BUREAU OF RECLAMATION

Technical Memorandum No. ENV-2022-59

Investigating Methods for Stochastic Flood Model Hydrograph Extraction

Dam Safety Technology Development Program

U.S. Department of the Interior **February 2022**

Mission Statements

The U.S. Department of the Interior protects and manages the Nation's natural resources and cultural heritage; provides scientific and other information about those resources; honors its trust responsibilities or special commitments to American Indians, Alaska Natives, and affiliated Island Communities.

The mission of the Bureau of Reclamation is to manage, develop, and protect water and related resources in an environmentally and economically sound manner in the interest of the American public.

Cover Photo – An aerial view of Shasta Dam and Lake (Bureau of Reclamation).

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
February 2022	Final	10/2020 – 03/2022

4. TITLE AND SUBTITLE
Technical Memorandum No. ENV-2022-59
Investigating Methods for Stochastic Flood Model Hydrograph Extraction
Dam Safety Technology Development Program

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Elise Madonna, emadonna@usbr.gov
Drew Allan Loney, dloney@usbr.gov, (303)445-2541
Amanda Stone, astone@usbr.gov, (303)445-2282

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Bureau of Reclamation, Technical Service Center
Water Resources Engineering & Management Support Group
Building 67, Mail Code: 86-68210
Denver, Colorado

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

13. SUPPLEMENTARY NOTE

14. ABSTRACT
The current project developed an automated hydrograph classification workflow using a two-stage classification procedure, first a self-organizing map (SOM) machine learning (ML) method followed by mean shift clustering. The SOM method groups hydrographs by evaluating their similarity in the shape and magnitude. The SOM groups are further refined with the mean shift clustering operation to yield a small number of hydrograph clusters that are representative of the range of behavior at a site. The developed ML workflow is an automated process with minimal user input that runs rapidly and scales to the number of hydrographs produced by stochastic rainfall/runoff models.

15. SUBJECT TERMS
Machine learning, hydrograph, time series, stochastic, model

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Amanda Stone; Drew Allan Loney
a. REPORT	**b. ABSTRACT**	**a. THIS PAGE**			**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U			(303) 445-2282; (303) 445-2541

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Technical Memorandum No. ENV-2022-59

Investigating Methods for Stochastic Flood Model Hydrograph Extraction

Dam Safety Technology Development Program

Prepared by:

Bureau of Reclamation
Technical Service Center
Denver, Colorado

U.S. Department of the Interior **February 2022**

Technical Memorandum No. ENV-2022-59

Investigating Methods for Stochastic Flood Model Hydrograph Extraction

Dam Safety Technology Development Program

ELISE MADONNA (Affiliate)
Digitally signed by ELISE MADONNA (Affiliate)
Date: 2022.02.22 09:14:00 -07'00'

Prepared by: Elise Madonna
Intern, Water Resources Engineering & Management Support Group, 86-68210

DREW LONEY
Digitally signed by DREW LONEY
Date: 2022.02.22 11:24:57 -07'00'

Prepared by: Drew Allan Loney, PhD, PE
Civil Engineer (Hydrologic), Water Resources Engineering & Management Support Group, 86-68210

AMANDA STONE
Digitally signed by AMANDA STONE
Date: 2022.02.22 12:17:30 -07'00'

Prepared by: Amanda Stone, PE
Civil Engineer (Hydrologic), Water Resources Engineering & Management Support Group, 86-68210

KATHLEEN HOLMAN
Digitally signed by KATHLEEN HOLMAN
Date: 2022.02.22 12:24:18 -07'00'

Peer reviewed by: Kathleen Holman PhD
Meteorologist, Water Resources Engineering & Management Support Group, 86-68210

Acronyms and Abbreviations

AUC	Area Under the Curve
CAS	Corrective Active Studies
DOI	U.S. Department of the Interior
IE	Issue Evaluation
ML	Machine Learning
NN	Neural Networks
Reclamation	Bureau of Reclamation
SEFM	Stochastic Event Flood Model
SOM	Self Organizing Map
TSC	Technical Service Center
USGS	United States Geological Survey

Contents

Page

Figures

Appendices

Executive Summary

Stochastic rainfall/runoff models are at the forefront of hydrologic modeling state-of-practice. These models have increasingly been used by the TSC to estimate flood magnitudes and associated return periods, along with uncertainty, for detailed flood hazard studies such as issue evaluations (IEs) and corrective action studies (CASs). Stochastic rainfall/runoff models simulate many thousands of potential flood realizations across frequency space to estimate probabilistic floods and support risk analyses. The resulting products of these studies are typically flood frequency curves representing peaks, volumes, and water surface elevations produced from a large number of modeled hydrographs. One challenge that arises with these large datasets comes when hydrographs must be used for additional analyses beyond determination of existing hydrologic loads, such as design, modification, or operational changes. In these scenarios, working with a smaller number of hydrographs becomes necessary. The process of selecting a subset of hydrographs is currently manual, time consuming, and dependent upon the judgement of the person tasked with selection.

The current project developed an automated hydrograph classification workflow using a two-stage classification procedure, first a self-organizing map (SOM) machine learning (ML) method followed by mean shift clustering. The SOM method groups hydrographs by evaluating their similarity in the shape and magnitude. The SOM groups are further refined with the mean shift clustering operation to yield a small number of hydrograph clusters that are representative of the range of behavior at a site. The developed ML workflow is an automated process with minimal user input that runs rapidly and scales to the number of hydrographs produced by stochastic rainfall/runoff models. The ML hydrograph classification workflow was tested across multiple gage and model instances. In each of these cases, the ML workflow was robust and produced a hydrograph classification that is representative of the site.

1.0 Introduction

Stochastic rainfall/runoff models are considered a best practice in hydrologic modeling and have been increasingly used for dam safety decision-making (Bureau of Reclamation, 2003b; Sorooshian et al., 2008). Rainfall-runoff modeling has been used extensively to assess flood magnitudes and associated return periods at facilities, but incorporation of stochastic methodologies is more recent (since the late 1990s at Reclamation). Stochastic methods in rainfall runoff modeling treat inputs and parameters as variables rather than set values. This allows for numerous combinations of input values to simulate some of the variability that is inherent in a hydrologic response. Stochastic rainfall runoff methods create a large set of "realizations" of potential floods in a basin, but also introduce the complexity of a much larger set of hydrographs with which to work for decision-making. The purpose of this study is to develop a methodology that Reclamation can apply to produce a smaller subset of representative hydrograph realizations and simplify the decision-making process.

Rainfall/runoff models combine precipitation-frequency estimates with a hydrologic model to evaluate runoff timing and magnitude (Beven, 2012; Devia et al., 2015; Sitterson et al., 2018). In contrast to streamflow measurements that are fixed to the precipitation events within the historical record, rainfall/runoff methods can be used for historical reanalysis as well as to explore behavior outside of the historical record (Dutta et al., 2012; Moore et al., 2001; W. Wang et al., 2021). Such exploration can include changing the precipitation magnitude, precipitation timing, antecedent basin conditions, or water management decisions. A hydrologic model used for a rainfall/runoff process would be calibrated/validated over the historical record and then driven with the changed conditions to understand the response of the basin. The flexibility, accuracy, and utility of rainfall/runoff methods have made them a key component across Reclamation water management and dam safety activities (Bureau of Reclamation, 2016, 2019).

When combined with a stochastic framework, a rainfall/runoff model can be used to estimate the likelihood, magnitude, and timing of flows with statistical rigor (Bureau of Reclamation, 2003b; Marco et al., 1993; Sorooshian et al., 2008). Probabilities are formulated for rainfall magnitude, rainfall distribution, and various antecedent conditions, including soil moisture and channel conditions, prior to application in the stochastic model (Tabari, 2019). These distributions are repeatedly sampled to create initial and boundary conditions for the rainfall/runoff model which is solved to obtain a hydrograph realization. By sampling the probability distributions many thousands of times, a flow-frequency or volume-frequency curve can be estimated with appropriate statistical assumptions. However, a peak streamflow or volume frequency curve distills the response of the basin to a single scalar representing the model solution. As useful and important to Reclamation are the hydrograph realizations resulting from the stochastic runs that give the flow as a function of time during the events.

Hydrograph shape and magnitude are used to understand hydrologic events and categorize the basin response. The flow magnitude, the number of peaks, and the spacing between peaks characterize the interactions of physical processes that govern the hydrology of a basin (Singh, 1997). A hydrograph is also utilized in flood operations and routing (Federal Emergency Management Agency, 2020). The shape and magnitude of a hydrograph determines how a

facility operator may release water for emergency response. Flood routings use hydrographs to determine inundation depths and durations across a facility (Bureau of Reclamation, 2003a). Having a set of hydrograph realizations that is representative of the range of possible basin behavior is particularly important for these latter cases to understand and plan for scenarios at a facility.

As the number of hydrographs increases with the number of stochastic runs, it becomes increasingly challenging to understand and use the hydrograph realizations. Current Reclamation hydrograph classification methods call for a hydrologist to review and manually group the hydrographs. Facing several thousand hydrograph realizations, a manual process is time consuming. Facing several hundred thousand hydrographs, a manual process is unrealistic. Hydrograph routing to estimate inundation is also a time-consuming process that is only feasible for a limited number of hydrographs. While classifying hydrograph realizations could reduce the stochastic hydrograph set to a representative sample, current manual hydrograph classification is not feasible to enable such a reduction. A new classification tool is therefore necessary to support hydrograph classification for both improving hydrologic understanding and flood routings.

This report summarizes a work funded by the Reclamation Dam Safety Office to develop automated hydrograph classification using machine learning (ML). The remaining sections of the report provide an overview of previous hydrologic classification efforts as well as the ML methods. The methodology is described, and example classifications are presented for observed and modeled hydrograph sets. Appendices are included to provide more detailed results, as well as detailed instructions for using the developed classification workflow.

2.0 Background

2.1 Basin Classification

Basins are often classified based on their hydrology, meteorology, geology, or other factors that provide a means to structure the understanding of basin properties and responses (Potyondy & Geier, 2011; U.S. Environmental Protection Agency, 2013). The resulting classifications are useful, to various degrees, in highlighting similarities in basin behavior (Khan et al., 2001; McManamay & DeRolph, 2019). Classifying basins in this way informs decision-making and provides a framework for how each basin should be treated. However, classifying basins accurately can be challenging due to the need for significant amounts of data (Khan et al., 2001; McManamay et al., 2014; Olden et al., 2012). In some Reclamation basins, hydro-climatic data does not exist and may not be obtainable remotely. Missing data must be manually collected by a hydrologist, which becomes time consuming when repeated for many different basins. A common question is which properties should be prioritized to categorize the basins into a number of reasonably sized, meaningful groups (Khan et al., 2001; McCuen, 1973; Vasudevan et al., 2021). Classification is further complicated in that individual basin properties can vary significantly depending on temporal effects associated with different seasons, climate change, and human activities that alter land cover and river flow (Fowler et al., 2018; LaFontaine et al.,

2015; Pianosi & Wagener, 2016). A basin classification methodology should be evaluated in the context of the needs, the time period of the classification, the classification parameters, and whether the classification is appropriate for the intended use.

The development of computational and machine learning algorithms to identify patterns has increased the ability to classify basins quantitively (Choubin et al., 2017; Olden et al., 2012). Clustering algorithms, self-organizing maps (SOMs), and neural networks (NN) have most commonly been used in basin classification (Land & Water Australia, 2009). A variety of unsupervised clustering algorithms are available, such as the common K-Means method (Bandyopadhyay & Saha, 2013). These methods find groups based on various representative metrics, typically a vectorized distance, and either aggregate or disaggregate data into the classification groups. Some clustering algorithms may require the number of groups be specified as an a priori argument (Land & Water Australia, 2009). Alternatively, SOMs have been implemented as another unsupervised classification method, though traditionally still require the user to manually specify the number of groups (Kohonen, 2001). SOMs operate to classify datapoints by projecting datapoints onto a lower dimensional output map according to the data similarity (Land & Water Australia, 2009). NNs have also been used in some preliminary studies to begin examining how deep learning can help to discover patterns between basins (Sit et al., 2020). Use of NNs is more challenging as the method is supervised, requiring some initial classified data as the basis to train the network. This has the potential to introduce bias into the NN and may not be robust to out-of-set conditions if not accounted for during training.

Each of the basin classification methods described above uses a variety of data as an input (Archfield et al., 2014; Kuentz et al., 2017; Monk et al., 2007; Mosley, 1981; Nathan & McMahon, 1990; Ouellet Dallaire et al., 2019; Schmitt et al., 2007). Raw streamflow data can give a model access to flow characteristics such as magnitude, frequency, duration, and rate of change (Land & Water Australia, 2009). Classification methods also have additional inputs which, together with the streamflow, can significantly alter the classification results depending on the parameters that are used and how they are preprocessed. Classification methods that use a vectorized distance metric are particularly sensitive to changing parameterization and preprocessing scaling. Because these model parameters have such a significant effect on classification results, it is necessary to systematically test and compare results from many different parameter values and scaling techniques to determine which values should be used prior to using the classification (Peñas et al., 2014; Snelder & J. Booker, 2013). Different model parameters may also be chosen based on whether the user desires coarse or fine classification of basins, with coarse classification resulting in fewer classes each containing a larger variety of basins, and fine classification resulting in a large number of classes, each containing only a few basins.

2.2 Hydrograph Classification

Unlike basin classification that determines similarity across non-coincident catchments, hydrograph classification determines the similarity among hydrographs within a single basin. Classification of hydrographs can be done to understand hydrologic regimes within an individual

basin, such as snowmelt, thunderstorms, extreme events, and drought (Brunner et al., 2018; Zaerpour et al., 2021). Hydrograph classification can also highlight differences within the behavior of a single hydrologic regime, such as the number of peaks within a flood event (Ternynck et al., 2016). While an intra-basin analysis eliminates many parameters necessary to understand the differences between basins, additional parameters may be necessary to correctly classify the hydrographs (Snelder & J. Booker, 2013). Understanding the hydrologic regimes present in a basin gives water managers a tool to plan facility response under various conditions. Additionally, hydrograph classification can reduce a large set of hydrographs to a representative subset that still fully describes the range of basin behavior (Land & Water Australia, 2009). This is particularly helpful for subsequent analysis tasks, such as flood routing, which are feasible for only a limited number of hydrographs.

Hydrograph classification approaches are structured based on the use case of the final classifications. Computational and machine learning algorithm implementations can be separated into approaches that use either solely scalar values or time series combined with scalar information (Hancer et al., 2020). If hydrograph shape information is not required, a hydrograph may be summarized by scalar values, such as the volume, peak magnitude, and number of peaks. This scalar information can be analyzed using similar methods as for basin classification (Land & Water Australia, 2009). However, there are many cases where time series shape information is important to the classification, for example a peak streamflow event in either an increasing or decreasing streamflow condition. In these cases, use of the full hydrograph timeseries can produce a more refined classification (Maharaj et al., 2019). The full hydrograph timeseries is used as the input vector to the classification, and any supplemental scalar information can be appended to the vector. Additional care in preprocessing must be taken to ensure that scaling (i.e., normalization) is consistent across the hydrograph timeseries as well as between the hydrograph timeseries and scalar values to prevent artificial bias and to maintain the distribution of the parameters.

Hydrograph classification using the full time series information benefits from classification methods that can handle high dimensional data (X. Wang et al., 2005). When including the full hydrograph timeseries, each additional measurement adds to the dimensionality of the dataset. For example, a 7-day hourly hydrograph has a dimensionality of 168 before including any additional scalar information. While the performance of some clustering algorithms can scale to high dimensional data, the distance classification metrics of common clustering algorithms are typically not sufficient to describe the variability of high dimensional data (Steinbach et al., 2004). Other ML methods, such as SOMs and NNs, which build additional steps into the classification process, can offer better performance through more advanced classification metrics (Maimon & Rokach, 2005). Numerous hydrograph classification methods have been developed with a variety of approaches, looking at an assortment of application cases (Hannah et al., 2000; Harris et al., 2000; Matsumoto & Miyamoto, 2018; Merritt et al., 2021; Poff & Ward, 1989; Sauquet et al., 2021; Siddiqui et al., 2021; Wohlfart et al., 2016).

3.0 Methods

3.1 Classification Process Overview

The developed hydrologic classification workflow is detailed in subsequent sections. However, an overview of the process is provided here to frame the discussion and to provide context for more detailed methodology descriptions. The developed workflow uses full hydrograph time series with additional scalar values included on the vectors representing the hydrograph realizations.

SOM Step

1. Obtain and format input streamflow timeseries data
2. Preprocess input data to remove missing values and section into hydrograph realizations
3. Calculate additional scalar parameters (number of peaks, peak magnitude) for each hydrograph realization
4. Scale additional scalar parameters to be centered around the mean of the hydrograph realizations
5. Apply SOM to the vectors representing the hydrograph realizations
6. Output plots of hydrographs and cell weight vector from each SOM node

Mean Shift Clustering Step

7. Calculate additional scalar parameters (area under the curve, number of peaks, and peak magnitude) from each SOM node weight vector
8. Scale SOM node scalar parameters individually to each be between zero and one, with the highest value from each parameter equaling one
9. Apply the mean shift clustering algorithm on the SOM node weight vectors
10. Output plots of hydrographs and cluster weight for each cluster, as well as distribution for mean node weight, volume, highest peak, number of peaks parameters in each cluster
11. Write results to an Excel workbook, summarizing hydrograph cluster assignment and cluster properties

3.2 Classification Data Sources

The workflow was constructed using four different test datasets, three observed hydrograph timeseries and one set of modeled timeseries from a study using the Stochastic Event Flood Model (SEFM). Although the target use case is focused on modeled timeseries, the observed series were utilized to ensure generality of the process across data sources as well as hydrograph magnitudes and shapes. The latter two features can be readily explored by selecting observation stations with different characteristics. Results for each dataset listed below are discussed individually within Section 4.

1. USGS 15 minute streamflow data for the Colorado River (U.S. Geological Survey, 2016b)

   This dataset is an example of moderate to high flows along a major river system.

2. USGS 15 minute streamflow data for the South Platte River (U.S. Geological Survey, 2016a)

   This dataset is an example of low to moderate flows through a medium river system.

3. Boise State University Data (Boise State University, 2021)

   This hourly streamflow dataset is from the Dry Creek Experimental Watershed Basin in Idaho and is an example of low flow in a small basin.

4. SEFM Data for El Vado Lake (Bureau of Reclamation, 2016)

   The simulated dataset used in testing was a set of hourly hydrographs generated by the SEFM for the El Vado Lake in New Mexico for 15-day event durations.

3.3 SOM

The SOM ML classification algorithm was selected as it is well suited to the project requirements. SOMs are particularly useful for sorting hydrograph realizations into representative categories because they can classify large amounts of non-linear, high-dimensional data into a lower dimension (Holman, 2018; Kohonen, 2013). The ability of the method to capture details in high dimensional data was relevant due to the importance of

hydrograph shape in the classification application cases. Additionally, SOMs generate a more sophisticated output layout than other clustering methods by creating an output node placement map that is indicative of similar nodes (Holman, 2018; Kohonen, 1982).

SOMs have been in use since the 1980s, when they were first proposed by Teuvo Kohonen (Miljkovic, 2017). SOMs function as single layer NNs that use an unsupervised learning algorithm, meaning that the algorithm discovers patterns in input data for which the correct categorization is unknown a priori by the user (Kohonen, 2001). Since their initial publication, SOMs have been broadly applied and adapted across a variety of use cases (Barreto, 2007; Kalteh et al., 2008; Kohonen et al., 1996).

3.3.1 SOM Procedure

SOMs create a two-dimensional *m* by *n* gridded output map where *m* and *n* are dimensions that have been specified by the user (Kohonen, 2001). Each grid node contains a weight vector representing the hydrograph realization vectors that are most similar to it (Kohonen, 2013). The node weight vectors are initialized with random values and then trained by comparing every vector from the input hydrograph realizations one at a time to the SOM weight vectors until a steady state is achieved or until the user-specified number of training iterations has been reached. Each input realization vector is assigned to a node by calculating the Euclidean distance between it and every SOM node weight vector, then choosing the node that minimizes that distance. When a realization vector is assigned to a node, the weight vectors of the node and its neighbors get updated to become more similar to the assigned realization vector. The weight vector is therefore representative of the data assigned to it at the end of the SOM training (Lin & Wang, 2006). The nodes are organized throughout the output map in a way that reflects their similarity (Kohonen, 2013). The most similar realization vectors should be contained in nodes that are adjacent in the output map.

Figure 3.1 illustrates the process through which SOMs take in input data vectors, compute the similarity metric among all nodes, and then rearranges the output nodes according to similarity on a two-dimensional output map. Each node contained in the output map is represented with a weight vector generated by the SOM that has the same dimensionality as the input realization vectors.

