Received: 19 March 2021 | Revised: 7 June 2021 | Accepted: 18 June 2021

DOI: 10.1002/hyp.14273

RESEARCH ARTICLE

An open web-based module developed to advance data-driven hydrologic process learning

Belize Lane[1] | Irene Garousi-Nejad[1] | Melissa A. Gallagher[2] | David G. Tarboton[1] | Emad Habib[3]

[1] Utah Water Research Laboratory, Utah State University, Logan, Utah, USA

[2] University of Houston, Houston, Texas, USA

[3] University of Louisiana at Lafayette, Louisiana, USA

Correspondence

Belize Lane, Utah Water Research Laboratory, Utah State University, Logan, UT, USA.
Email: belize.lane@usu.edu

Funding information

Utah State University Open Educational Resources Award; Utah Water Research Laboratory; Utah State University; Division of Undergraduate Education; National Science Foundation, Grant/Award Numbers: 1725989, 1726965

Abstract

The era of 'big data' promises to provide new hydrologic insights, and open web-based platforms are being developed and adopted by the hydrologic science community to harness these datasets and data services. This shift accompanies advances in hydrology education and the growth of web-based hydrology learning modules, but their capacity to utilize emerging open platforms and data services to enhance student learning through data-driven activities remains largely untapped. Given that generic equations may not easily translate into local or regional solutions, teaching students to explore how well models or equations work in particular settings or to answer specific problems using real data is essential. This article introduces an open web-based module developed to advance data-driven hydrologic process learning, targeting upper level undergraduate and early graduate students in hydrology and engineering. The module was developed and deployed on the HydroLearn open educational platform, which provides a formal pedagogical structure for developing effective problem-based learning activities. We found that data-driven learning activities utilizing collaborative open web platforms like CUAHSI HydroShare and JupyterHub to store and run computational notebooks allowed students to access and work with datasets for systems of personal interest and promoted critical evaluation of results and assumptions. Initial student feedback was generally positive, but also highlighted challenges including trouble-shooting and future-proofing difficulties and some resistance to programming and new software. Opportunities to further enhance hydrology learning include better articulating the benefits of coding and open web platforms upfront, incorporating additional user-support tools, and focusing methods and questions on implementing and adapting notebooks to explore fundamental processes rather than tools and syntax. The profound shift in the field of hydrology toward big data, open data services and reproducible research practices requires hydrology instructors to rethink traditional content delivery and focus instruction on harnessing these datasets and practices in the preparation of future hydrologists and engineers.

KEYWORDS
CUAHSI, data-driven learning, education, HydroLearn, HydroShare, JupyterHub, physical hydrology, web-based

1 | INTRODUCTION

Hydrologists investigate the distribution and variation of water across a range of spatial and time scales. In the face of mounting water resources challenges—due to a growing population, climate and land use change, and shifting societal values—hydrology has evolved from a mainly applied engineering discipline to a fundamental underpinning of geo and environmental sciences (Eagleson, 1991; National Research Council, 1991; Vogel et al., 2015; Wagener et al., 2007). As an applied and interdisciplinary science, hydrology benefits from first-hand knowledge gained by working with many different datasets. Generic equations are not easily translated into local or regional solutions, and experience with specific systems and datasets is critical for hydrologic practice and research. Such data-driven analysis is often needed to conceptualize complex processes and to explore how well models or equations work in particular settings or to answer a specific problem.

As demands on hydrologists have grown, so have calls to enhance hydrology education at the upper division and graduate levels to adequately prepare students for both research and industry (Merwade & Ruddell, 2012; Ruddell & Wagener, 2015; Wagener et al., 2021). Enhancing students' ability to conceptualize, analyze and interpret complex hydrologic processes is an area of much research (Bourget, 2006; Habib et al., 2018; Marshall et al., 2013; Merwade & Ruddell, 2012; Ngambeki et al., 2012; Ruddell & Wagener, 2015; Wagener et al., 2007, 2012). Educators have recognized a need to augment traditional teacher-centred lectures centred on fundamental physical laws with student-centred, data-driven learning activities that enable students to explore the hydrological system using authentic datasets and modelling tools (Merwade & Ruddell, 2012). Problem-based learning activities that include the use of authentic, real-world problems and datasets have been shown to enhance engineering and hydrology learning outcomes and career preparation (Gallagher et al., 2021; Habib et al., 2012, 2019; Litzinger et al., 2011; Lyon & Teutschbein, 2011; Merck et al., 2021; Sanchez et al., 2016). As a result, several web-based educational platforms that offer learning in an internet-based environment have been developed to incorporate real-world data and modelling resources in hydrology learning activities (e.g., SERC, CSDMS, COMET, HydroViz, RWater).

