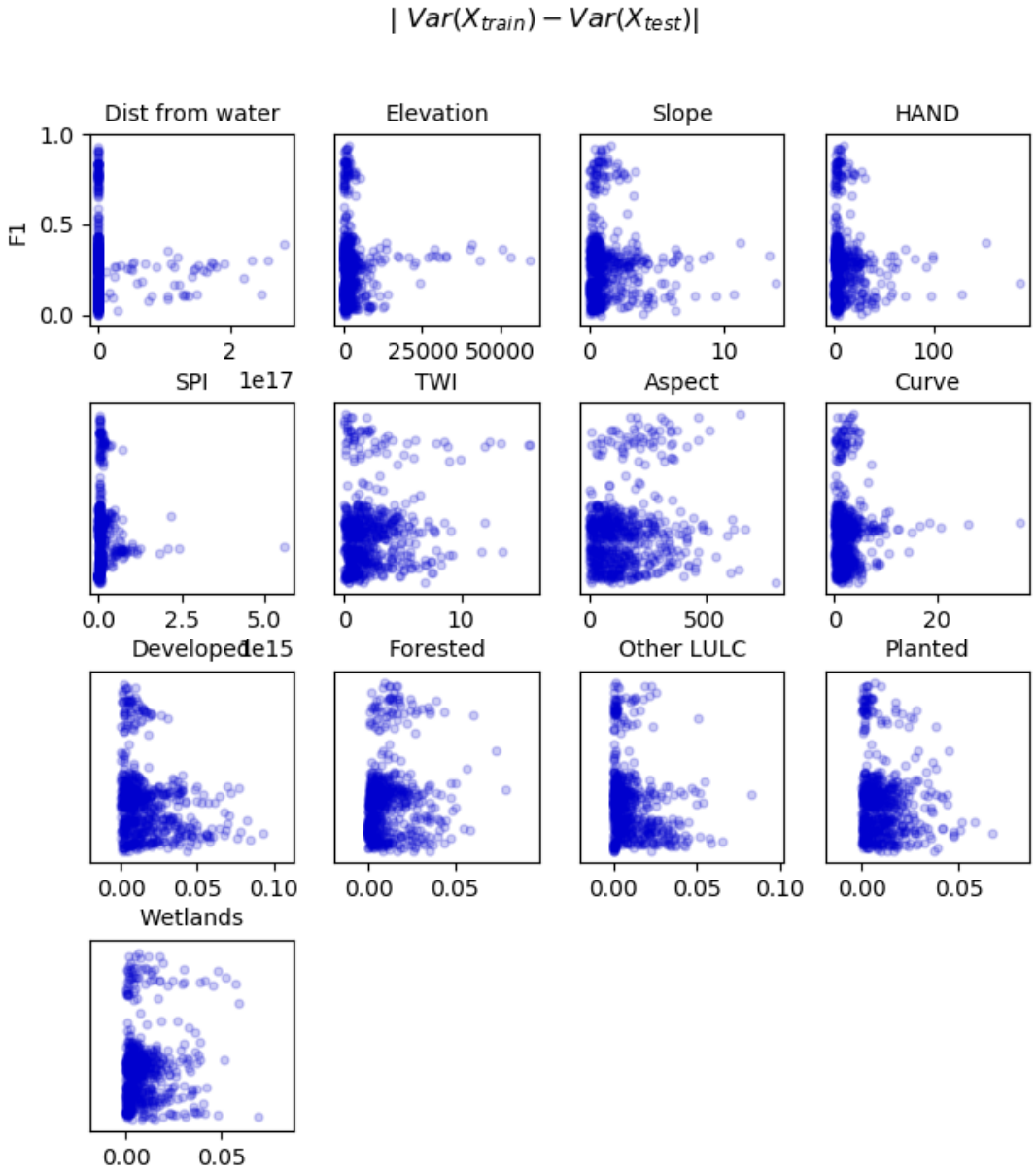

Figure 5A: F1 scores vs. absolute difference in variances of train and prediction datasets in random cloud trials.

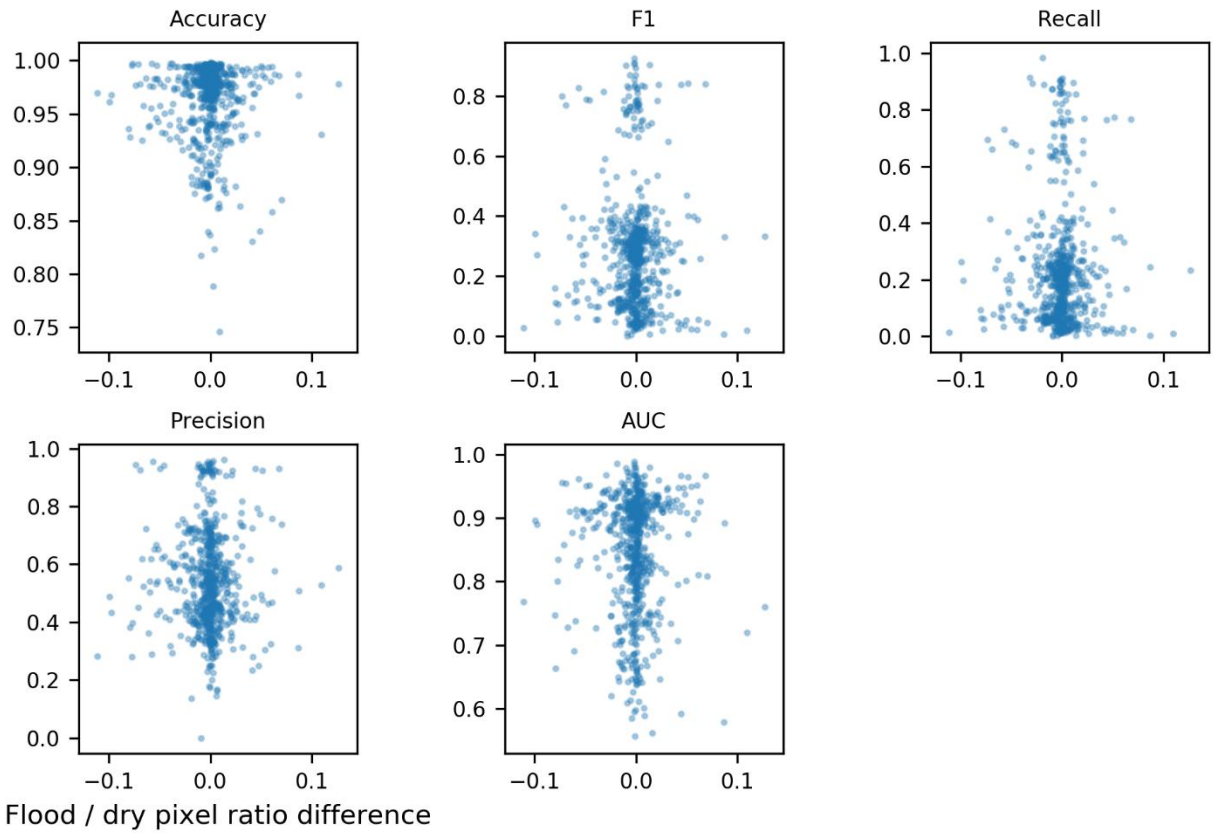

Figure 6A: Difference between flood dry ratio in cloud-free and cloudy datasets for each image.