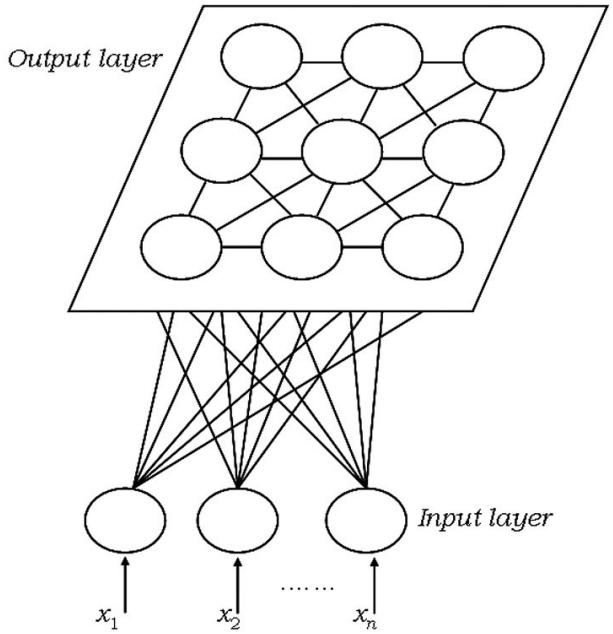


Figure 3.1.—Illustration of a SOMs transformation of input vectors to a two dimensional output layer map, where output position indicates the similarity of vectors or nodes (Milos Gajdos, 2017).

3.3.2 SOM Preprocessing Scaling

While SOMs tend to be robust across different datatypes, the input data must be formatted and normalized due to the utilization of a Euclidian distance metric during training. If an input variable has a significantly larger magnitude than the others, the larger magnitude variable will have more influence on the SOM. It is therefore necessary that all data being input to the SOM is in the same units and of similar magnitude.

The SOM classification skill was increased when the hydrograph realization vector included two additional scalar parameters – the number of peaks and largest peak value in each hydrograph – in addition to hydrograph timeseries. Both of these parameters are calculated from the hydrograph realization prior to training the SOM and are appended to the hydrograph to form a vector representing the realization.

The peak streamflow in each hydrograph realization is calculated by taking a maximum of the hydrograph timeseries. The maximum is then normalized by first dividing by the mean streamflow of the hydrograph realization and then multiplying by the mean streamflow of all hydrograph realizations. This places the peak value on the same scale as the mean of the input data while preserving the distribution of the peak values. Including this variable improved the ability of the SOM to partition the hydrograph realizations based on magnitude.

The number of peaks in each hydrograph is calculated in reference to the 85th percentile of the hydrograph realization. The 85th percentile is subtracted from the hydrograph realization, and the number of sign changes in the resulting series is used to count the number of peaks. Referencing the number of peaks to the 85th percentile suppresses small peaks from the count that are less relevant to the classification. The number of peaks is normalized for each hydrograph realization by dividing by the maximum number of peaks in any hydrograph realization and then multiplying by the overall mean streamflow of all hydrograph realizations. This again places the number of peaks on the same scale as the mean of the input data while preserving the count distribution. Including this variable improved the ability of the SOM to discriminate the shape of the hydrographs.

3.3.3 Size of SOM Space and Training Iterations

A 25 by 25 grid was set as the default SOM dimensionality, giving 625 possible nodes to utilize in the classification. The SOM dimensions were determined through a sensitivity analysis on the distance map evaluated across the test datasets. The distance map represents the normalized sum of the Euclidian distance between the nodal weight vectors of a node and its neighbors. The neighborhood function used to relate nodes was taken as a Gaussian distribution. Changes in distance between nodal weight vectors indicate how well resolved and trained the SOM classification is based on grid size and training iterations. As one increases the number of nodes available in the grid, the SOM algorithm will reach a size at which additional nodes no longer affect the classification. When using an unsupervised learning algorithm without any a priori knowledge of the correct classification structure, over resolving the SOM allows the algorithm to utilize an arbitrary number of classification groups without artificially forcing the classification.

Using 625 nodes ensures that, regardless of the input hydrograph realizations, the SOM algorithm will not be constrained by the number of available nodes. Figures 3.2 and 3.3 show the effect of SOM dimensions on two test datasets. This size allows the SOM enough nodes in the output to fully capture representative characteristics of the hydrographs, while also limiting the number of empty cells present in the output. Although this size results in more categories than are likely possible as meaningful hydrologic groups, the number of SOM groups is further refined in later stages of the classification process. Additionally, this resolution simplifies the use of the classification process by eliminating need for the user to specify SOM dimensions that best fit each new dataset.

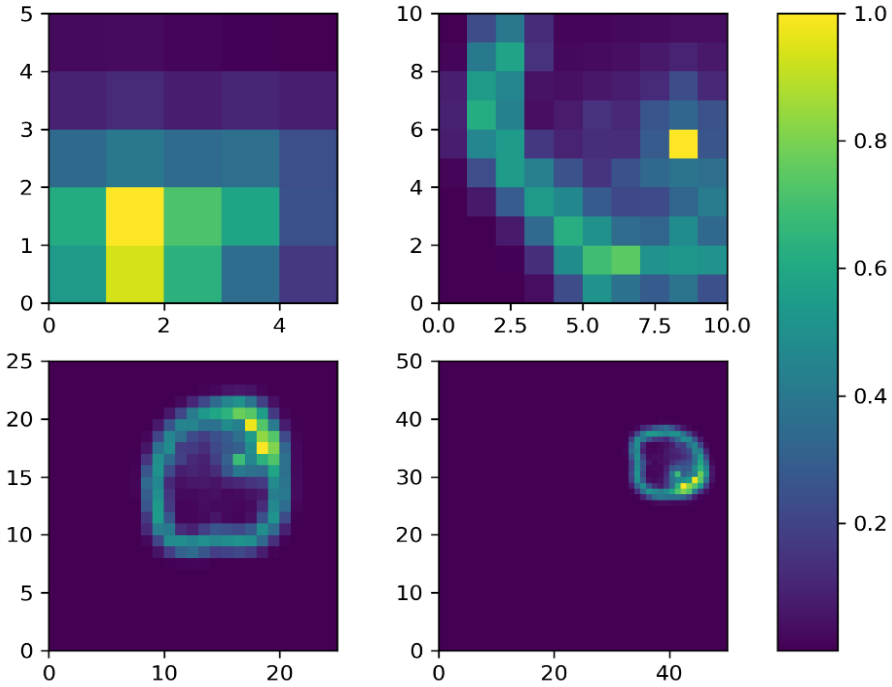


Figure 3.2.—Visualizations of distance maps for SOM dimensions of 5 x 5, 10 x 10, 25 x 25, and 50 x 50 from left to right for El Vado SEFM data.

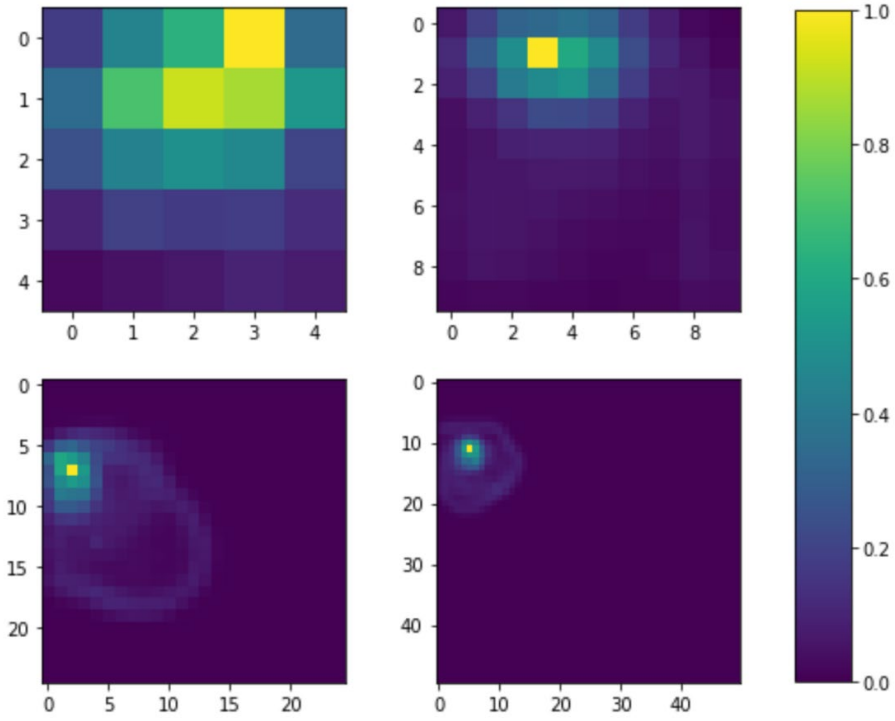


Figure 3.3.—Visualizations of distance maps for SOM dimensions of 5 x 5, 10 x 10, 25 x 25, and 50 x 50 from left to right for Colorado River gage data.

Specifying the number of training iterations is also important when constructing the SOM. With each training iteration, the model goes through every hydrograph realization, calculating the closest node by distance and updating the nodal weight vectors. The number of training iterations dictates the total number of times the algorithm loops through the hydrograph realizations to update the nodes. Increasing the number of iterations improves the classification accuracy of the SOM but also requires additional compute time. These two considerations are balanced by setting a tolerance threshold (a percent difference of 0.1%) between the results of the current and previous training iterations, and then testing how many iterations are necessary to be within the threshold. It was determined that using 2,500 training iterations is sufficient to meet the tolerance threshold using the test datasets. This number of iterations is therefore used as the default value in the classification process to train SOMs, but it may be updated by the user if it proves insufficient for a specific case.

3.3.4 SOM Limitations

While SOMs have proven to be powerful tools for classifications of this type, there are important limitations to the use of the method and interpretation of its output. First among these are resolution and accuracy. The dimensionality of the SOM is proportional to the resolution of the SOM. Because the number of output nodes is specified as an input to the algorithm, care must be taken not to generate a SOM that is either too small to encompass the entirety of variability among the input data or so large that the majority of SOM nodes do not contain any realization vectors (Céréghino & Park, 2009). Finding this balance is a common challenge in creating ML models; creating a model with a resolution that is too low results in “underfitting” the input data, while a model with a resolution that is too high “overfits” the input data. An underfitted SOM has too few nodes, and because there are so many hydrograph realizations within the nodes, these models will not be flexible enough to accurately capture characteristics in the input data assigned to them. In contrast, an overfitted SOM has too many nodes, which leads to models that are too flexible and are therefore influenced too much by small differences and background noise in the input data. An overfitted model will most likely have many nodes that each only represent one or two input vectors, while an underfitted model will lump the majority of input vectors into several nodes.

The key objective is to choose a number of SOM nodes that results in a model which allows good representation of the input data and is not overly skewed by background noise. It can be difficult to make this decision because the dimensionality of the SOM must be specified prior to analysis. To determine the SOM dimensions that result in the desired resolution, the user may need to use a trial-and-error method and run the SOM multiple times with different dimensions, as done in Figures 3.2 and 3.3. However, determining the correct SOM resolution for an accurate classification remains challenging, particularly for individuals without a data analytics background. This classification process therefore purposefully overfits the SOM map to reduce the problem from thousands of hydrograph realizations to less than a thousand SOM weight vectors that are representative of the grouped hydrographs. This is then used by a second clustering operation, outlined subsequently, to further reduce the number of groupings. This

developed two-step workflow is thought to be more robust across different input datasets and minimizes user input. Additionally, it provides for a large reduction in the data dimensionality, which improves the skill of clustering algorithms.

Furthermore, as the SOM is unsupervised, standard methods of comparing training and testing error cannot be used to assess performance (Palacio-Niño & Berzal, 2019). Cross validation techniques are often used to assess supervised models (Kumar, 2020). However, unsupervised methods lack an a priori classification that can be used to measure classification skill (Bandyopadhyay & Saha, 2013). The performance of the SOMs can be assessed visually by generating plots for each SOM node and observing how well each node weight vector represents characteristics of the assigned the hydrographs. Figure 3.4 shows an example of how the SOM node weight (shown as a thick line on top of the others) represents the hydrographs contained within the node (with each individual hydrograph shown as a thin line) for the USGS Colorado River dataset (U.S. Geological Survey, 2016b). Plots of this type were used to assess the quality of the SOM hydrograph classification when individually applied to each test dataset.

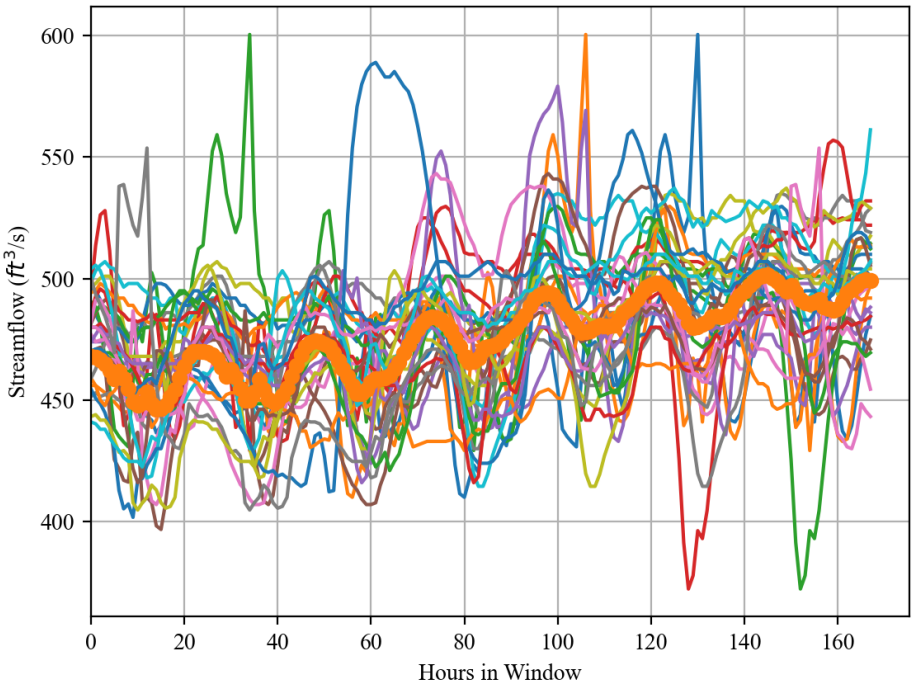


Figure 3.4.—An example of the hydrograph plots used to assess quality of the SOM's fit to the data.
The individual hydrographs are shown as thin lines, and the weight vector of the SOM is shown as a thicker line over the top of the hydrographs. Data is taken from the USGS Colorado River test dataset (U.S. Geological Survey, 2016b).

3.4 Clustering

A clustering algorithm is used subsequent to the SOM classification to further consolidate the SOM groups into a smaller, more hydrologically meaningful number of categories. This process follows the same fundamental concept as hierarchical clustering, which uses multiple clustering algorithms sequentially to achieve an optimal classification. A significant feature of hierarchical clustering is that clusters are assumed to follow an ordered network structure; in other words, each successive round of clusters is assumed to uniquely contain the previous classes nested within the new classes rather than completely reclassifying the input data (Gordon, 1987; Rao & Srinivas, 2008). This concept saves time by allowing each clustering iteration to build on the previous iterations without recalculating any of the previous clustering. It also facilitates identification of similarities among the classes of the previous iteration and can reduce the data dimensionality within the analysis to improve the clustering skill (Ayesha et al., 2020; Song et al., 2013). Hierarchical clustering is also useful when different features of the data need to be emphasized in different portions of the clustering analysis.

3.4.1 Mean Shift Clustering

Mean shift clustering, another unsupervised ML algorithm, operates by using kernel density and centroid estimation to evaluate similarity. This clustering method represents the parameter space as continuous and iteratively shifts each datapoint until it is as close as possible to the mean of the nearest kernel density estimation surface peak that represents a cluster (Comaniciu & Meer, 2002; Pedregosa, F. et al., 2021). Between iterations, the kernel density surface is updated, and the centroids of the clusters are recalculated. The algorithm is converged when the difference in the centroids of the clusters is within a tolerance between iterations.

Multiple clustering algorithms were tested as options for further refinement of the SOM classification. Mean shift clustering was chosen over the other methods because it is not necessary to specify a clustering depth or number of clusters prior to the analysis. Mean shift clustering models also allow the user to specify and adjust a radial bandwidth function which represents the datapoints to indirectly influence the number of clusters. However, many mean shift clustering implementations also provide a means to estimate a bandwidth using various heuristics. Because the clustering model is unsupervised, traditional methods of comparing training and validation error cannot be used to assess model performance. Therefore, clustering performance is assessed visually by generating plots and metrics for each cluster and observing how distinctly each cluster represents hydrograph characteristics.

Figures 3.5 and 3.6 illustrate how the kernel bandwidth affects cluster estimation. In the first figure, when a bandwidth value of two is used, the peaks in the kernel density estimation surface encompass a wider range of datapoints, and each datapoint ends up being sorted into three major clusters. However, in Figure 3.6, when a bandwidth value of 0.8 is chosen, the peaks in the kernel density estimation surface are more selective and not all the points belong to a large cluster in the final results. An advantage of mean shift clustering is that the number of clusters is not a priori specified and is instead indirectly adjusted by altering the bandwidth parameter. This

allows the algorithm to automatically estimate the number of clusters appropriate to represent the input data which can help prevent underfitting/overfitting from grouping into a prespecified number of clusters.

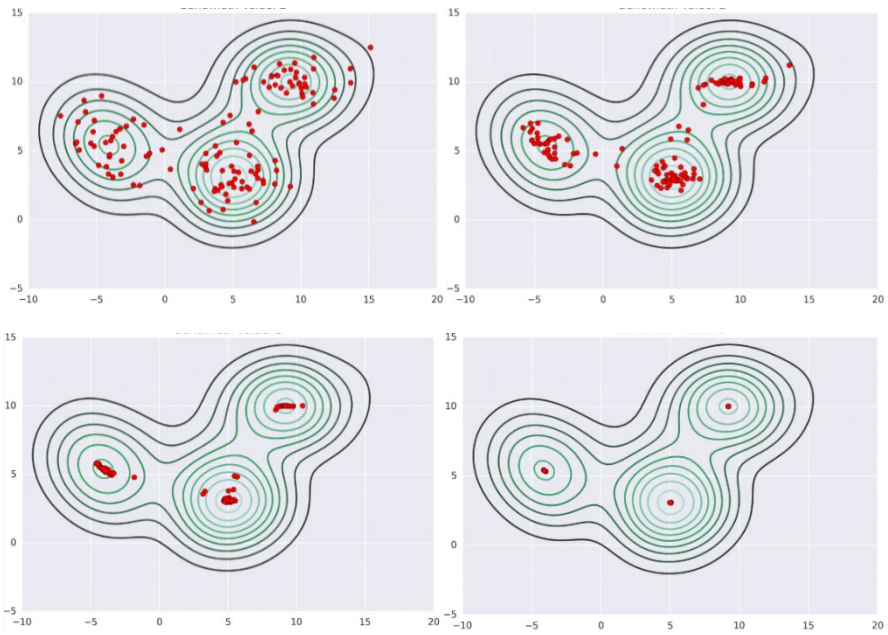


Figure 3.5.—A two-dimensional projection of the mean shift parameter space, showing algorithm iterations with a kernel bandwidth parameter of two.
The red dots are the items to be classified. Contours give the final kernel density surface from summing the item kernels. Iterations proceed from the top left to the bottom right. (Yugesh Verma, 2021).

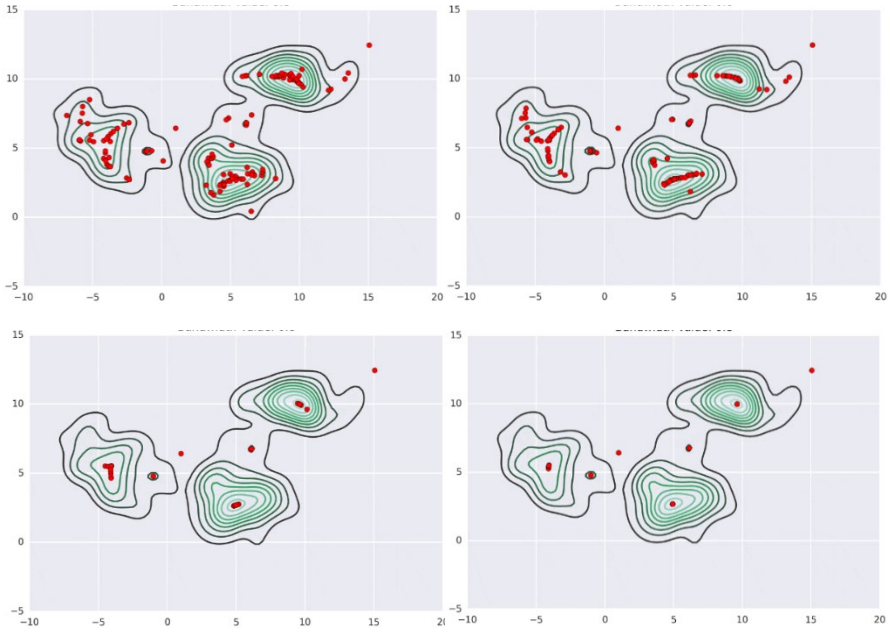


Figure 3.6.—A two-dimensional projection of the mean shift parameter space, showing algorithm iterations with a kernel bandwidth parameter of 0.8.
The red dots are the items to be classified. Contours give the final kernel density surface from summing the item kernels. Iterations proceed from the top left to the bottom right. (Yugesh Verma, 2021).