At the same time, the sheer volume and access to hydrologic data has grown rapidly through breakthroughs in remote sensing and in situ data collection and data services. “Big data” promises to provide new hydrologic insights to address mounting water resources challenges, and collaborative open web-based platforms are being increasingly developed and adopted by the global hydrologic science community to harness these datasets (Goodall et al., 2017; Slater et al., 2019). “Open” in this case implies that data and computational resources can be openly shared, discovered, and accessed among the community (Chen et al., 2020) while the underlying software may, in some cases, be commercial (i.e., not open-source). Open-source software by contrast is free to use, distribute, and modify. Open-source software provides unique opportunities in education for accessibility (Rajib et al., 2016) and in research for transparency and reproducibility as it reduces the financial and time costs for others to reproduce results (Rosenberg et al., 2020); however, it may not always have extensive technical support.

As in many fields, hydrology is trending toward a standardized open web-based structuring of data services, formats, and metadata to facilitate data management, analysis, and sharing needs. For example, the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) has developed an array of web-based data services and information systems specifically for the hydrologic science community (Goodall et al., 2017; Horsburgh et al., 2008; Horsburgh, Aufdenkampe, et al., 2016). Other open web-based platforms not specific to hydrology are also being increasingly adopted. For instance, GoogleColab, Google Earth Engine, and Jupyter Notebooks all allow users to create and share documents that contain live code, equations, visualizations, narrative text and link to web-based data services. Collaborative platforms like these provide convenient, standard workspaces and tools for the hydrology community, but they also demand that hydrologists and hydrology instructors keep pace with the rapid advancements.

With the promises of “big data” in hydrology come new challenges related to data management and reproducibility. Reproducibility is a critical requisite to advancing hydrologic discovery and innovation, and to subsequent integration and reuse of findings by the community (Choi et al., 2021; Essawy et al., 2020; Hutton et al., 2016; Stagge et al., 2019; Wilkinson et al., 2016). The complexity and diversity of hydrologic systems reflected in emerging data requires that scientists can reproduce methods developed in specific settings more broadly across a range of scales and locations to robustly evaluate hypotheses and assumptions (Ceola et al., 2015; Clark et al., 2016; Hutton et al., 2016). Particularly as datasets and models become more complex, analysis procedure and code need to be transparent and well-documented to allow for reproduction (Rosenberg et al., 2020; Stagge et al., 2019). The increasing use of open and open-source software by the hydrologic science community underpins these dual aims of accessibility and reproducibility.

The shift in data availability and analysis capabilities offered by open web-based platforms and the call for reproducible research have fundamentally transformed the role of hydrology instructors from disseminators of knowledge to guides in learning, critical thinking, and good research practices. However, these changes have not yet fully translated into changes in the education of future hydrologists. While educational platforms are emerging to support authentic, problem-based learning, as described above, they are mostly static and lack mechanisms for harnessing the emerging open data services and practices being adopted by the professional community. They also generally lack a formalized pedagogical structure to help instructors develop their own learning activities with these aims in mind. One exception is HydroLearn, a web-based collaborative hydrology education platform that provides a formalized and validated pedagogical structure—including tools to support instructors in creating learning objectives, formative assessment questions—to develop authentic, problem-based learning activities. Student learning of concepts and technical skills has been found to increase after using HydroLearn modules (Gallagher et al., 2021; Merck et al., 2021). However, HydroLearn's capacity to harness emerging open platforms and data services to enhance conceptual understanding in hydrology through data-driven learning remains largely untapped.

Advancing understanding in hydrological processes requires a workforce trained in working with data and learning from data, and learning platforms and modules designed to facilitate data-driven learning have the potential to change the way hydrologists do research that advances hydrological processes. This article describes a HydroLearn physical hydrology learning module targeting advanced undergraduate and early graduate students in hydrology and engineering. The aims of the learning module are (1) to harness emerging open web-based platforms in order to (2) develop data-driven learning skills whereby students actively explore key concepts using real data that is relevant and meaningful to them thereby (3) enhancing student learning of fundamental hydrology concepts while (4) providing experience applying good data management and reproducibility practices. The article briefly describes several open web-based platforms for hydrology and their potential educational utility, introduces the learning module including integration of these platforms, and offers initial student perceptions and instructor reflections on the module.