3.4.2 Clustering Parameters and Scaling

The mean shift clustering algorithm utilizes the output of the SOM classification to reduce the dimensionality of its input data and as an initial classification. Unlike the SOM, which uses the hydrograph timeseries in addition to scalar parameters, the mean shift clustering algorithm is not given the hydrograph timeseries as an input and uses only scalar parameters. The mean shift clustering algorithm incorporates the distance map from the SOM and, from the nodal weight vectors, the area under the curve (AUC), the number of peaks, and the largest peak magnitude. The weight vectors of each SOM node are utilized rather than the hydrographs directly because, through the SOM training process, the weight vectors are updated to reflect the hydrograph realizations that have been assigned to the SOM node. The behavior of many input hydrograph realizations can therefore be described by a single nodal weight vector. Additionally, the SOM distance map contains information about the hydrograph shape as similar features are placed more closely on the map. Use of the distance map in this manner reduces the dimensionality of including the hydrograph shape from a high dimensional problem based on the number of measurements in the timeseries to a single scalar that can serve as a proxy for the same information.

The distance map is output by the trained SOM and includes a value between zero and one for every SOM node that indicates its proximity to the surrounding cells. The nodal distances are a final training product that represent the similarity of the nodes in the SOM algorithm map. The distance map information is included in the clustering as a mechanism to transfer the shape and magnitude information from the SOM to the mean shift algorithm. Use of the SOM distance map with the clustering provides the primary information for the clustering analysis.

AUC for the nodal weight vector is calculated by using trapezoidal integration. The AUC is analogous to the volume of the hydrograph realizations being represented by the weight vector and was introduced to the clustering algorithm to better distinguish the integrated magnitude of the nodal weight vector.

Finally, the number of nodal weight vector peaks and largest magnitudes peaks are calculated similar to the SOM inputs. These parameters are retained in the mean shift clustering analysis to improve discrimination of the hydrograph realization shape. While the distance map gives the relative overall similarity between two nodal weight vectors, it does not provide specific information about the shape of the nodal weight vector. These two additional scalar parameters improve the ability of the mean shift algorithm to discriminate hydrograph shape when different shapes are closely spaced in the SOM distance map.

In contrast to the calculated parameters which are input to the SOM, the mean shift clustering parameters are not scaled to the mean of the hydrographs. Instead, a standard scalar is used to normalize each parameter from zero to one, with the maximum value in the individual parameter being set equal to one. Given the dimensionality of the SOM, clustering will have at most 625 inputs. In practice however, the number of inputs from the SOM are much lower. Among the four datasets tested, the SOM categorizes input hydrographs into a minimum of 46 nodes and a maximum of 214 nodes.

3.4.3 Clustering Sensitivity Tests

The sensitivity of the clustering output to the bandwidth was examined for two test datasets. An initial bandwidth is estimated as the average maximum distance among randomly ordered groups of datapoints estimated using k-Nearest Neighbor (kNN) (Pedregosa, F. et al., 2021). The kNN searches for 30% the number of datapoints as neighbors and uses a Euclidean distance metric. Because a large number of randomized groupings are used to obtain an average distance, the sensitivity to kNN initialization is minimized.

This maintained a constant initial bandwidth for each dataset to which a bandwidth correction was applied. The correction factor is defined as:

$$\beta' = \frac{\beta}{k}$$

where β is the initial bandwidth, k is a constant correction factor, and β' is the updated bandwidth estimate. Larger correction factors therefore produce a smaller bandwidth and will tend to increase the number of clusters produced by the mean shift algorithm. Sensitivity to the bandwidth was determined by keeping a constant initial bandwidth and progressively changing values of the correction factor.

Figure 3.7 shows that for both the SEFM dataset and the USGS Colorado River dataset, decreasing the bandwidth results in a larger number of clusters. This follows intuition as the mean shift algorithm contracts its search area with smaller bandwidth, which will in turn cause the algorithm to identify more clusters in order to classify the input data.

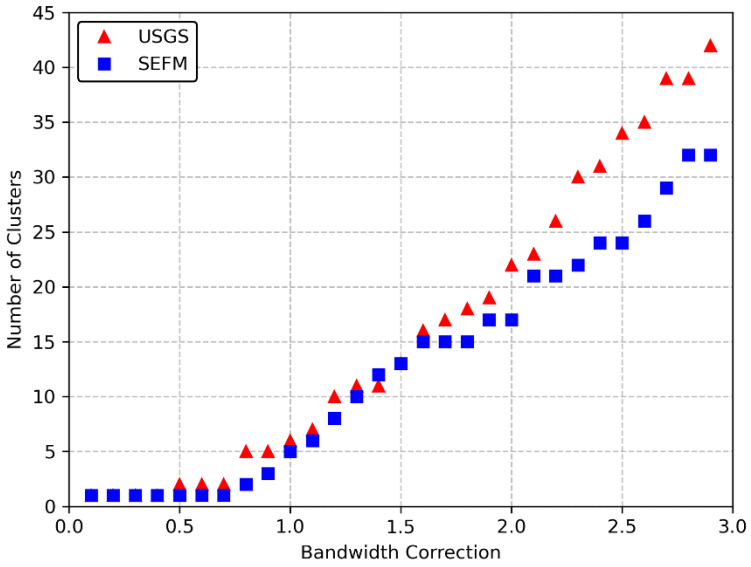


Figure 3.7.—Sensitivity test results for the effect of the bandwidth correction factor on the final number of clusters. The effects of changing the correction value were tested on the SEFM El Vado data and the USGS Colorado River data.

A default bandwidth correction of 1.25 is used in the mean shift clustering algorithm because it resulted in a moderate number of clusters for all examined datasets. Additionally, upon visual inspection of the clustering, differences in hydrograph characteristics could be identified among the clusters. Figures 3.8 and 3.9 demonstrate how changing the bandwidth correction from 1.25 to 1.8, respectively, results in over twice the number of clusters. The plots show every hydrograph realization contained in and the average weight vector for each cluster, which is calculated by averaging the weight vectors from every SOM node included in each cluster. Although the clusters generated using a bandwidth correction of 1.8 are able to better represent hydrograph characteristics, it was judged that the additional clusters were overfit and represented only small changes within the hydrograph realizations.

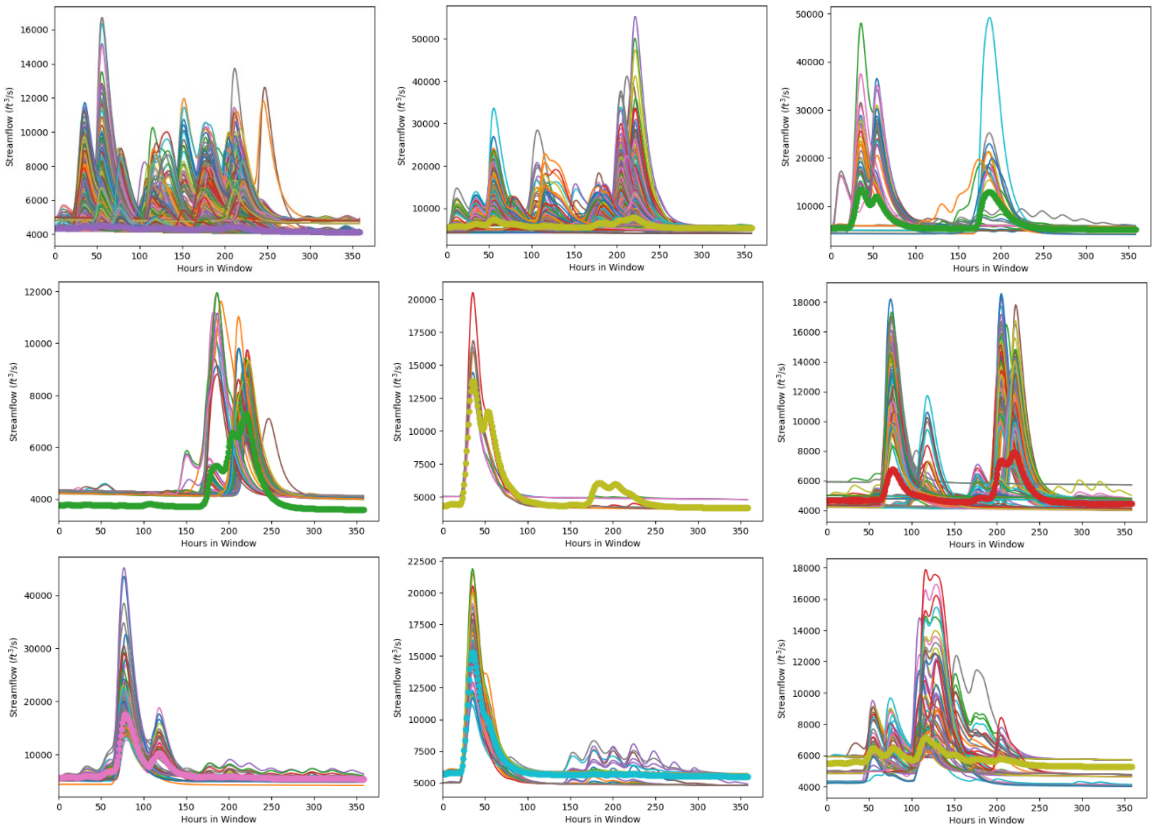


Figure 3.8.—Cluster plots generated for the SEFM data using a bandwidth correction of 1.25.

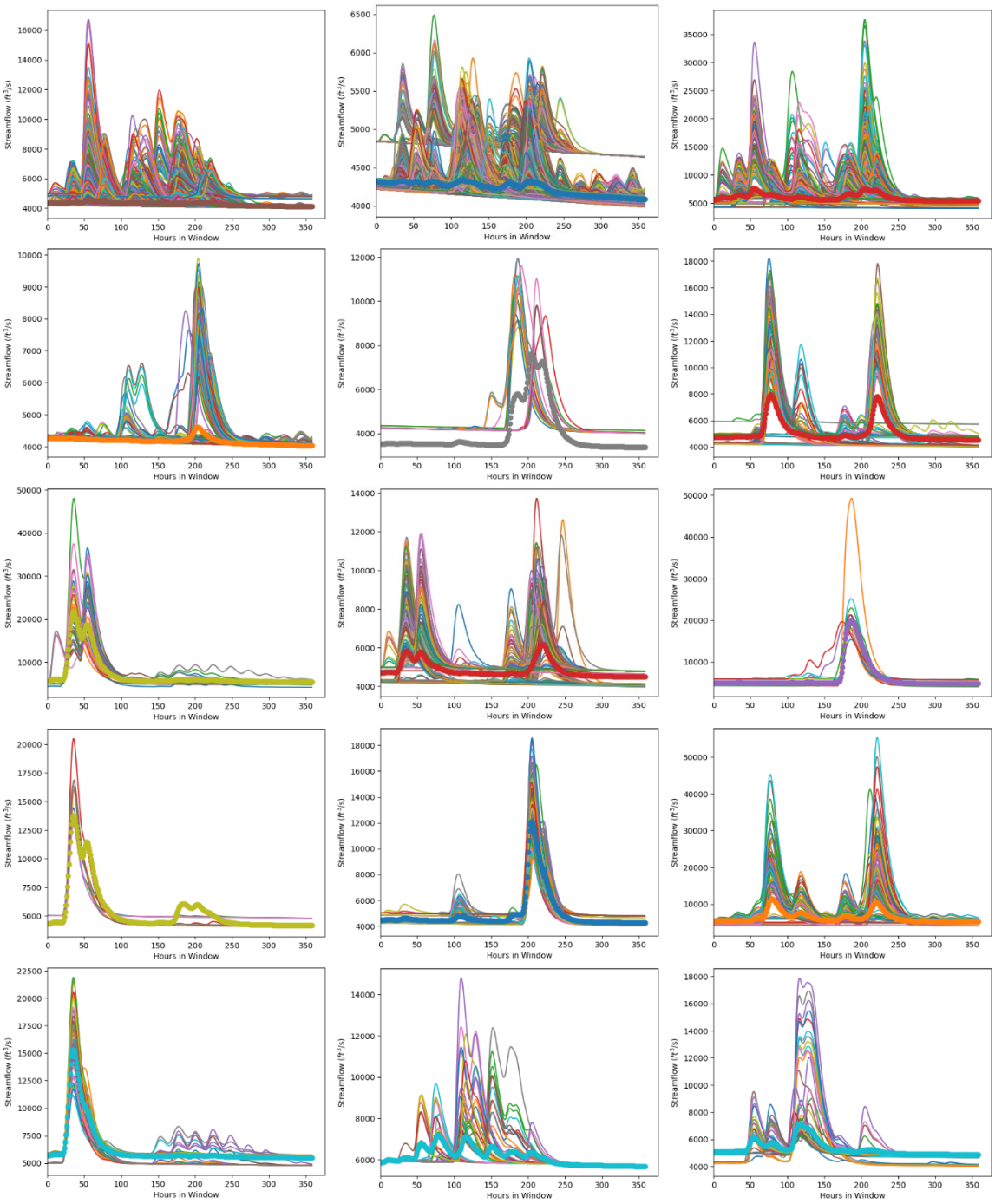


Figure 3.9.—Cluster plots generated for SEFM data using a bandwidth correction of 1.8.

While it is anticipated that the initial bandwidth and correction factor will perform well for most input datasets, review of the output data remains necessary to ensure that the bandwidth is correct for a particular analysis. If the initial bandwidth is either too reductive or aggressive, the bandwidth correction should be modified to improve the clustering algorithm skill.

3.5 Platform Choice Discussion

R and Python each have relative benefits as development environments used to create and train SOM and clustering algorithms. There are three major SOM packages available in R and three available in Python. These packages were compared based on simplicity of use, parameter options, documentation availability, and included visualization functions. As the script implementing the classification process will be deployed and maintained at the end of the project, the ease of use and maintenance of the necessary dependencies is an important consideration in the decision of which package to use. Because clustering algorithms are available for each language and are largely interchangeable, differences in the clustering packages were not explicitly investigated.

3.5.1 Packages in R

The Kohonen, Somoclu, and Popsom packages were explored as methods for SOM building in R (Lutz Hamel, 2021; Peter Wittek et al., 2017; Ron Wehrens & Johannes Kruisselbrink, 2018). Of these packages, the Kohonen package is the most popular due to its robustness and high level of documentation (Ron Wehrens & Johannes Kruisselbrink, 2018). Additionally, Kohonen is known for being simple to use, adept at analysis of high dimensional datasets, and computationally efficient (Ron Wehrens & Johannes Kruisselbrink, 2018). Neither Somoclu nor Popsom is as well documented or popular as Kohonen; essentially, these two packages were created to meet a specific need that is not included in the Kohonen package. Somoclu was created for the purpose of creating SOMs in settings where significant parallel processing is required (Peter Wittek et al., 2017). Popsom provides additional visualization functions and several functions which help evaluate model quality (Lutz Hamel, 2021). Of the packages available through R, the Kohonen package is most suited to the needs of this project for ease of use and documentation availability.

3.5.2 Packages in Python

The Minisom, SOMPY, and Scikit-learn SOM packages were tested in Python (Giuseppe Vettigli, 2018; Riley Smith, 2021; Vahid Moosavi, 2021). The Scikit-learn SOM package is an extension of the commonly used Scikit-learn package on Python, making it the most easily accessible and simple to use of the three Python SOM packages. However, the Scikit-learn SOM package compromises robustness and adjustable parameters for ease of use and is therefore not as powerful as the other two packages for complex analysis (Riley Smith, 2021). Minisom is a SOM package based in Numpy, a popular scientific computing Python package, and is designed

to be as simple as possible while still including adjustable parameters and additional visualization options. The third package tested, SOMPY, is somewhat similar to the Somoclu package in R. The goal of SOMPY is to provide a way for the user to create SOMs in a parallel processing environment (Vahid Moosavi, 2021).

Of the Python and R SOM packages, the Minisom package was chosen to be most viable for this project due to its relatively comprehensive documentation compared to the other packages as well as its inclusion of adjustable parameters. Python was ultimately determined to be the most viable platform for this project due to the significant amount of documentation available on ML applications in Python as well as the project team's previous experience in Python. Additionally, numerous clustering algorithms are implemented within the scikit-learn package and could be readily integrated with the Minisom package. (Pedregosa, F. et al., 2021)

In addition to testing SOM package options available in Python, three different clustering packages provided by Scikit-learn in Python were tested. The first package, agglomerative clustering, was ultimately rejected due to the requirement of the user to specify the number of clusters. Because automation is a primary goal of this project, it is important that the clustering mechanism be able to determine an optimal number of clusters for a dataset without user interference. Therefore, the other clustering methods tested from Scikit-learn, affinity propagation and mean shift clustering, were chosen based on their ability to cluster data without the use of a pre-specified "number of clusters" parameter. After visually assessing results from both clustering methods, Mean shift clustering was chosen to move forward with due to its greater ability to group outputs from the SOM into well-represented clusters.

3.6 Data Preprocessing

Two versions of the classification process were ultimately created in separate scripts, the first for observed streamflow and the second for simulated streamflow. Although the scripts are designed to be robust and can handle a variety of input data, there are significant differences in input data format and preprocessing needs between observed gage and simulated datasets. Separating the classification workflow based on the type allows these preprocessing operations to be effectively implemented.

3.6.1 Simulated Hydrograph Realizations

The simulated datasets used to test the classification process were from Stochastic Event Flood Model (SEFM) analyses previously conducted by the TSC Water Resources Engineering and Management Group. The data was available as an SEFM analysis outputs folder of .PLT files, with each file containing one hydrograph realization. Therefore, the simulated data script finds and reads all .PLT files in a designated folder and records each .PLT file as a 15-day hourly hydrograph. As there are no missing values included in simulated input data, the script continues immediately to calculating and normalizing parameters to be input to the SOM.

While the current script for simulated datasets includes only the structure for SEFM data, it is straightforward for a user to change the data import operations for any other model type or file structure. After this modification, the remainder of the script should apply without any additional modifications.

3.6.2 Observed Hydrograph Realizations

There is additional preprocessing necessary when working with observed data to handle data quality issues and to partition into hydrograph realizations. Missing values are found in the data and filled using linear interpolation if less than three continuous datapoints are missing. Any data gaps that are three datapoints and larger are replaced with a missing data flag. The two datapoint limit is maintained regardless of the timestep of the input hydrograph. A stringent interpolation limit is set to ensure the process does not artificially skew the hydrograph realizations, particularly when using longer timesteps.

The data is then partitioned into hydrograph realizations using a sliding window. This both accounts for dependencies in the data over time and performs data expansion to increase its quantity for the ML algorithms (Shorten & Khoshgoftaar, 2019). Each window includes seven days of data. Each consecutive window drops the first day included in the previous window and adds one more day to the end of the window until all the data has been included, keeping six days of overlap between the previous and current hydrograph realizations. In other words, each day is included in seven windows, or hydrograph realizations, total. The windows of data are then reformatted and consolidated into a table that includes columns for the start date of each window and the number of samples obtained per day. Any hydrograph realization that contains the missing data flag is dropped from the set.

Data expansion, also known as data augmentation, is a process by which an initial set of ML inputs are increased in number to improve algorithm performance. This is done because the skill of ML algorithms tends to increase with the amount of presented data. Additionally, additional data helps an algorithm to avoid misclassifications due to small changes in the inputs. Data expansion operations are well established in the ML literature and are standard practice in most workflows when feasible (Mikolajczyk & Grochowski, 2018; Shorten & Khoshgoftaar, 2019; Wong et al., 2016). Sliding window data expansion allows significantly more hydrograph realizations to be presented to the classification process, which can be appropriate when looking broadly across a time series for differences in hydrograph characteristics or event partitioning when metrics for event identification are not specified a priori. However, a sliding window is not suitable in all use cases. Because the same day is repeated throughout the window period, classification of a hydrograph realization can be a function of how the window intersects with a particular event. For example, a large magnitude event may cause a cluster to form based on magnitude alone with significantly different hydrograph shapes as the event moves across the window. It also means that the same day may be present across multiple clusters. The overall classification process remains valid regardless of whether the user chooses to use either a sliding or fixed window, which must be determined based on the desired application.

The code structure in the script file is designed so that the file can handle data sampled at different frequencies (e.g., samples taken every five minutes or every hour), and the number of days per window can be easily adjusted. After preprocessing, the script files for the observed and simulated data are nearly identical.

4.0 Results

This section presents results of the classification process for the selected test datasets. The test locations were chosen to be representative of a wide range of hydrology as well as a mix of observed and simulated data sources. It is possible that there are other parameter combinations besides those described in Section 3.0that would give better results for SEFM or one of the observed datasets individually; however, the goal is to make the clustering process as robust as possible to allow its use with a variety of future datasets without significant user adjustment. All datasets are therefore present using the default parameters.