2 | OPEN WEB-BASED PLATFORMS AND PROGRAMMING PACKAGES FOR HYDROLOGIC ANALYSIS

Collaborative open web-based platforms and tools are being increasingly adopted by the hydrologic science community. A comprehensive review of available resources is beyond the scope of this article. Instead, here we briefly summarize the platforms and programming packages utilized in the HydroLearn physical hydrology learning module and their potential educational utility, including CUAHSI HydroShare and JupyterHub, and ESRI Story Maps.

2.1 | HydroShare

HydroShare is a web-based collaborative platform for hydrology data storage, retrieval, sharing, and processing (Essawy et al., 2020; Horsburgh, Morsy, et al., 2016; Tarboton et al., 2014). Hydrology instructors and students increasingly use HydroShare to access free cloud-based versions of several software programs and hydrologic models and use them for various research and learning applications, or to access previously uploaded static teaching resources (Ward et al., 2020).

2.2 | JupyterHub

CUAHSI JupyterHub is an open cloud-based environment for computational notebooks that allows users to create and share documents that contain live code, equations, visualizations and narrative text (Choi et al., 2021). Jupyter notebooks (https://jupyter.org/) are used to write, build, and run codes as well as run pre-installed software (e.g., TauDEM, Tesfa et al., 2011; Tarboton, 2018), but can also be used as teaching tools to build programming and data management skills.

2.3 | ESRI Story Maps

Finally, ESRI Story Maps combine narrative text with immersive content that fills the screen with maps, images, or videos for an engaging learning experience. While the code for ESRI Story Maps is not open-source, these cloud-accessed resources harness ArcGIS's analysis tools and GIS platforms, and can be hosted and made publicly available directly through ArcGIS Online. Story Maps allow students to directly interact with data through a personalized hands-on experience (e.g., Kerski, 2019). Alternatively, students can be assigned to create their own Story Maps to dynamically communicate project results (e.g., Battersby & Remington, 2013).

2.4 | Programming packages

In addition to the platforms described above, numerous programming packages can be used in learning activities to facilitate hydrologic data retrieval (e.g., *DataRetrieval*; *rnrfa)*, analysis (e.g., *TauDEM*, Tarboton, 2018), modelling (e.g., Choi et al., 2021), and visualization. For example, the R *waterData* package allows a user to import daily hydrological data from the United States Geological Survey (USGS) web services and plot time-series data (R Core Team, 2020; Ryberg & Vecchia, 2012). A detailed description of R packages relevant for hydrologic analysis is provided in Slater et al. (2019).

The integration of open web-based platforms and programming packages allows engineering and hydrology students to use authentic data to make sense of the concepts they are learning in their courses, while learning about the data and tools that are openly available.

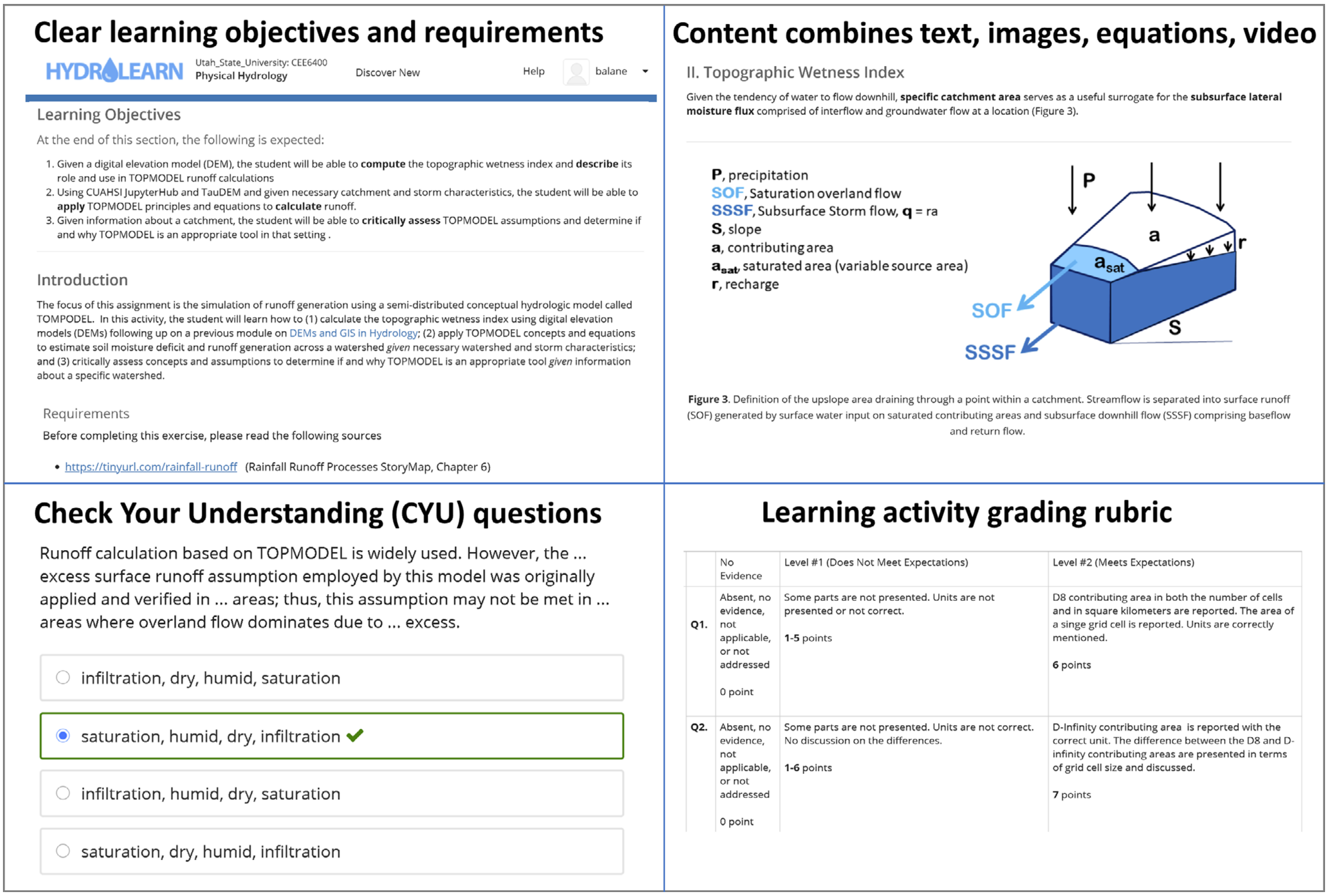
3 | HYDROLEARN PHYSICAL HYDROLOGY LEARNING MODULE

HydroLearn is itself an open web-based platform that aims to help hydrology instructors develop, share, and adapt learning modules. It combines research-based active learning methods with authentic online learning modules. The modular nature of HydroLearn and the dynamic computational notebooks allow instructors to use, combine, or adapt content, datasets and scripts from existing learning modules to their specific instructional needs and geographic settings. Active learning is supported through the ability to embed video- and image-based content, questions, other websites, and learning activity templates (Figure 1). Common elements of HydroLearn modules include Check-Your-Understanding (CYU) questions, quantitative problems, and authentic learning activities. CYU question formats include multiple choice, checkbox, drag-and-drop questions, and open response to higher-level questions related to process interpretation. By contrast, authentic tasks are high cognitive-demand tasks built to reflect how knowledge is used in real life and to simulate the type of problems that a professional might tackle. Each learning activity has a grading rubric, an assessment tool intended to set clear expectations for students and make grading more objective. The platform provides wizards and templates to help instructors develop strong learning objectives and align the teaching activities, learning outcomes, and assessments, a process referred to in the learning literature as constructive alignment (Biggs & Tang, 2011; Kandlbinder, 2014).

The HydroLearn Physical Hydrology learning module incorporates the above elements and consists of six sections: (M1) data analysis and statistics in hydrology, (M2) geographical information systems in hydrology, (M3) runoff generation, (M4) water in the soil, (M5) infiltration modelling, and (M6) calculating runoff using TOPMODEL concepts. Below we briefly discuss Sections M1, M2, M3, and M6 of the module to highlight the use of active and authentic learning, open web-based platforms, and data-driven learning skills. Sections M4 and M5 of the module are not covered here because they are similar to other sections in format and learning elements used. The entire module, including these sections is available online (Lane & Garousi Nejad, 2018). Table 1 lists key learning objectives, learning activities, open web-based platforms and data sources for each section.