As described in Section 3.0, the results for each dataset are assessed visually by generating plots of the hydrographs within each SOM node and mean shift cluster. The classification process outputs diagnostic information to aid in understanding the skill of the algorithm. The generated results output into a folder that includes subfolders for plots of the hydrographs in each cluster (displayed at both fixed and changing y-axis ranges to highlight either magnitude or hydrograph shape), plots of the hydrographs in each SOM node (with and without the SOM node weight vector included), plots of distributions from each cluster, and summary Excel sheets of several assessment metrics for each cluster (such as hydrograph volume). Additionally, the Excel output contains information about which SOM node and mean shift cluster each hydrograph has been classified into as well as the mean shift cluster in which each SOM node has been placed and the weight vector of each SOM cell. The user has the option of not generating plot outputs in order to decrease the runtime of the script. However, while the default classification process should be robust across input data, it is strongly recommended that the user review the output to confirm classification skill.

The remainder of the section discusses the classification skill for each of the test datasets. Further information about results outputs as well as instructions for use of the classification script are available in the report appendices.

4.1 Gage Dataset Results

4.1.1 USGS 15 Minute Streamflow Data for the Colorado River

The first set of gage data is from United States Geological Survey (USGS) site 09058000 on the Colorado River near Kremmling, Colorado (U.S. Geological Survey, 2016b). This dataset contains streamflow measurements taken every 15 minutes from June, 2005 through June, 2021, and was retrieved from the USGS water data database on June 29th, 2021. The streamflow

values in this dataset range from 158 ft^3/s to 6230 ft^3/s and have an average around 1030 ft^3/s. Figure 4.1 plots the streamflow over the analyzed period. This dataset is considered to be an example of moderate to high flow gage data along a major river system.

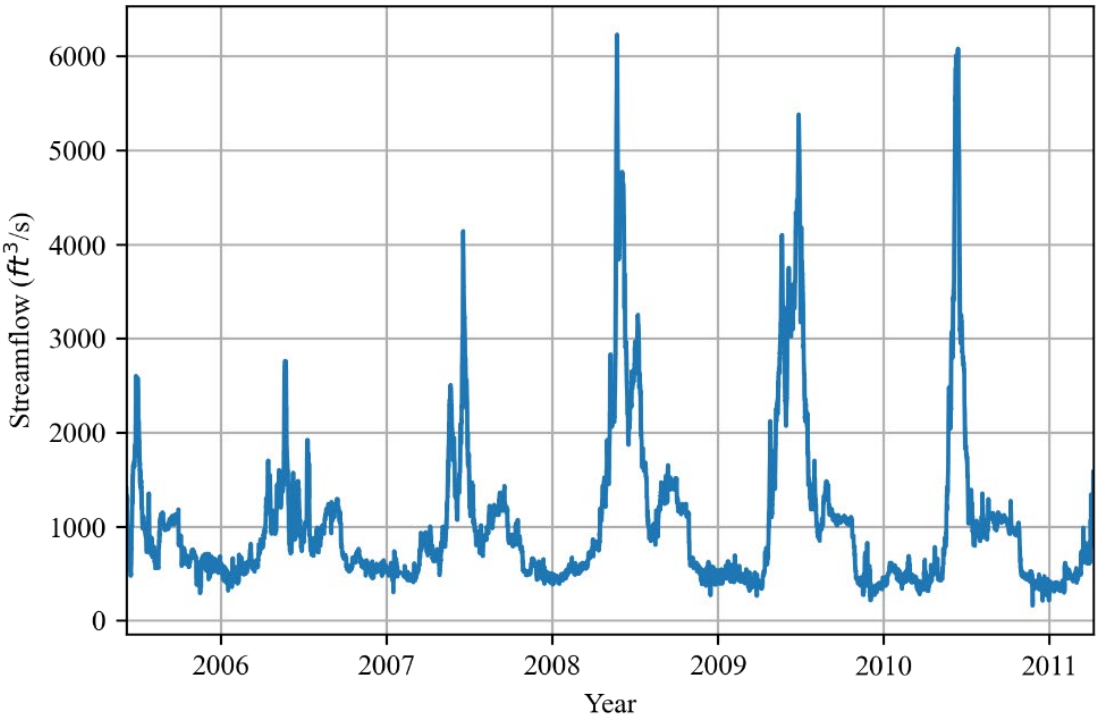


Figure 4.1.—A plot of the USGS Colorado River streamflow dataset over the period of record considered in the classification process (U.S. Geological Survey, 2016b).

The full clusters are given in Figure B.1 and Figure B.2 in appendix B. The plots show favorable results in that each cluster is clearly differentiated in either magnitude or shape (or both) from all other clusters. The SOM and mean shift clustering models have together been able to consolidate the hydrographs into a relatively small number of final categories which are all distinct from one another for this dataset.

The SOM uses several factors when it is classifying the input hydrographs as illustrated in Figure 4.2. The hydrograph magnitude and shape over time are the most intuitive factors from these plots; however, the SOM also uses other patterns and more complex factors which are less easy to observe such as the overall shape of the hydrograph. Additional SOM node weight plots for each dataset were generated when analyzing the dataset. These plots show that the SOM performs well to categorize the hydrograph realizations into representative groups.

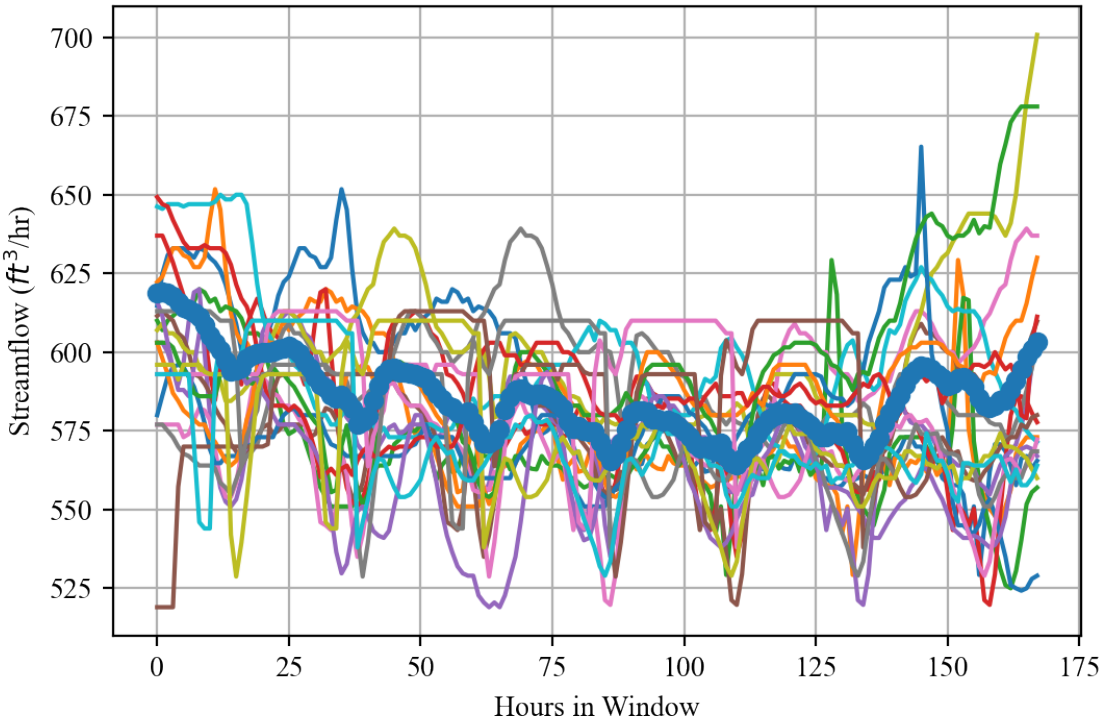


Figure 4.2.—A plot of all hydrographs (shown as thin lines) and the weight vector (shown as a thick line over the hydrograph lines) from a single SOM cell of the USGS Colorado River dataset (U.S. Geological Survey, 2016b).

The dataset was also analyzed with a non-sliding window by partitioning the series into arbitrary seven-day windows from the first day of the series. The classification processes used 138 SOM cells and resulted in six final clusters. The full plots of the clusters are given in Figure B.3 and Figure B.4 in appendix B. The clusters partitioned strongly on hydrograph volume with shape being a secondary characteristic, as evidenced by several clusters containing hydrographs with both increasing and decreasing mean slope. If hydrograph shape is an important consideration, an increase in the bandwidth correction factor would be necessary to increase the number of output clusters.

4.1.2 USGS 15 Minute Streamflow Data for the South Platte River

A second set of gage data was obtained from USGS site 06759500 on the South Platte River at Fort Morgan, Colorado (U.S. Geological Survey, 2016a). The streamflow measurements in this dataset were taken every 15 minutes from June, 2005 through June, 2010 and was retrieved for this project on June 15, 2021. The streamflow values in the dataset have a minimum of 20 ft^3/s, a maximum of 3730 ft^3/s, and an average of 440 ft^3/s. Figure 4.3 plots the streamflow over the analyzed period. This dataset has examples of low and moderate flow through a medium river system.

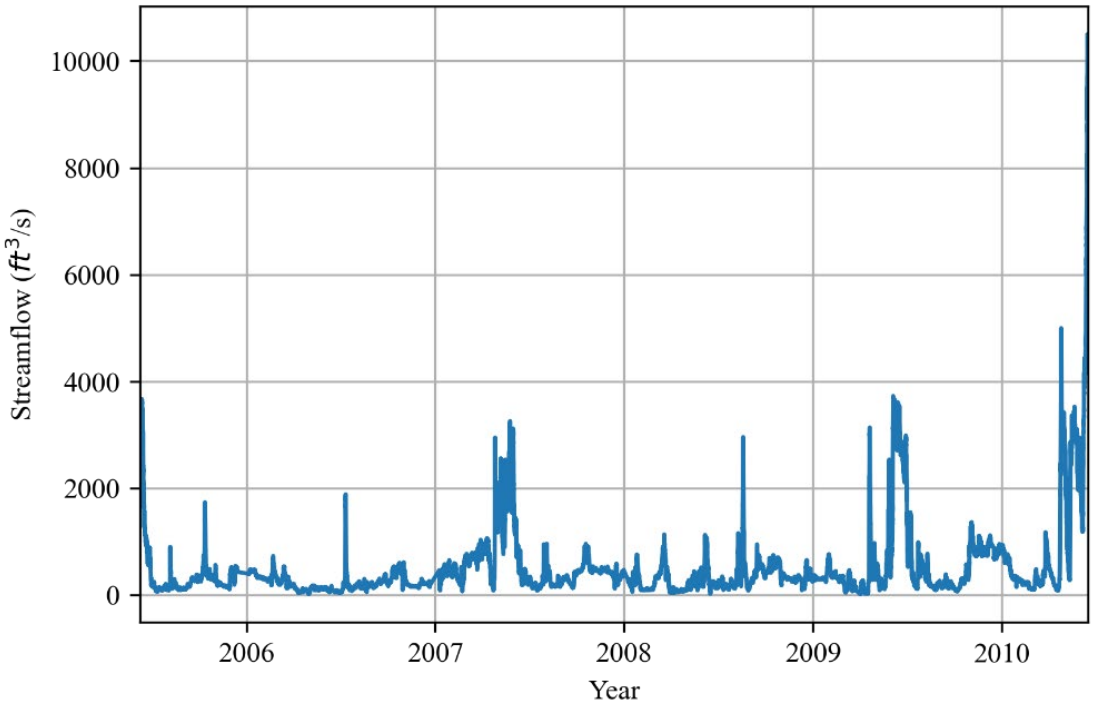


Figure 4.3.—A plot of the USGS South Platte River streamflow dataset over the period of record considered in the classification process (U.S. Geological Survey, 2016a).

When this dataset was input to the SOM, 160 cells were used to categorize the hydrographs. These cells were then further classified into 12 categories by the mean shift clustering algorithm, which are displayed in Figure B.5 and Figure B.6 in appendix B.

The clustering results achieved for this dataset were assessed to be positive due to the distinctness of each of the final clusters as well as the relatively low number of final clusters. Each cluster has a distinct magnitude, mean slope, and shape and provided a meaningful classification of the hydrograph realizations.

4.1.3 Boise State University Data

The third set of gage data is from the Watershed Process Research Group of Boise State University (Boise State University, 2021). This data is from the Dry Creek Experimental Watershed Basin in Idaho and consists of streamflow measurements taken hourly from January, 1999 through December, 2017. The dataset was retrieved on June 2nd, 2021. The streamflow values in the dataset range from 0 ft^3/s to 112 ft^3/s, with an average flow of 4.5 ft^3/s. Figure 4.4 plots the streamflow over the analyzed period. This dataset is considered an example of low flow in a small basin.

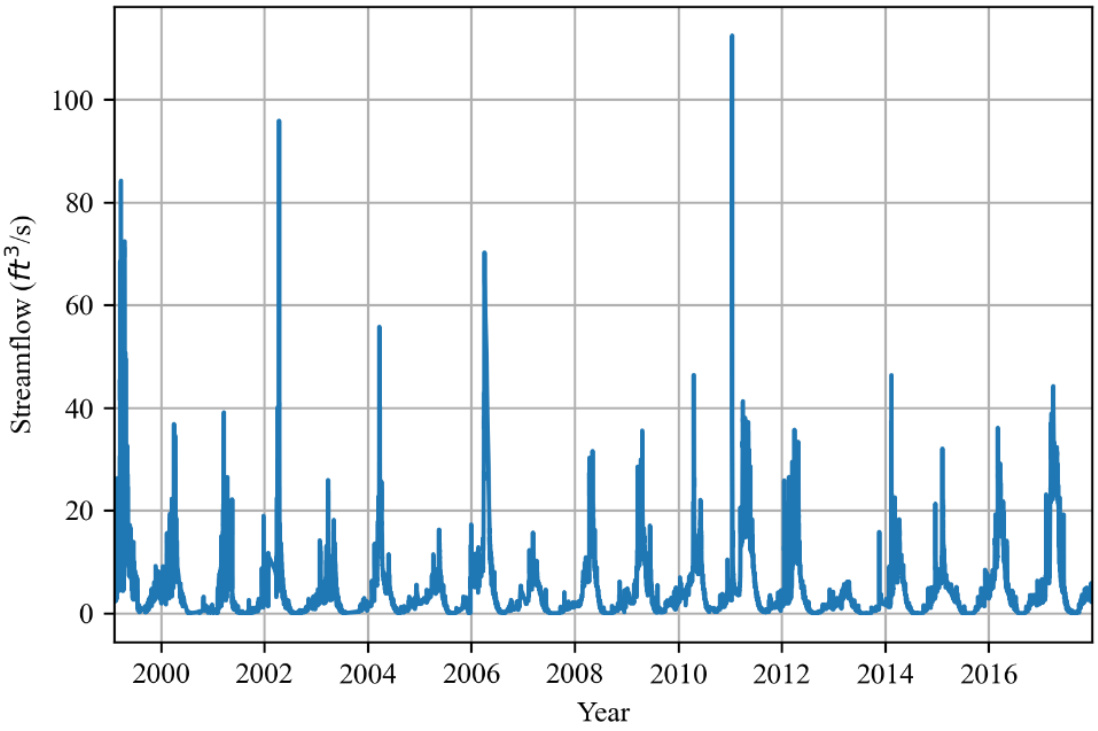


Figure 4.4.—A plot of the Dry Creek streamflow dataset over the period of record considered in the classification process (Boise State University, 2021).

The SOM categorizes the input hydrographs from the Boise gage dataset into 208 cells, which the clustering model then groups into seven total clusters. The hydrograph plots for each of these clusters are shown by Figure B.7 and Figure B.8 in appendix B. Mean slope, magnitude, and shape were again well represented across the final clusters.

4.2 Simulated Data Results

4.2.1 SEFM Data for El Vado Lake

The simulated dataset used in testing was a set of hydrographs generated by the Stochastic Event Flood Model (SEFM) for the El Vado Lake in New Mexico (Bureau of Reclamation, 2016). Hourly streamflow values were generated for each 15-day hydrograph, and the data ranged from 3981 ft^3/s to 55,230 ft^3/s with an average of 4,409 ft^3/s. These values are selected from a case in which Heron Dam is at maximum capacity, and all inflows are passed immediately downstream. Additionally, while the flows in the observed datasets tend to gradually increase or decrease and have multiple peaks at varying locations, the hydrographs in the simulated datasets mostly contain a small number of significant peaks and otherwise remain at a baseflow. Figure 4.5 shows an example of one SOM cell containing SEFM hydrograph data.

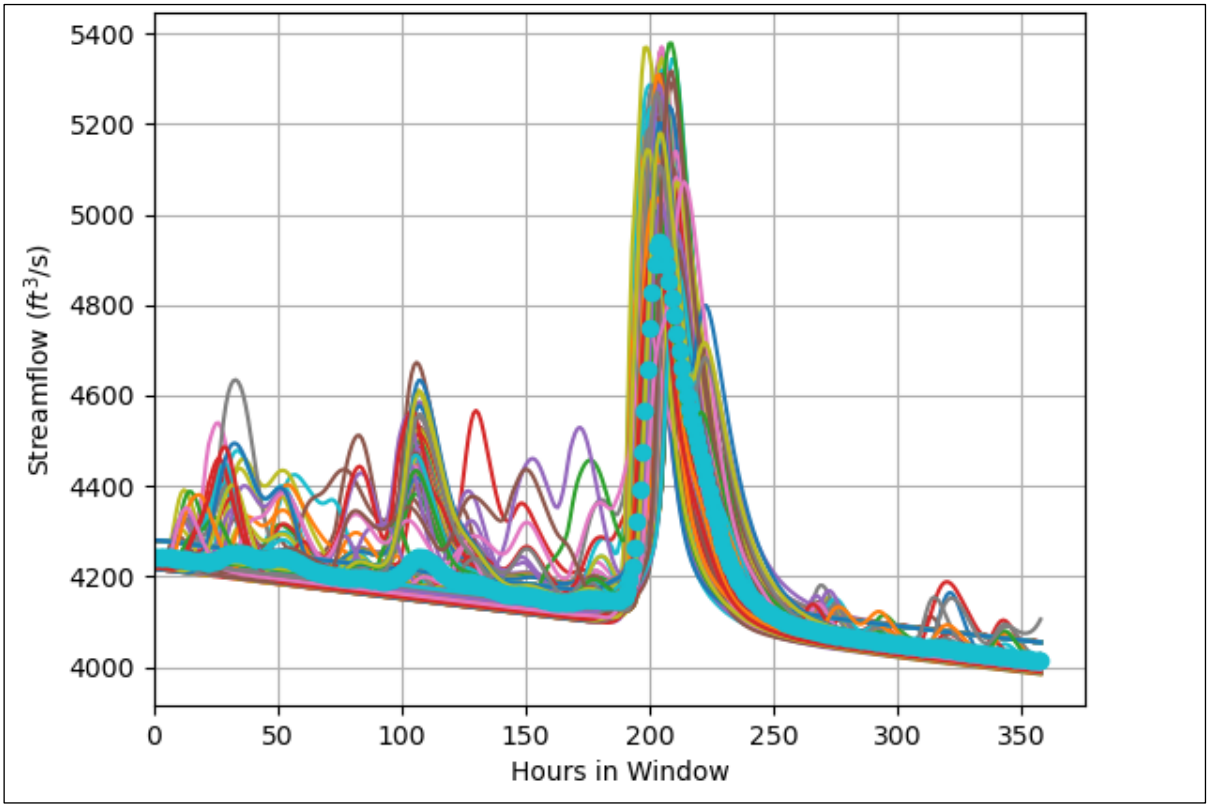


Figure 4.5.—One example of a SOM cell containing hydrographs from SEFM El Vado Lake data (Bureau of Reclamation, 2016).
The individual hydrographs contained in the cell are shown as thin lines and the representative weight vector of the cell is shown as a thicker line on top of the individual hydrographs.

The 20,000 total hydrographs in the SEFM data were classified into 80 SOM cells and nine final cluster categories. Plots for these clusters are included in appendix B as Figure B.9 and Figure B.10.

Although the input hydrographs from this simulated data differ significantly from the those of the observed gage data, the SOM and mean shift clustering methods maintain a high degree of classification skill. The differences in hydrograph shape and magnitude remain well captured. This i likely helped by the smoothness of the modeled hydrograph giving emphasis to small changes in the hydrograph shape. The degree of classification skill gives confidence in the overall classification workflow across different hydrograph sources.

5.0 Conclusions

This work developed a process to classify hydrograph realizations by similarity, providing a means to summarize many thousands of hydrograph realizations into a few representative groups that describe the overall streamflow behavior. This process integrated two classification algorithms within the workflow. The first classification employed a SOM ML algorithm for a broad first classification and dimensionality reduction. The second classification used the SOM outputs within the mean shift clustering algorithm to further reduce the number of classification groups. Validation of the approach was done by a qualitative assessment across test datasets representing a range of observed and simulated hydrologic behavior.

Multiple use cases are envisioned for the hydrograph classification process. The primary use is anticipated to be within stochastic hydrology applications. A reduction to representative hydrographs from a larger set of hydrograph realizations will facilitate better communication of hydrologic behavior. Additionally, it will simplify subsequent analyses that use the hydrographs as input while helping to ensure that the selected hydrographs are representative of the full range of basin behavior. This use case is particularly important to the Reclamation Dam Safety program which utilizes output from stochastic hydrologic models as input to subsequent flood routing computations and ultimately hydrologic risk. Another anticipated use case is analysis of observed gage information to better understand the range of historical behavior. This can be useful in the construction of stochastic hydrologic models, to ensure that the simulated output is representative, and for water management, to better understand the pattern of common events.