3.1 | Section M1: Data analysis and statistics in hydrology

The first section of the learning module addresses all four aims outlined in the introduction. Students are introduced to fundamental concepts in hydrology while learning basic data analysis and management



Clear learning objectives and requirements

Utah_State_University: CEE6400
Physical Hydrology

Learning Objectives

At the end of this section, the following is expected:

1. Given a digital elevation model (DEM), the student will be able to **compute** the topographic wetness index and **describe** its role and use in TOPMODEL runoff calculations
2. Using CUAHSI JupyterHub and TauDEM and given necessary catchment and storm characteristics, the student will be able to **apply** TOPMODEL principles and equations to **calculate** runoff.
3. Given information about a catchment, the student will be able to **critically assess** TOPMODEL assumptions and determine if and why TOPMODEL is an appropriate tool in that setting .

Introduction

The focus of this assignment is the simulation of runoff generation using a semi-distributed conceptual hydrologic model called TOMPODEL. In this activity, the student will learn how to (1) calculate the topographic wetness index using digital elevation models (DEMs) following up on a previous module on DEMs and GIS in Hydrology; (2) apply TOPMODEL concepts and equations to estimate soil moisture deficit and runoff generation across a watershed *given* necessary watershed and storm characteristics; and (3) critically assess concepts and assumptions to determine if and why TOPMODEL is an appropriate tool *given* information about a specific watershed.

Requirements

Before completing this exercise, please read the following sources

- https://tinyurl.com/rainfall-runoff (Rainfall Runoff Processes StoryMap, Chapter 6)

Content combines text, images, equations, video

II. Topographic Wetness Index

Given the tendency of water to flow downhill, **specific catchment area** serves as a useful surrogate for the **subsurface lateral moisture flux** comprised of interflow and groundwater flow at a location (Figure 3).

P, precipitation
SOF, Saturation overland flow
SSSF, Subsurface Storm flow, **q** = ra
S, slope
a, contributing area
a$_{sat}$, saturated area (variable source area)
r, recharge

Figure 3. Definition of the upslope area draining through a point within a catchment. Streamflow is separated into surface runoff (SOF) generated by surface water input on saturated contributing areas and subsurface downhill flow (SSSF) comprising baseflow and return flow.

Check Your Understanding (CYU) questions

Runoff calculation based on TOPMODEL is widely used. However, the ... excess surface runoff assumption employed by this model was originally applied and verified in ... areas; thus, this assumption may not be met in ... areas where overland flow dominates due to ... excess.

○ infiltration, dry, humid, saturation

● saturation, humid, dry, infiltration ✓

○ infiltration, humid, dry, saturation

○ saturation, dry, humid, infiltration

Learning activity grading rubric

	No Evidence	Level #1 (Does Not Meet Expectations)	Level #2 (Meets Expectations)
Q1.	Absent, no evidence, not applicable, or not addressed 0 point	Some parts are not presented. Units are not presented or not correct. **1-5** points	D8 contributing area in both the number of cells and in square kilometers are reported. The area of a singe grid cell is reported. Units are correctly mentioned. **6** points
Q2.	Absent, no evidence, not applicable, or not addressed 0 point	Some parts are not presented. Units are not correct. No discussion on the differences. **1-6** points	D-Infinity contributing area is reported with the correct unit. The difference between the D8 and D-infinity contributing areas are presented in terms of grid cell size and discussed. **7** points

FIGURE 1 Key components of HydroLearn learning modules include: Clear learning objectives and requirements (top-left, then clockwise), content combines multiple media, learning activity grading rubrics, and check-your-understanding (CYU) questions

TABLE 1 Detailed chart of key sections in the HydroLearn physical hydrology learning module, including learning objectives, learning activities, and open web-based platforms and datasets