This effort gives Reclamation an improved hydrograph classification capability, and future efforts could provide additional refinement. While several representative test datasets were used to configure the default classifications parameters, this does not encompass the full range of input data. The classification process should be subject to continuing validation as it is applied to more data. If the classification skill becomes unsatisfactory, the SOM size or mean shift bandwidth default parameterization should be updated with additional logic based on the identified deficiencies. Additionally, the current approach is focused on classifying pre-existing hydrograph realizations. This workflow could readily be expanded to produce hydrograph realizations that are representative of the cluster. This could improve computation time by reducing the need for rainfall/runoff modeling to explore a particular hydrologic regime.

The work completed here to develop a methodology for hydrograph classification can be used to support future Dam Safety work where analyses of large numbers of hydrographs are required for decision-making. While each project represents unique needs and challenges when it comes to selection of hydrographs for use, this workflow provides a missing link in moving from full dataset to a smaller subset of representative hydrographs. The code can easily be incorporated into tailored code developed for each specific project need. Examples of such dam safety applications include construction risk analysis, design support, or refinement of risk.

6.0 References

Archfield, S. A., Kennen, J. G., Carlisle, D. M., & Wolock, D. M. (2014). AN OBJECTIVE AND PARSIMONIOUS APPROACH FOR CLASSIFYING NATURAL FLOW REGIMES AT A CONTINENTAL SCALE: CLASSIFICATION OF NATURAL FLOW REGIMES. *River Research and Applications*, *30*(9), 1166–1183. https://doi.org/10.1002/rra.2710

Ayesha, S., Hanif, M. K., & Talib, R. (2020). Overview and comparative study of dimensionality reduction techniques for high dimensional data. *Information Fusion*, *59*, 44–58. https://doi.org/10.1016/j.inffus.2020.01.005

Bandyopadhyay, S., & Saha, S. (2013). *Unsupervised Classification: Similarity Measures, Classical and Metaheuristic Approaches, and Applications*. Springer, Berlin, Heidelberg.

Barreto, G. A. (2007). Time Series Prediction with the Self-Organizing Map: A Review. In B. Hammer & P. Hitzler (Eds.), *Perspectives of Neural-Symbolic Integration* (pp. 135–158). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-73954-8_6

Beven, K. J. (2012). *Rainfall-Runoff Modelling: The Primer* (2nd ed.). Wiley-Blackwell.

Boise State University. (2021, November 4). *Dry Creek Experimental Watershed* [Dry Creek Experimetnal Watersheed]. https://www.boisestate.edu/drycreek/

Brunner, M. I., Viviroli, D., Furrer, R., Seibert, J., & Favre, A. (2018). Identification of Flood Reactivity Regions via the Functional Clustering of Hydrographs. *Water Resources Research*, *54*(3), 1852–1867. https://doi.org/10.1002/2017WR021650

Bureau of Reclamation. (2003a). *Probabilistic Extreme Flood Hydrographs That Use PaleoFlood Data for Dam Safety Applications* (Dam Safety Office No. 03–03). Department of the Interior.

Bureau of Reclamation. (2003b). *Stochastic Modeling Methods* (Dam Safety Office No. 03–04). Department of the Interior.

Bureau of Reclamation. (2016). *El Vado Dam Hydrologic Hazard for Corrective Action Study* (Technical Memorandum No. 8250-2016–014). Technical Services Center.

Bureau of Reclamation. (2019). *Stochastic Flood Frequency Analysis at Folsom Dam:* (Technical Memorandum ENV-2019-070; p. 232). Bureau of Reclamation.

Céréghino, R., & Park, Y.-S. (2009). Review of the Self-Organizing Map (SOM) approach in water resources: Commentary. *Environmental Modelling & Software*, *24*(8), 945–947. https://doi.org/10.1016/j.envsoft.2009.01.008

Choubin, B., Solaimani, K., Habibnejad Roshan, M., & Malekian, A. (2017). Watershed classification by remote sensing indices: A fuzzy c-means clustering approach. *Journal of Mountain Science*, *14*(10), 2053–2063. https://doi.org/10.1007/s11629-017-4357-4

Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(5), 603–619. https://doi.org/10.1109/34.1000236

Devia, G. K., Ganasri, B. P., & Dwarakish, G. S. (2015). A Review on Hydrological Models. *Aquatic Procedia*, *4*, 1001–1007. https://doi.org/10.1016/j.aqpro.2015.02.126

Dutta, D., Welsh, W. D., Vaze, J., Kim, S. S. H., & Nicholls, D. (2012). A Comparative Evaluation of Short-Term Streamflow Forecasting Using Time Series Analysis and Rainfall-Runoff Models in eWater Source. *Water Resources Management*, *26*(15), 4397–4415. https://doi.org/10.1007/s11269-012-0151-9

Federal Emergency Management Agency. (2020). *Guidance for Flood Analysis and Mapping* (Guidance Document No. 85). Department of Homeland Security. https://www.fema.gov/sites/default/files/documents/fema_flood-profiles-guidance.pdf

Fowler, K., Coxon, G., Freer, J., Peel, M., Wagener, T., Western, A., Woods, R., & Zhang, L. (2018). Simulating Runoff Under Changing Climatic Conditions: A Framework for Model Improvement. *Water Resources Research*, *54*(12), 9812–9832. https://doi.org/10.1029/2018WR023989

Giuseppe Vettigli. (2018). *MiniSom: Minimalistic and NumPy-based implementation of the Self Organizing Map* (2.2.9) [Python]. https://github.com/JustGlowing/minisom

Gordon, A. D. (1987). A Review of Hierarchical Classification. *Journal of the Royal Statistical Society. Series A (General)*, *150*(2), 119. https://doi.org/10.2307/2981629

Hancer, E., Xue, B., & Zhang, M. (2020). A survey on feature selection approaches for clustering. *Artificial Intelligence Review*, *53*(6), 4519–4545. https://doi.org/10.1007/s10462-019-09800-w

Hannah, D. M., Smith, B. P. G., Gurnell, A. M., & McGregor, G. R. (2000). An approach to hydrograph classification. *Hydrological Processes*, *14*(2), 317–338.

Harris, N. M., Gurnell, A. M., Hannah, D. M., & Petts, G. E. (2000). Classification of river regimes: A context for hydroecology. *Hydrological Processes*, *14*(16–17), 2831–2848. https://doi.org/10.1002/1099-1085(200011/12)14:16/17<2831::AID-HYP122>3.0.CO;2-O

Holman, K. D. (2018). Characterizing Antecedent Conditions Prior to Annual Maximum Flood Events in a High-Elevation Watershed Using Self-Organizing Maps. *Journal of Hydrometeorology*, *19*(11), 1721–1730. https://doi.org/10.1175/JHM-D-17-0229.1

Kalteh, A. M., Hjorth, P., & Berndtsson, R. (2008). Review of the self-organizing map (SOM) approach in water resources: Analysis, modelling and application. *Environmental Modelling & Software*, *23*(7), 835–845. https://doi.org/10.1016/j.envsoft.2007.10.001

Khan, M. Q. I., Venkataratnam, L., Rao, B. R. M., Rao, D. P., & Subrahmanyam, C. (2001). International Classification and Codification of Watersheds and River Basins. *Journal of Water Resources Planning and Management*, *127*(5), 306–315. https://doi.org/10.1061/(ASCE)0733-9496(2001)127:5(306)

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, *43*(1), 59–69. https://doi.org/10.1007/BF00337288

Kohonen, T. (2001). *Self-Organizing Maps* (3rd ed., Vol. 30). Springer, Berlin, Heidelberg.

Kohonen, T. (2013). Essentials of the self-organizing map. *Twenty-Fifth Anniversay Commemorative Issue*, *37*, 52–65. https://doi.org/10.1016/j.neunet.2012.09.018

Kohonen, T., Oja, E., Simula, O., Visa, A., & Kangas, J. (1996). Engineering applications of the self-organizing map. *Proceedings of the IEEE*, *84*(10), 1358–1384. https://doi.org/10.1109/5.537105

Kuentz, A., Arheimer, B., Hundecha, Y., & Wagener, T. (2017). Understanding hydrologic variability across Europe through catchment classification. *Hydrology and Earth System Sciences*, *21*(6), 2863–2879. https://doi.org/10.5194/hess-21-2863-2017

Kumar, S. (2020, September 13). *Understanding 8 types of Cross-Validation*. Towards Data Science. https://towardsdatascience.com/understanding-8-types-of-cross-validation-80c935a4976d

LaFontaine, J. H., Hay, L. E., Viger, R. J., Regan, R. S., & Markstrom, S. L. (2015). Effects of Climate and Land Cover on Hydrology in the Southeastern U.S.: Potential Impacts on Watershed Planning. *JAWRA Journal of the American Water Resources Association*, *51*(5), 1235–1261. https://doi.org/10.1111/1752-1688.12304

Land & Water Australia. (2009). *Ecohydrological regionalisation of Australia: A tool for management and science*. Land and Water Australia. http://lwa.gov.au

Lin, G.-F., & Wang, C.-M. (2006). Performing cluster analysis and discrimination analysis of hydrological factors in one step. *Advances in Water Resources*, *29*(11), 1573–1585. https://doi.org/10.1016/j.advwatres.2005.11.008

Lutz Hamel. (2021). *Popsom* (5.2) [R]. https://github.com/lutzhamel/popsom

Maharaj, E. A., D'Urso, P., & Caiado, J. (2019). *Time Series Clustering and Classification* (1st ed.). Chapman and Hall/CRC.

Maimon, O., & Rokach, L. (2005). *Data mining and knowledge discovery handbook*. Springer.

Marco, J. B., Harboe, R., & Salas, J. D. (1993). *Stochastic Hydrology and its Use in Water Resources Systems Simulation and Optimization* (Vol. 237). Springer, Dordrecht.

Matsumoto, K., & Miyamoto, M. (2018). *Clustering Multiple Hydrographs Using Mathematical Optimization*. 1358–1349. https://doi.org/10.29007/6r61

McCuen, R. H. (1973). The role of sensitivity analysis in hydrologic modeling. *Journal of Hydrology*, *18*(1), 37–53.

McManamay, R. A., Bevelhimer, M. S., & Kao, S.-C. (2014). Updating the US hydrologic classification: An approach to clustering and stratifying ecohydrologic data. *Ecohydrology*, *7*(3), 903–926. https://doi.org/10.1002/eco.1410

McManamay, R. A., & DeRolph, C. R. (2019). A stream classification system for the conterminous United States. *Scientific Data*, *6*(1), 190017. https://doi.org/10.1038/sdata.2019.17

Merritt, A., Lane, B., & Hawkins, C. (2021). Classification and Prediction of Natural Streamflow Regimes in Arid Regions of the USA. *Water*, *13*(3), 380. https://doi.org/10.3390/w13030380

Mikolajczyk, A., & Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, 117–122. https://doi.org/10.1109/IIPHDW.2018.8388338

Miljkovic, D. (2017). Brief review of self-organizing maps. *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 1061–1066. https://doi.org/10.23919/MIPRO.2017.7973581

Milos Gajdos. (2017, January 13). *Self-organizing Maps in Go*. Cybernetist. https://cybernetist.com/2017/01/13/self-organizing-maps-in-go/

Monk, W. A., Wood, P. J., Hannah, D. M., & Wilson, D. A. (2007). Selection of river flow indices for the assessment of hydroecological change. *River Research and Applications*, *23*(1), 113–122. https://doi.org/10.1002/rra.964

Moore, R. J., Bell, V. A., & UK Environment Agency. (2001). *Comparison of rainfall-runoff models for flood forecasting—Part 1: Literature review of models.* UK Environment Agency.

Mosley, M. P. (1981). Delimitation of New Zealand hydrologic regions. *Journal of Hydrology*, *49*(1–2), 173–192. https://doi.org/10.1016/0022-1694(81)90211-0

Nathan, R. J., & McMahon, T. A. (1990). Identification of homogeneous regions for the purposes of regionalisation. *Journal of Hydrology*, *121*(1–4), 217–238. https://doi.org/10.1016/0022-1694(90)90233-N

Olden, J. D., Kennard, M. J., & Pusey, B. J. (2012). A framework for hydrologic classification with a review of methodologies and applications in ecohydrology. *Ecohydrology*, *5*(4), 503–518. https://doi.org/10.1002/eco.251

Ouellet Dallaire, C., Lehner, B., Sayre, R., & Thieme, M. (2019). A multidisciplinary framework to derive global river reach classifications at high spatial resolution. *Environmental Research Letters*, *14*(2), 024003. https://doi.org/10.1088/1748-9326/aad8e9

Palacio-Niño, J.-O., & Berzal, F. (2019). Evaluation Metrics for Unsupervised Learning Algorithms. *ArXiv:1905.05667 [Cs, Stat]*. http://arxiv.org/abs/1905.05667

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2021). *Scikit-learn: Machine Learning in Python* (1.0.1) [Python]. https://scikit-learn.org/stable/index.html

Peñas, F. J., Barquín, J., Snelder, T. H., Booker, D. J., & Álvarez, C. (2014). The influence of methodological procedures on hydrological classification performance. *Hydrology and Earth System Sciences*, *18*(9), 3393–3409. https://doi.org/10.5194/hess-18-3393-2014

Peter Wittek, Shi Chao Gao, Ik Soo Lim, & Li Zhao. (2017). Somoclu: An Efficient Parallel Library for Self-Organizing Maps. *Journal of Statistical Software*, *78*(9), 1–21. https://doi.org/10.18637/jss.v078.i09

Pianosi, F., & Wagener, T. (2016). Understanding the time-varying importance of different uncertainty sources in hydrological modelling using global sensitivity analysis. *Hydrological Processes*, *30*(22), 3991–4003. https://doi.org/10.1002/hyp.10968

Poff, N. L., & Ward, J. V. (1989). Implications of Streamflow Variability and Predictability for Lotic Community Structure: A Regional Analysis of Streamflow Patterns. *Canadian Journal of Fisheries and Aquatic Sciences*, *46*(10), 1805–1818. https://doi.org/10.1139/f89-228

Potyondy, J. P., & Geier, T. W. (2011). *Watershed Condition Classification Technical Guide* (Forest Service No. 978; p. 49). U.S. Department of Agriculture.

Rao, A. R., & Srinivas, V. V. (2008). Regionalization by Hybrid Cluster Analysis. In *Regionalization of Watersheds* (Vol. 58, pp. 17–55). Springer Netherlands. https://doi.org/10.1007/978-1-4020-6852-2_2

Riley Smith. (2021). *Sklearn-som* (1.1.0) [Python]. https://pypi.org/project/sklearn-som/

Ron Wehrens & Johannes Kruisselbrink. (2018). kohonen: Supervised and Unsupervised Self-Organising Maps. *Journal of Statistical Software*, *87*(7), 1–8. https://doi.org/10.18637/jss.v087.i07

Sauquet, E., Shanafield, M., Hammond, J. C., Sefton, C., Leigh, C., & Datry, T. (2021). Classification and trends in intermittent river flow regimes in Australia, northwestern Europe and USA: A global perspective. *Journal of Hydrology*, *597*, 126170. https://doi.org/10.1016/j.jhydrol.2021.126170

Schmitt, L., Maire, G., Nobelis, P., & Humbert, J. (2007). Quantitative morphodynamic typology of rivers: A methodological study based on the French Upper Rhine basin. *Earth Surface Processes and Landforms*, *32*(11), 1726–1746. https://doi.org/10.1002/esp.1596

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, *6*(1), 60. https://doi.org/10.1186/s40537-019-0197-0

Siddiqui, S. F., Zapata-Rios, X., Torres-Paguay, S., Encalada, A. C., Anderson, E. P., Allaire, M., Costa Doria, C. R., & Kaplan, D. A. (2021). Classifying flow regimes of the Amazon basin. *Aquatic Conservation: Marine and Freshwater Ecosystems*, *31*(5), 1005–1028. https://doi.org/10.1002/aqc.3582

Singh, V. P. (1997). Effect of spatial and temporal variability in rainfall and watershed characteristics on stream flow hydrograph. *Hydrological Processes*, *11*(12), 1649–1669. https://doi.org/10.1002/(SICI)1099-1085(19971015)11:12<1649::AID-HYP495>3.0.CO;2-1

Sit, M., Demiray, B. Z., Xiang, Z., Ewing, G. J., Sermet, Y., & Demir, I. (2020). A comprehensive review of deep learning applications in hydrology and water resources. *Water Science and Technology*, *82*(12), 2635–2670. https://doi.org/10.2166/wst.2020.369

Sitterson, J., Knightes, C., Parmar, R., Wolfe, K., Avant, B., & Muche, M. (2018). An Overview of Rainfall-Runoff Model Types. *International Congress on Environmental Modelling and Software.*, *41*, 10.

Snelder, T. H., & J. Booker, D. (2013). NATURAL FLOW REGIME CLASSIFICATIONS ARE SENSITIVE TO DEFINITION PROCEDURES: SENSITIVITY OF FLOW REGIME CLASSIFICATIONS TO DEFINITION PROCEDURE. *River Research and Applications*, *29*(7), 822–838. https://doi.org/10.1002/rra.2581

Song, M., Yang, H., Siadat, S. H., & Pechenizkiy, M. (2013). A comparative study of dimensionality reduction techniques to enhance trace clustering performances. *Expert Systems with Applications*, *40*(9), 3722–3737.

Sorooshian, S., Hsu, K., Coppola, E., Tomassetti, B., Verdecchia, M., & Visconti, G. (2008). *Hydrological modelling and the water cycle: Coupling the atmospheric and hydrological models*. Springer.

Steinbach, M., Ertöz, L., & Kumar, V. (2004). The Challenges of Clustering High Dimensional Data. In L. T. Wille (Ed.), *New Directions in Statistical Physics: Econophysics, Bioinformatics, and Pattern Recognition* (pp. 273–309). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-08968-2_16

Tabari, H. (2019). Statistical Analysis and Stochastic Modelling of Hydrological Extremes. *Water*, *11*(9), 1861. https://doi.org/10.3390/w11091861

Ternynck, C., Ben Alaya, M. A., Chebana, F., Dabo-Niang, S., & Ouarda, T. B. M. J. (2016). Streamflow Hydrograph Classification Using Functional Data Analysis. *Journal of Hydrometeorology*, *17*(1), 327–344. https://doi.org/10.1175/JHM-D-14-0200.1

U.S. Environmental Protection Agency. (2013). *Level III Ecoregions of the Continental United States*. https://www.epa.gov/eco-research/level-iii-and-iv-ecoregions-continental-united-states

U.S. Geological Survey. (2016a). *USGS Water Data for the Nation—Gage 06759500*. U.S. Geological Survey. https://waterdata.usgs.gov/co/nwis/inventory/?site_no=06759500

U.S. Geological Survey. (2016b). *USGS Water Data for the Nation—Gage 09058000*. U.S. Geological Survey. https://waterdata.usgs.gov/monitoring-location/09058000/#parameterCode=00065&period=P7D

Vahid Moosavi. (2021). *SOMPY: A Python Library for Self Organizing Map (SOM)* (1.1.1) [Python]. https://github.com/sevamoo/SOMPY

Vasudevan, R. K., Ziatdinov, M., Vlcek, L., & Kalinin, S. V. (2021). Off-the-shelf deep learning is not enough, and requires parsimony, Bayesianity, and causality. *Npj Computational Materials*, *7*(1), 16. https://doi.org/10.1038/s41524-020-00487-0

Wang, W., Liu, J., Li, C., Liu, Y., & Yu, F. (2021). Data Assimilation for Rainfall-Runoff Prediction Based on Coupled Atmospheric-Hydrologic Systems with Variable Complexity. *Remote Sensing*, *13*(4), 595. https://doi.org/10.3390/rs13040595

Wang, X., Smith, K. A., & Hyndman, R. J. (2005). Dimension Reduction for Clustering Time Series Using Global Characteristics. In V. S. Sunderam, G. D. van Albada, P. M. A. Sloot, & J. Dongarra (Eds.), *Computational Science – ICCS 2005* (Vol. 3516, pp. 792–795). Springer Berlin Heidelberg. https://doi.org/10.1007/11428862_108

Wohlfart, C., Liu, G., Huang, C., & Kuenzer, C. (2016). A River Basin over the Course of Time: Multi-Temporal Analyses of Land Surface Dynamics in the Yellow River Basin (China) Based on Medium Resolution Remote Sensing Data. *Remote Sensing*, *8*(3), 186. https://doi.org/10.3390/rs8030186

Wong, S. C., Gatt, A., Stamatescu, V., & McDonnell, M. D. (2016). Understanding data augmentation for classification: When to warp? *ArXiv:1609.08764 [Cs]*. http://arxiv.org/abs/1609.08764

Yugesh Verma. (2021, August 8). *Hands-On Tutorial on Mean Shift Clustering Algorithm*. Analytics India Magazine. https://analyticsindiamag.com/hands-on-tutorial-on-mean-shift-clustering-algorithm/

Zaerpour, M., Hatami, S., Sadri, J., & Nazemi, A. (2021). A global algorithm for identifying changing streamflow regimes: Application to Canadian natural streams (1966–2010). *Hydrology and Earth System Sciences*, *25*(9), 5193–5217. https://doi.org/10.5194/hess-25-5193-2021

7.0 Acknowledgments

The authors would like to thank Kathleen Holman, Jonathan East, and Subhrendu Gangopadhyay for their thoughts and feedback during the project. The authors would also like to thank the Reclamation Dam Safety Office for funding the project.

Appendix A

Instructions

A.1 Descriptions of Files Included in Bundle

Section A.1 provides descriptions of each file in the setup bundle. These files can be obtained from the project Gitlab repository along with example datasets that can be used to test the script file performance.

1. Environment File: *som_script_env.yml*

   This file is used to create an environment for running the script. It downloads all packages that are required for running the script so that the user does not have to do any additional set up (other than adjusting user inputs in the script itself).

2. SEFM script file: *som_script.py*

   This script contains code to run the SOM and clustering algorithms for simulated SEFM input data.

3. Gage script file: *som_script_gage.py*

   This script contains code to run the SOM and clustering algorithms for observed gage data.

A.2 Input Data Format/Preprocessing

Section A.2 gives a description of how the input data should be formatted prior to running the script file. The formatting requirements are different for simulated and gage data.

A.2.1 Simulated Data

SEFM data should be formatted as each hydrograph contained in one .plt file within a folder. The script begins reading streamflow values in the third line from the top of the file. Therefore, each hydrograph file may have up to two header lines that are not streamflow values. Each file should contain one column of streamflow values in units of ft^3/s.

A.2.2 Gage Data

Gage data should be contained within one comma separated (csv) file. There should be two columns: the first with dates and times for each measurement entry (formatted as month/day/year hour:minute with hours ranging from 0 to 23) and the second column with streamflow measurements in units of ft^3/s. Only one header line containing column names should be present above the streamflow values.

A.3 Environment Setup

Section A.3 provides instructions for setting up the environment to be used for running the script once the bundle of setup files has been downloaded.

1. Install Miniforge and select the option to "add to path" during the installation. Choose the most current version of Miniforge that is available for the operating system being utilized.

2. Open a command prompt window such as Git Bash or Windows Command Prompt within the setup folder.

3. In the command line, type "*conda env create -f som_script_env.yml*" and hit enter to create a build environment from the provided configuration file.

4. In the command line, type "*conda activate som_script_env*". The configuration file has already automatically named this new environment "*som_script_env*".

    a. To verify that the environment was correctly installed and available, type "*conda env list*" into the command line, and check that "*som_scrip_env*" is listed in the output.

A.4 User Inputs to the Script

Section A.4 provides details on user inputs to the script that must be entered according to properties of the input data. The script file can be edited through any text editor, including Windows Notepad or Notepad++. It can also be edited though a Python integrated development environment, such as PyCharm.

1. In line 78, set *fileName* or *fileLocation* equal to the name of the file containing input data.

2. In line 80, set *d* equal to the number of days that make up one hydrograph. For gage data, this number will dictate how many days are in each sliding window created from the input data.

3. In line 82, set *sample_freq* equal to the number of datapoints recorded in one day for the input data. For example, if the data is hourly, then "*sample_freq = 24*".

4. In line 85, the user can set plots equal to either "*True*" or "*False*" to choose whether intermediate plots (hydrographs in SOM cells, distribution plots for each cluster, etc.) will be generated. Cluster plots are generated every time the script runs regardless of the user's choice. Setting plots to "False" will reduce runtime.

5. In line 88, the user can choose to adjust the correction factor that adjusts cluster sorting depth. Increasing this value will increase the number of clusters. The default correction value was chosen based on sensitivity tests for multiple types of data. Users should review the output classifications prior to adjusting this parameter.

A.5 Running the Script File

Section A.5 describes how to run the script file in a command prompt window.

1. Prior to running the script, type "*activate som_script_env*" in the command line to ensure that the previously created environment is being used.

2. In the command line, type either "python som_script_SEFM.py" or "python som_script_gage.py" depending on whether the input data will be simulated from SEFM or observed gage data.

3. As the script runs, it will output updates on its progress in the command window. The datasets used for testing required around 30 minutes of runtime. However, computation time will vary based on the size of the dataset and the computer configuration.

A.6 Classification Outputs

The classification process outputs numerous plots and files at the end of the analysis. These include class assignments for the hydrograph realizations as well as diagnostic data. The diagnostic data can be optionally disabled to improve the solution times. It is strongly recommended that users carefully review the outputs prior to using the classification in case modifications to the algorithm input parameters are necessary to refine the assignments of the hydrograph realizations.

A.6.1 Results Spreadsheet

1. The "*Hydrograph Key*" sheet contains a list of every hydrograph from the input data as well as which SOM node and final mean shift cluster the hydrograph realization was classified.

2. All sheets after the "*Hydrograph Key*" are in pairs, with two sheets for each cluster. The first sheet contains all hydrograph data from the cluster. The second sheet lists the SOM cells within the cluster and the weight vector for each cell.

A.6.2 Cluster Metrics Folder

1. Contains a csv file for each cluster that gives the SOM nodes contained in the mean shift cluster. For each SOM node, the parameters include the average hydrograph volume, minimum/maximum difference between a hydrograph value in the SOM node and the node weight vector, average slope of the SOM node weight vector, and number of intersections of hydrographs with the SOM node weight vector.

A.6.3 Cluster Plots Folder

1. Contains a png image for each cluster. Each image shows a plot of every hydrograph contained in the cluster along with a thicker line that represents the average nodal weight vector of the cluster.

2. The x and y-axis ranges are not held constant between plots so that the shape over time in each cluster is more apparent.

A.6.4 Fixed range Cluster Plots Folder

1. Contains a png image for each cluster. Each image shows a plot of every hydrograph contained in the cluster along with a thicker line that represents the average nodal weight vector of the cluster.

2. The x and y-axis ranges are held constant in every plot so that flow magnitude can be compared between clusters.

A.6.5 SOM Cell Plots Folder

1. Contains a png image for every SOM node. Each image shows a plot of every hydrograph contained in each node of the SOM.

2. The x and y-axis ranges are not held constant between plots so that shape over time is more apparent.

A.6.6 SOM Cell Plots Weight Folder

1. Contains a png image for every SOM node. Each image shows a plot of every hydrograph contained in one node of the SOM along with a thicker line that represents the weight vector of the SOM cell.

2. The x and y-axis ranges are not held constant between plots so that shape over time is more apparent.

A.6.7 Distributions Folder

1. Contains four distribution plots for each cluster: hydrograph mean, number of peaks, largest peak value, and hydrograph volume.

2. Each distribution plot is generated by calculating the parameter for each SOM node included in the cluster using the hydrographs within the SOM node.

A.7 Interpreting the Classification

Users should review the classification output critically to confirm that it is capturing behavior for the basin under analysis. While the particular features under investigation will change among basins, the output structure is intended to highlight different features at various stages of the classification to facilitate the user review. The following descriptions illustrate some the features that could be reviewed for each analysis output.

A.7.1 Results Spreadsheet

1. This output can be used to determine which cluster or SOM node a specific hydrograph was sorted into, or to examine the weights of the SOM nodes contained within each cluster.

A.7.2 Cluster Metrics Folder

1. The csv files contained in this folder can be used to identify which SOM nodes have been sorted into each cluster. The user could then use the SOM node indices from these files to gain further understanding of the classification represented in the cluster by referencing the SOM node plots to visualize the hydrographs in the SOM node.

2. The other metrics, especially hydrograph volume, displayed in each cluster's file can also be used to gauge the difference between the flow magnitudes and shapes in different clusters, and can be used as a check to make sure that the clusters are distinct from each other.

A.7.3 Cluster/Fixed Range Cluster Plots Folders

1. The images in the Cluster Plots folder can be used to view the shape over time in each cluster and to assess whether the cluster weight line is an accurate representation of each cluster's characteristics.

2. The images in the Fixed Range Cluster Plots folder can be used to assess how each cluster's flow magnitude compares to the others.

A.7.4 SOM Cell Plot Weight Folders

1. These plots can be used to view the hydrographs categorized into each SOM node, and visually check that the categorizations seem distinct from one another. The plots with the SOM node weight vector included can be used to check whether the SOM node is accurately representing the characteristics of the hydrographs within it.

A.7.5 Distributions Folder

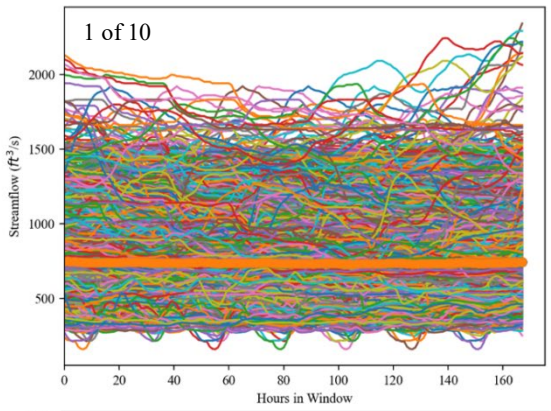
1. The histograms in this folder are visualizations of the parameters contained in the Cluster Metrics spreadsheets. Therefore, they can be used similarly to the Cluster Metrics to confirm that the clusters each have distinct properties such as mean and volume distribution as well as to gain a deeper understanding of what types of hydrographs each cluster represents.

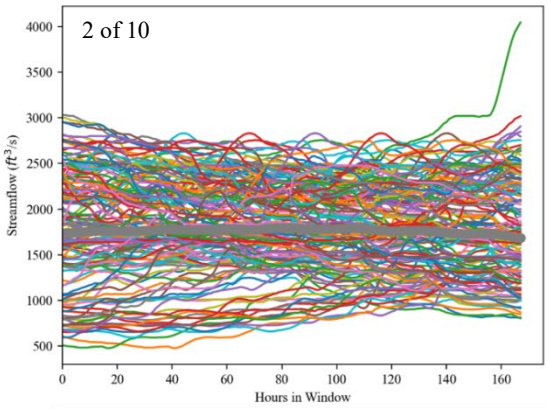
Appendix B

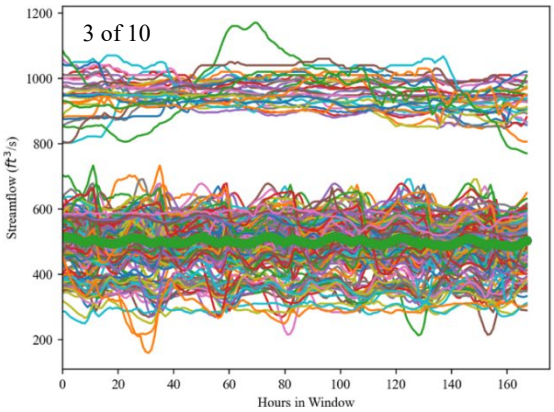
Cluster Plots

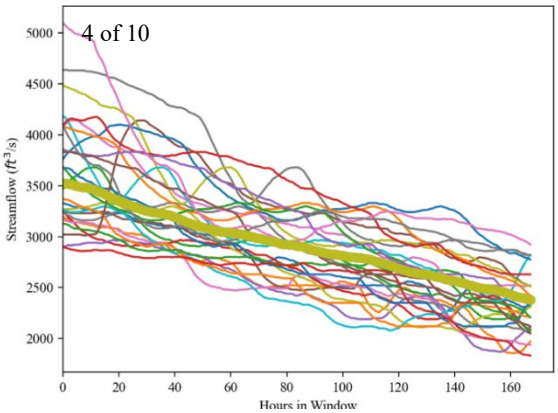
Outputs of the classification analyses for the test datasets are provided in this appendix. For each of the test datasets, the final output of the classification process are given with both fixed vertical axis range and varying vertical axis range. The former is intended to highlight the differences among the classification groups while the latter is intended to highlight the variability within an individual classification group.

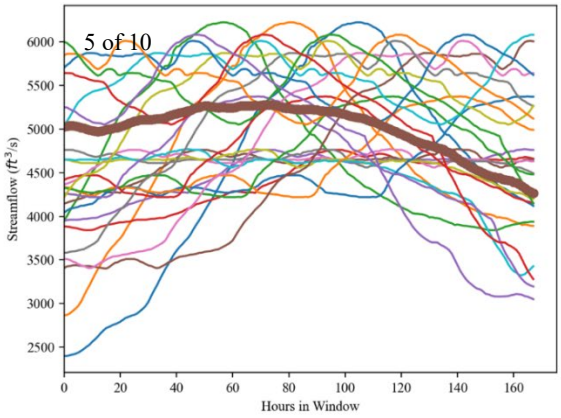
The figures illustrate the individual hydrographs assigned to the cluster as thin lines. The average SOM weight vector, obtained by averaging the weight vectors from the SOM nodes assigned to cluster by the mean shift algorithm, are given as a thicker set of dots that will often appear as a bold line across the timeseries. The clusters are given in no particular order within the figures.

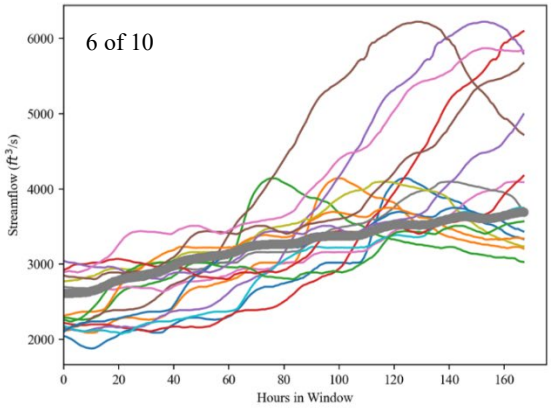

1 of 10
Streamflow (ft^3/s)
Hours in Window

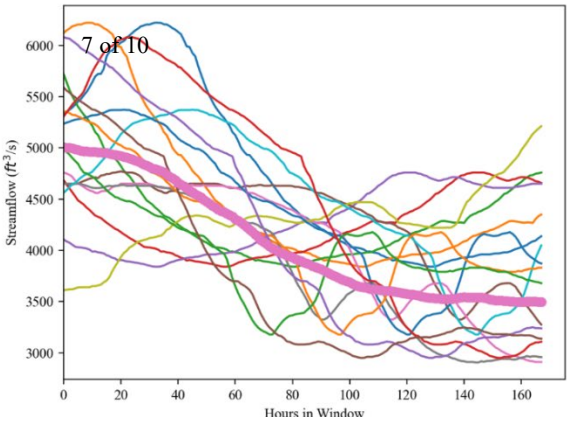

2 of 10
Streamflow (ft^3/s)
Hours in Window

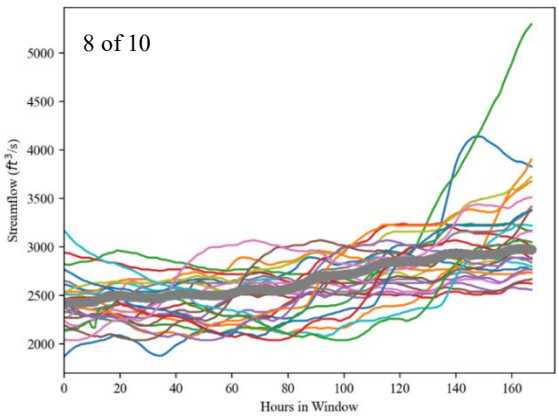

3 of 10
Streamflow (ft^3/s)
Hours in Window


4 of 10
Streamflow (ft^3/s)
Hours in Window


5 of 10
Streamflow (ft^3/s)
Hours in Window


6 of 10
Streamflow (ft^3/s)
Hours in Window


7 of 10
Streamflow (ft^3/s)
Hours in Window


8 of 10
Streamflow (ft^3/s)
Hours in Window

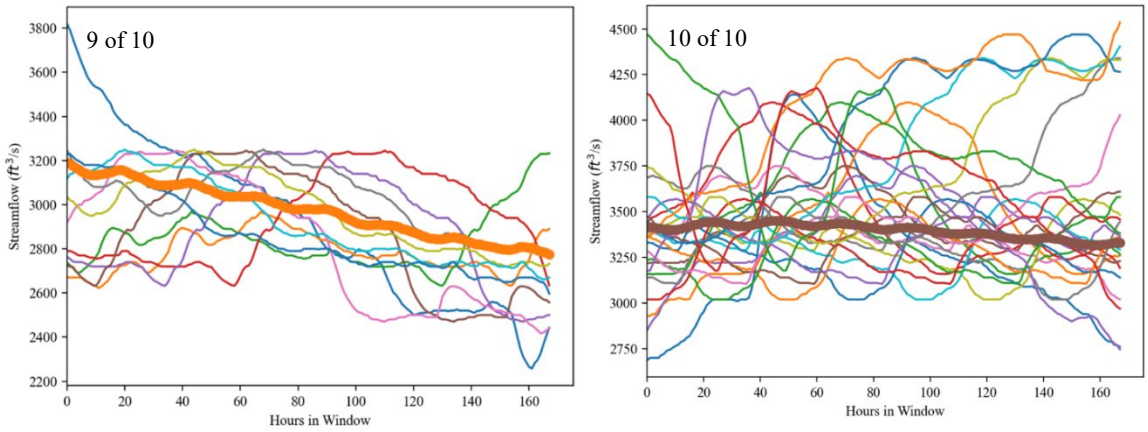
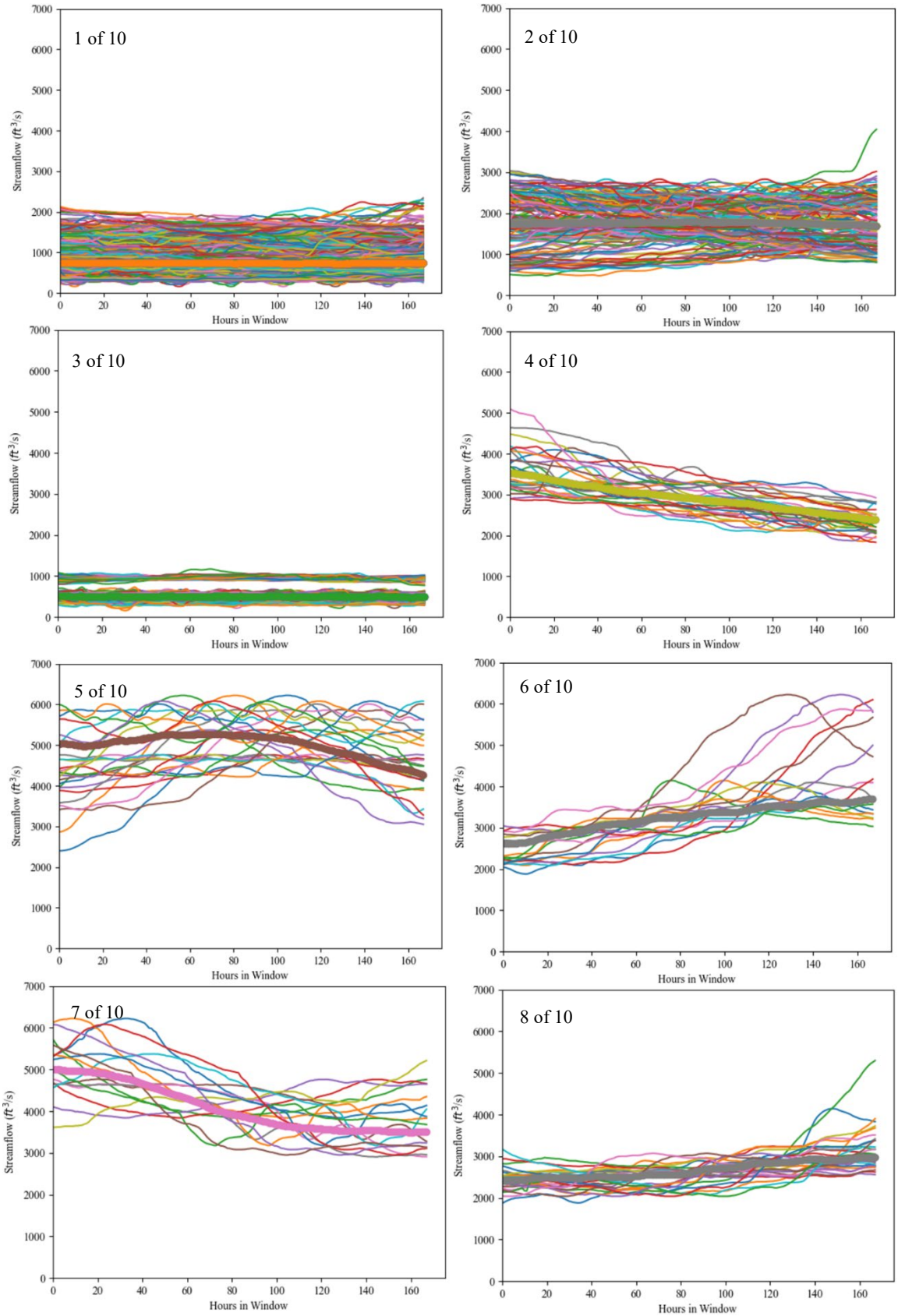


Figure B.1.—Cluster plots generated for USGS Colorado River data. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines. These plots do not have a fixed vertical range in order to highlight the differences in hydrograph characteristics between the clusters.


1 of 10
2 of 10
3 of 10
4 of 10
5 of 10
6 of 10
7 of 10
8 of 10
Streamflow (ft^3/s)
Hours in Window

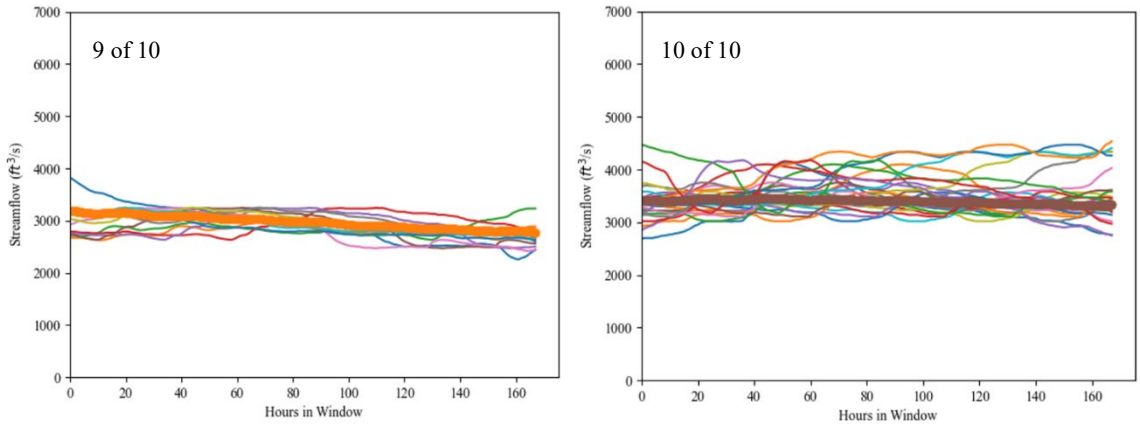


Figure B.2.—Cluster plots generated for USGS Colorado River data with fixed ranges to highlight differences in flow magnitude between the clusters. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines.

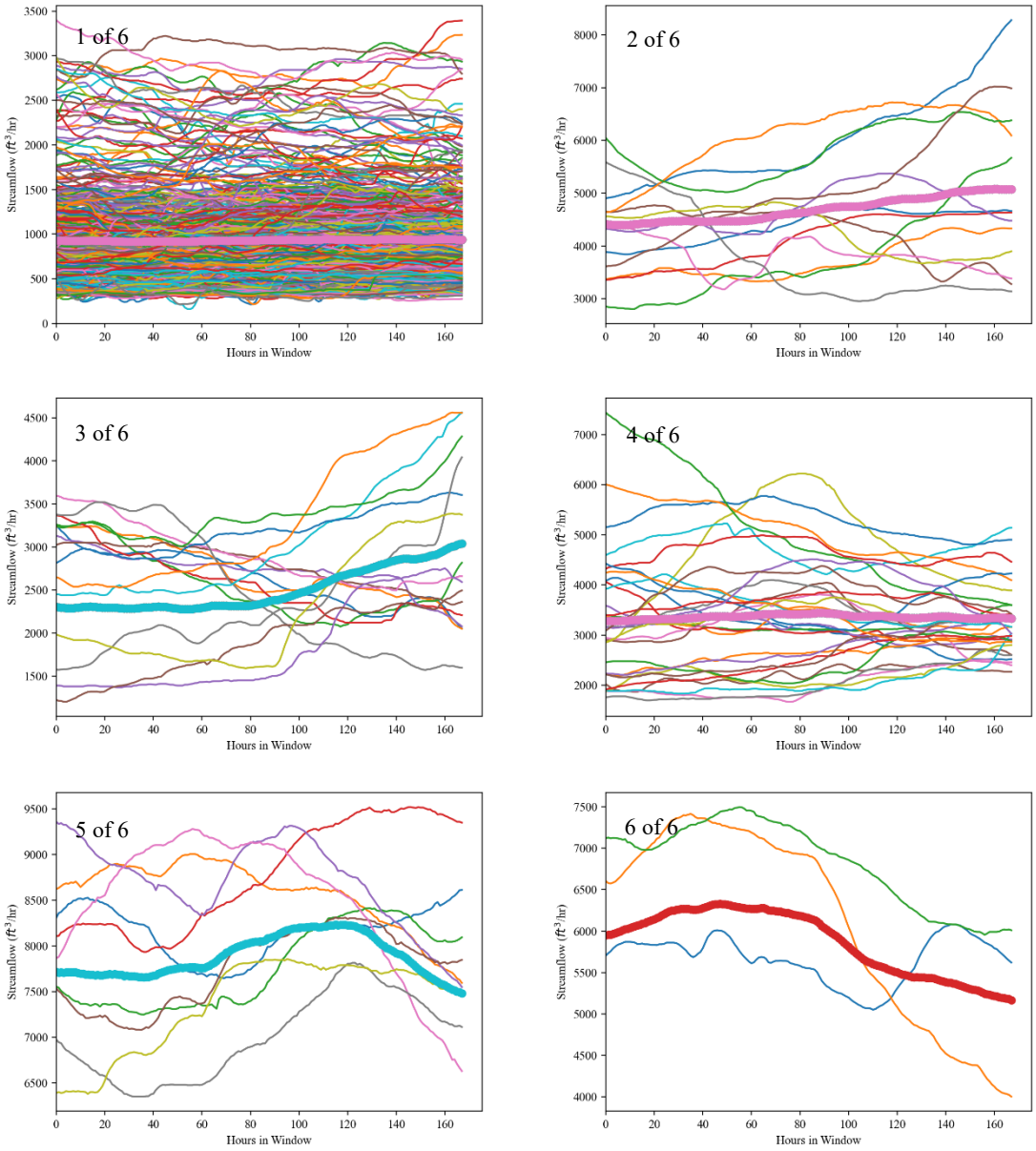


Figure B.3.—Cluster plots generated for USGS Colorado River data with a non-sliding window. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines. These plots do not have a fixed vertical range in order to highlight the differences in hydrograph characteristics between the clusters.

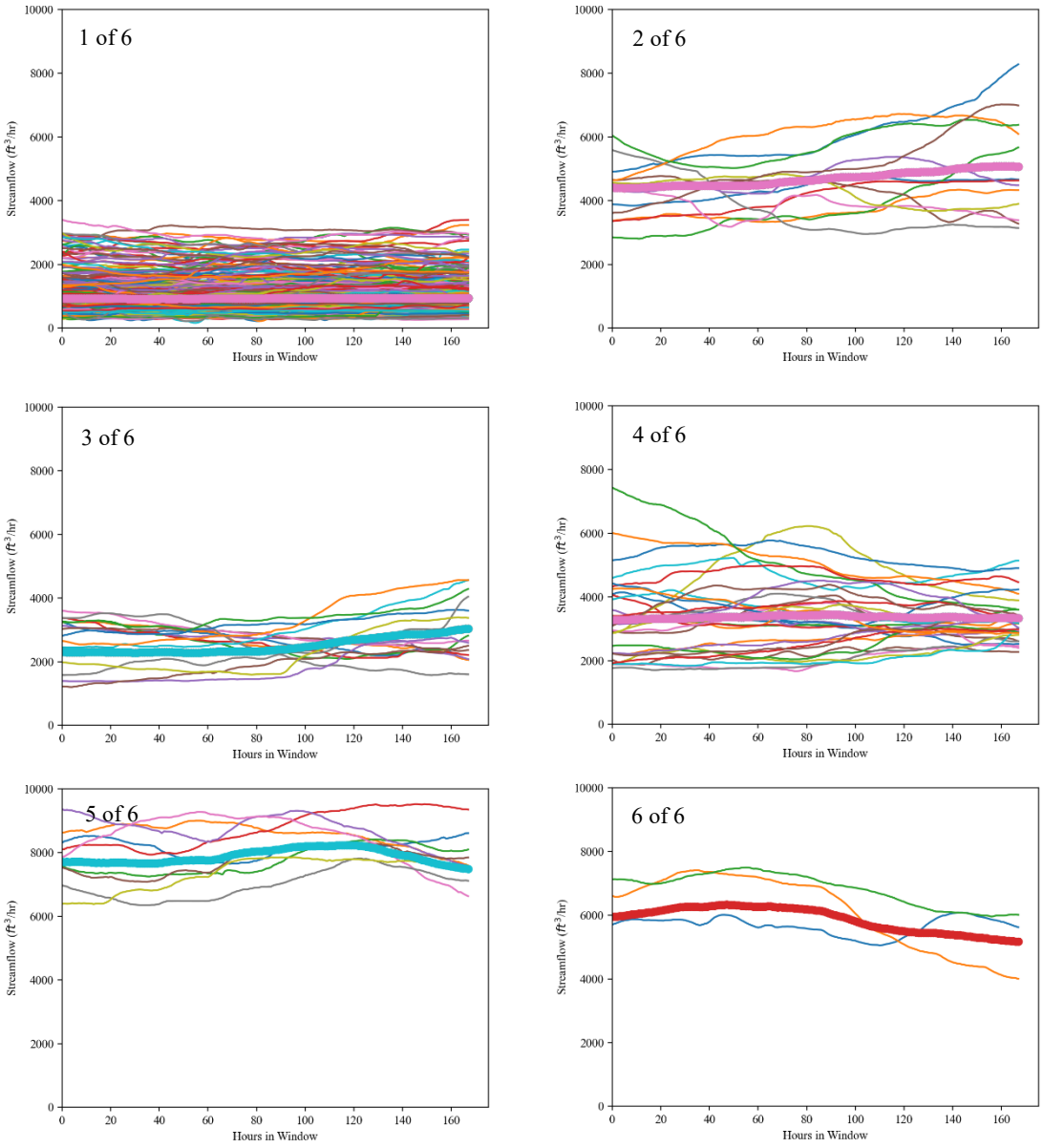
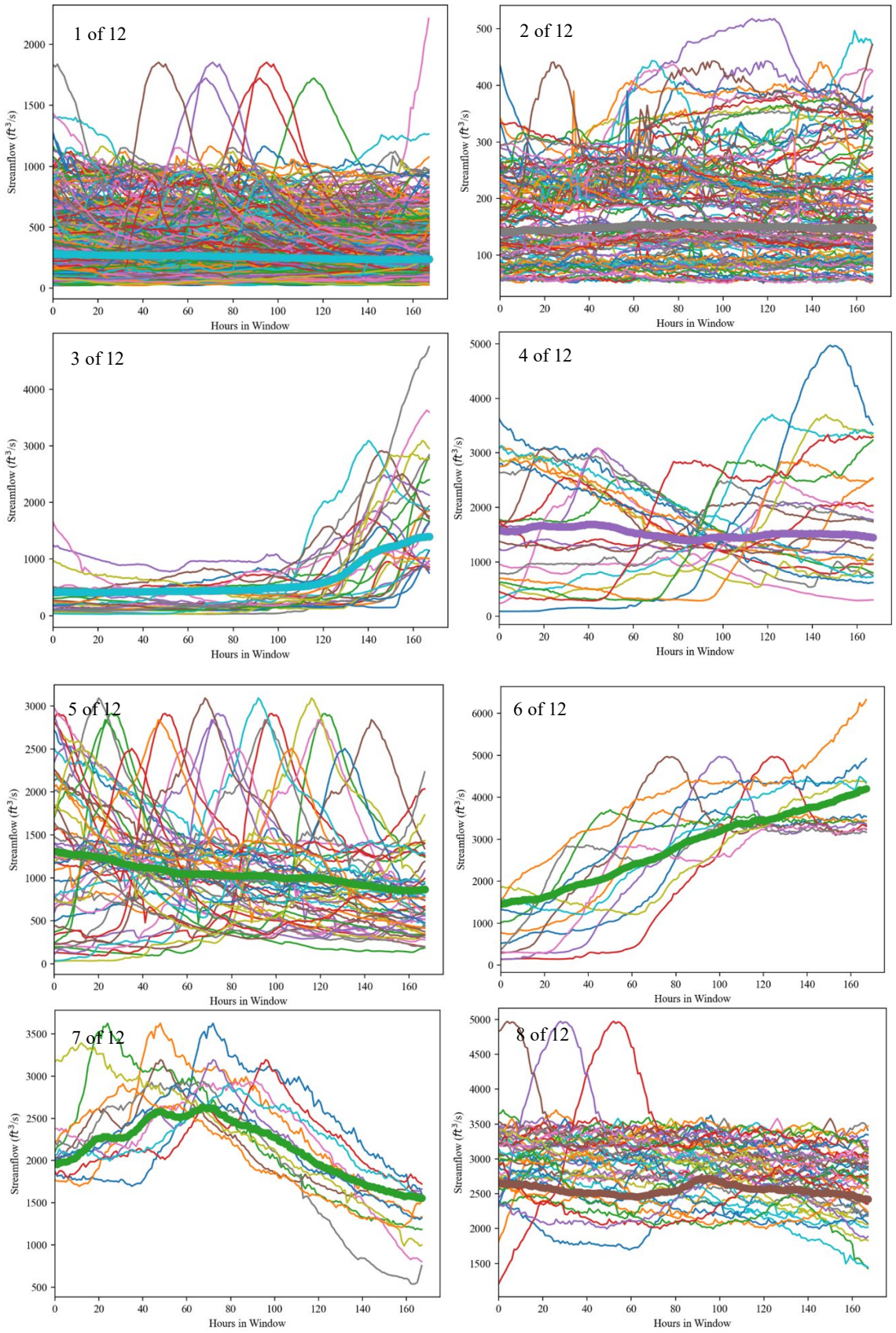


Figure B.4.—Cluster plots generated for USGS Colorado River data with a non-sliding window and fixed ranges to highlight differences in flow magnitude between the clusters. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines.


1 of 12
2 of 12
3 of 12
4 of 12
5 of 12
6 of 12
7 of 12
8 of 12
Streamflow (ft^3/s)
Hours in Window

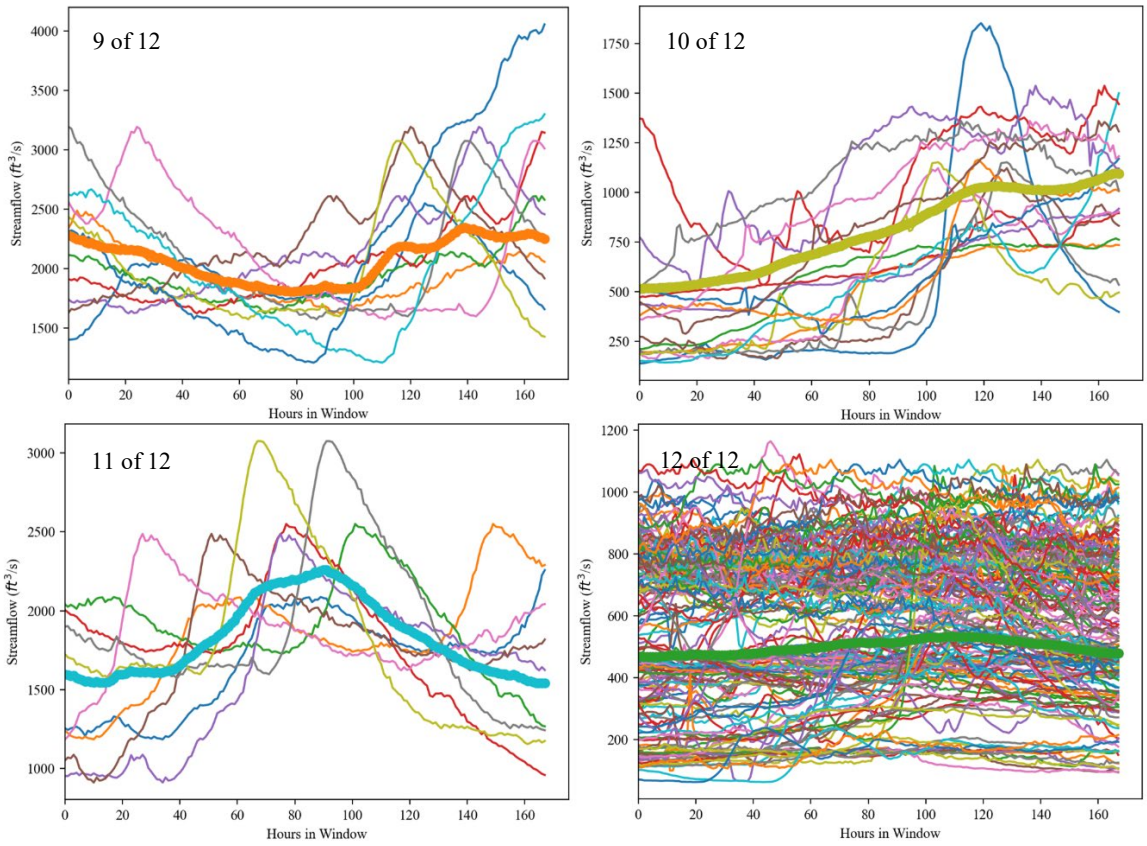
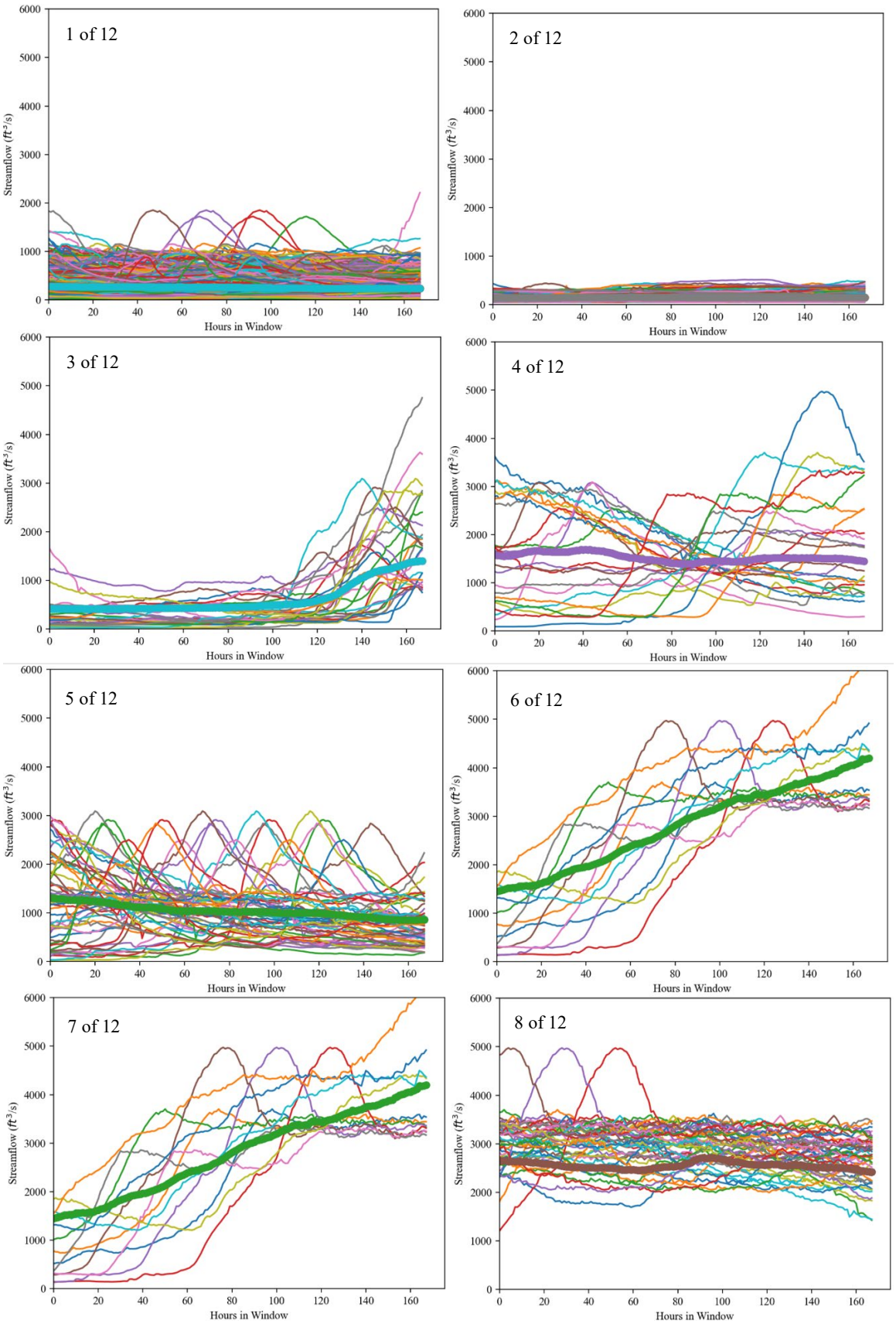


Figure B.5.—Cluster plots generated for USGS South Platte River data. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines. These plots do not have a fixed vertical range in order to highlight the differences in hydrograph characteristics between the clusters.


1 of 12
2 of 12
3 of 12
4 of 12
5 of 12
6 of 12
7 of 12
8 of 12
Streamflow (ft³/s)
Hours in Window

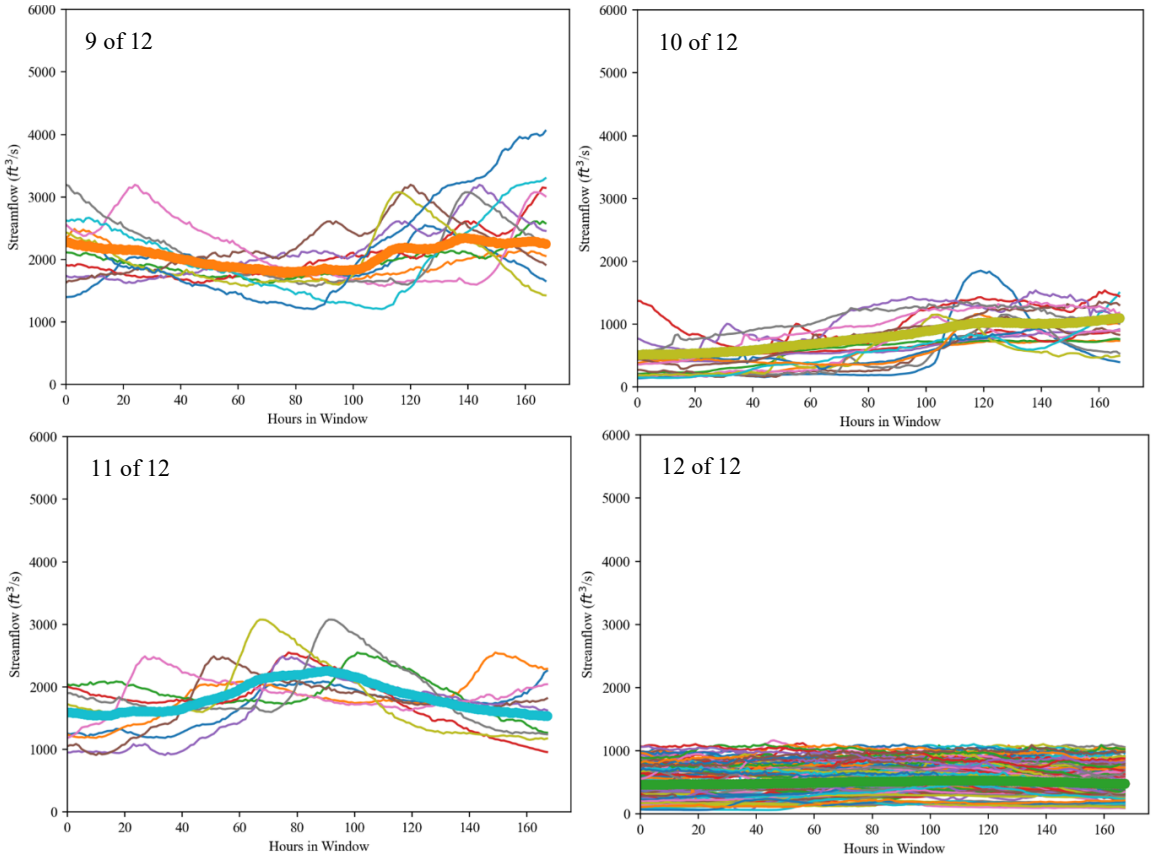


Figure B.6.—Cluster plots generated for USGS South Platte River data with fixed ranges to highlight differences in flow magnitude between the clusters. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines.

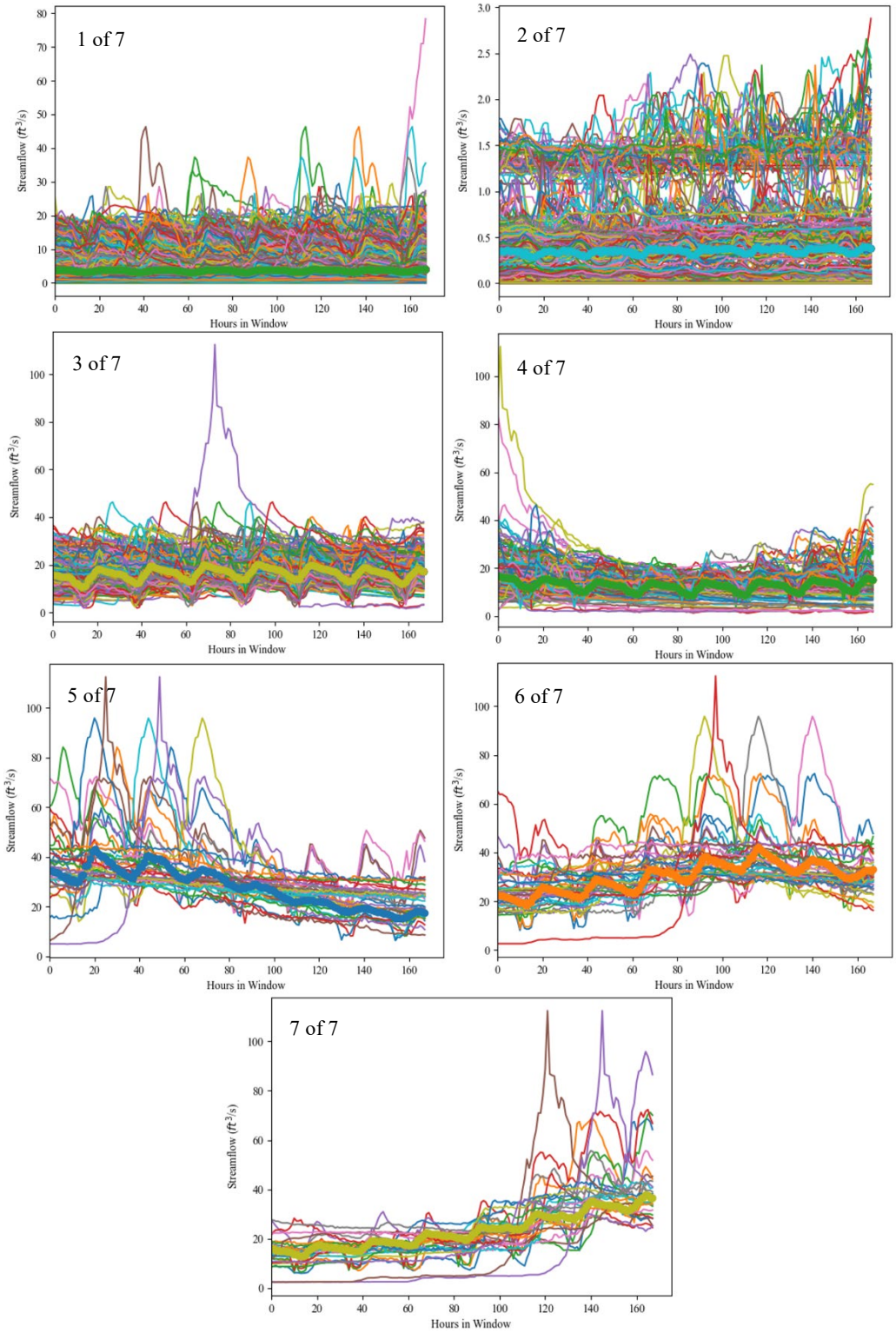


Figure B.7.—Cluster plots generated for Boise University dataset. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines. These plots do not have a fixed vertical range in order to highlight the differences in hydrograph characteristics between the clusters.

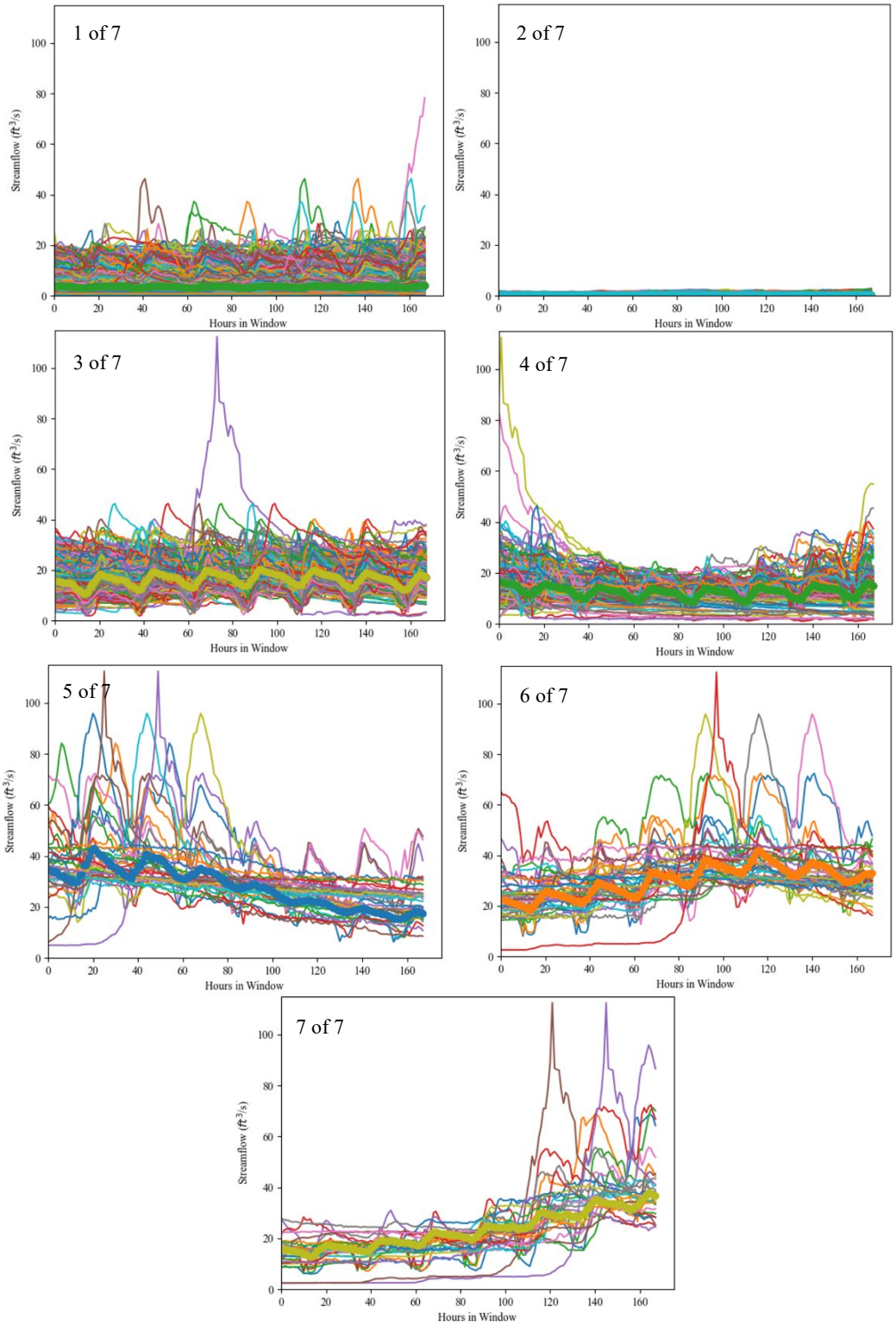
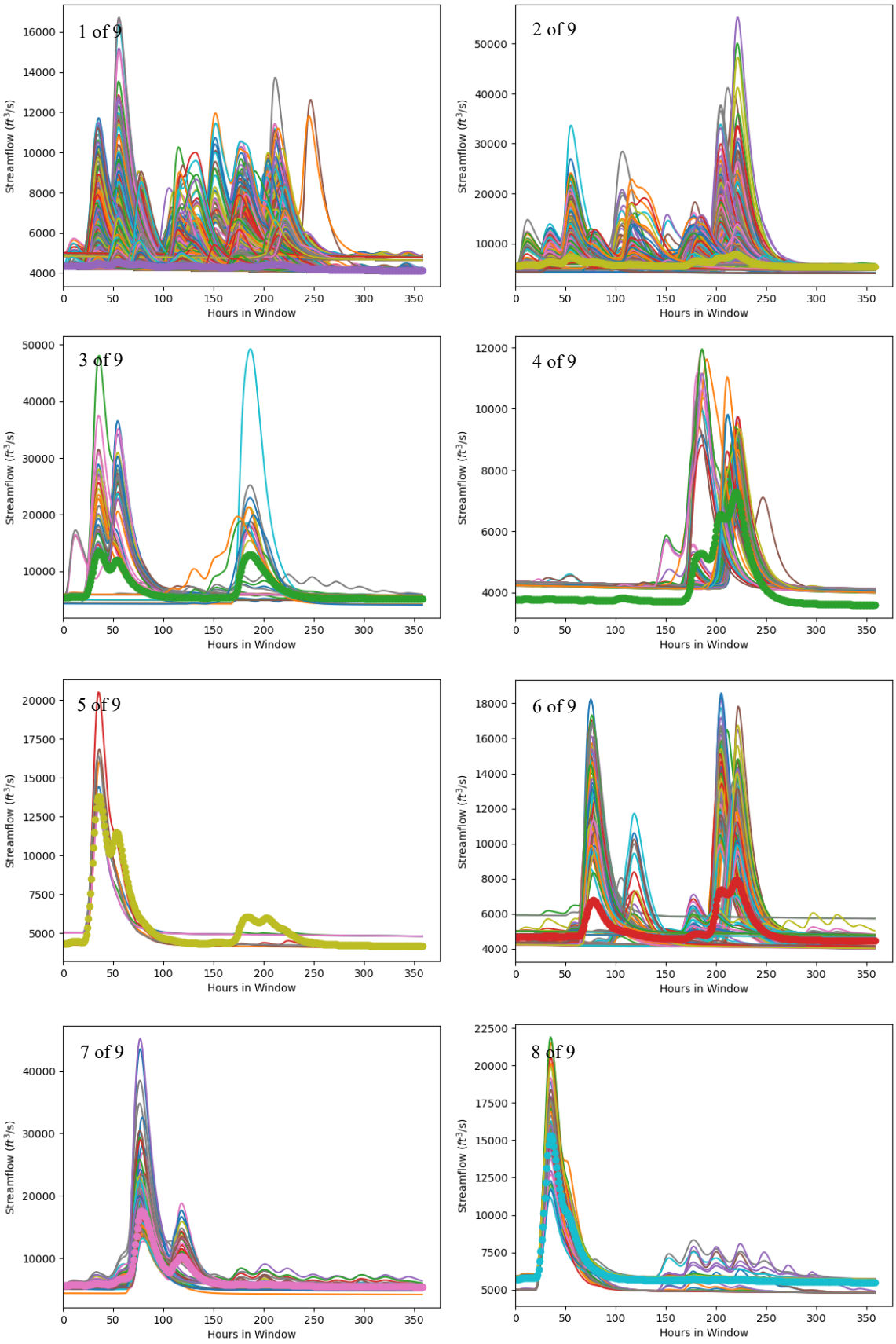


Figure B.8.—Cluster plots generated for Boise University data with fixed ranges to highlight differences in flow magnitude between the clusters. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines.


1 of 9
2 of 9
3 of 9
4 of 9
5 of 9
6 of 9
7 of 9
8 of 9
Streamflow (ft³/s)
Hours in Window

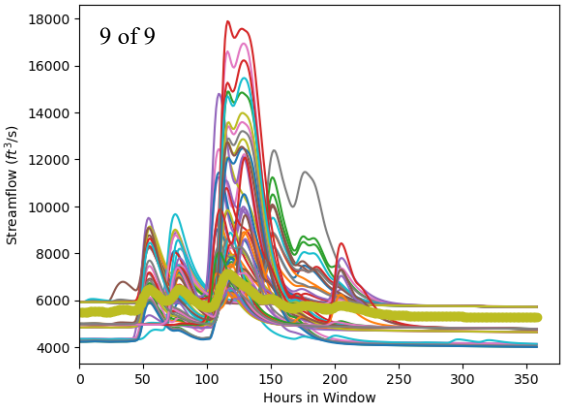
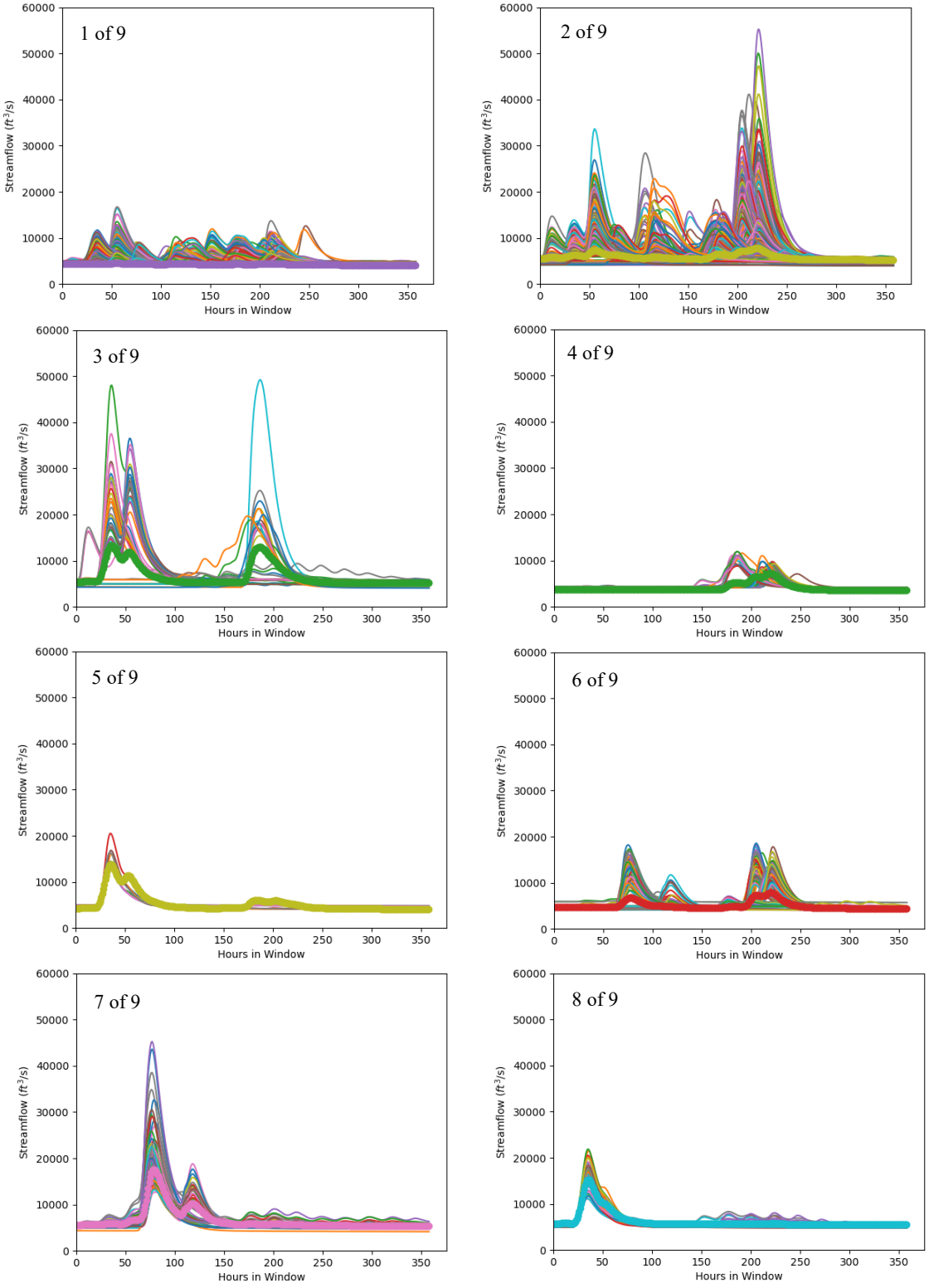


Figure B.9.—Cluster plots generated for SEFM El Vado data. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines. These plots do not have a fixed vertical range in order to highlight the differences in hydrograph characteristics between the clusters.


1 of 9
2 of 9
3 of 9
4 of 9
5 of 9
6 of 9
7 of 9
8 of 9
Streamflow (ft^3/s)
Hours in Window

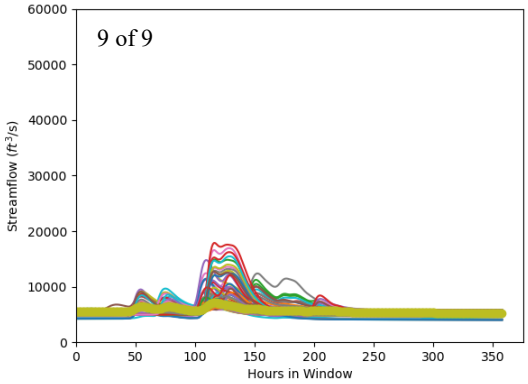


Figure B.10.—Cluster plots generated for SEFM El Vado data with fixed ranges to highlight differences in flow magnitude between the clusters. Hydrographs from each cluster are shown as thin lines and the average weight vector for all SOM cells in the cluster is shown as a thicker line over the hydrograph lines