Module section	Learning objectives (the student will be able to…)	Learning activities	Open web-based platforms and datasets
(M1) Data analysis and statistics	Calculate water storage, fluxes, and uncertainty in components of the hydrologic cycle	Problems: water balance, uncertainty	ESRI story map-introduction to physical hydrology
	Navigate public websites to extract key hydrologic information	Problems: streamflow time series analysis	StreamStats; USGS NWIS
	Perform basic hydrologic data analysis for a watershed of interest	Authentic task: for user-selected stream gage: (i) describe watershed attributes, (ii) retrieve streamflow data, (iii) perform statistical analyses including daily and seasonal plots, flow duration curve, exceedance, etc.	Jupyter notebook (R), including packages: data retrieval, zoo, ggplot
	Assess and interpret hydrologic trends in the context of a specific watershed	CYU: multiple choice and open response	
(M2) Geographical information systems	Derive hydrologically useful information from digital elevation models (OEMs)	Problems, CYU: open response	USGS National Elevation Dataset
	Describe the sequence of steps involved in mapping stream networks, catchments, and watersheds from DEMs	Problems: basic terrain analysis for hydrologic research	Jupyter notebook (Python), including packages: TauDEM, gdal, geopandas, rasterio, rasterstats
	Compute an approximate water balance for a watershed using open data	Problems: water balance, runoff ratio	
(M3) Runoff generation	Use appropriate terms to describe the processes involved in runoff generation	CYU: concepts and definitions of runoff generation mechanisms	ESRI story map- rainfall-runoff processes
	Differentiate between runoff generation mechanism and when and where each is likely to occur	Problems: multiple choice	
	Justify why and how specific changes in physical and climate attributes will influence dominant runoff mechanism and storm hydrograph shape	CYU: open response	
(M6) Calculating runoff using TOPMODEL	Compute the topographic wetness index and describe its role and use in runoff calculations, given a watershed DEM	CYU: topographic wetness index, variable source area, hydrologic models	Jupyter notebook (Python), StoryMap
	Apply TOPMODEL principles and equations to calculate runoff given catchment and storm characteristics	Authentic learning activity: simulate runoff using a semi-distributed hydrologic model	
	Critically assess assumptions and determine if and why model is appropriate, given catchment and storm characteristics	Authentic learning activity: estimate runoff across a watershed	

Note: Module sections M4 and M5 are not included since they are similar to other sections.
Abbreviation: CYU, check-your-understanding questions.

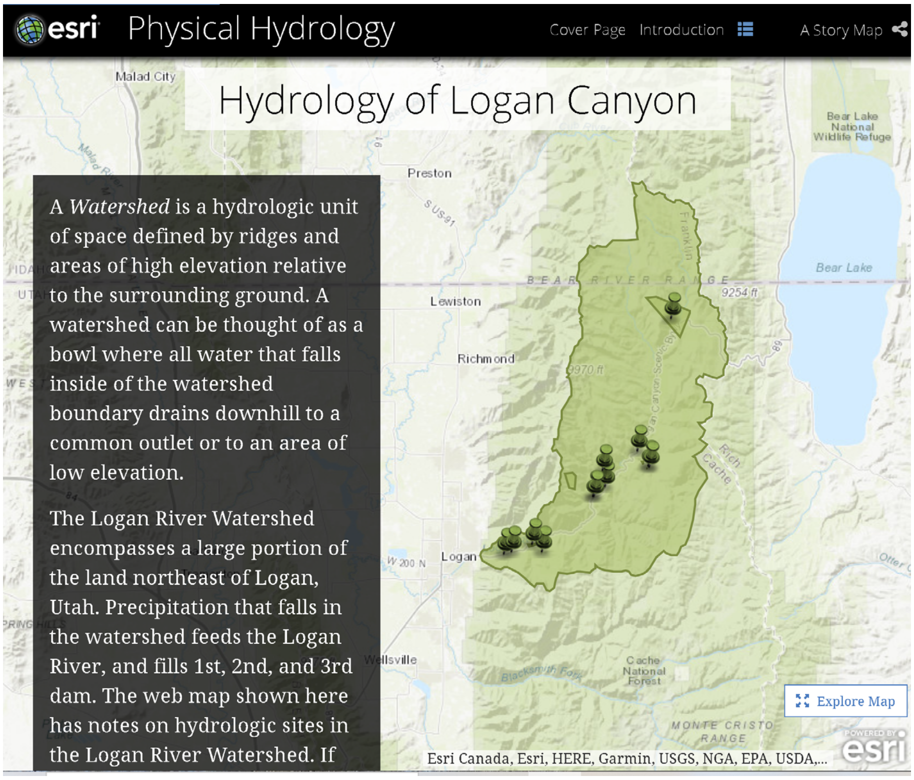
skills through a set of problems and an authentic learning activity. Key terms and concepts are introduced using an ESRI Story Map. The problems and authentic learning activity are performed in a Jupyter notebook accessed through HydroLearn. Following the established HydroLearn structure, the section starts by delineating the learning objectives and provides key background information, a detailed grading rubric, learning activity instructions and summary questions (Figure 1).

Through a link at the beginning of the first module section (Data Analysis and Statistics in Hydrology, M1), students are first directed to the linked standalone ESRI story map *Introduction to Physical Hydrology* (Figure 2) that provides a map-based virtual tour of watershed hydrology in the Logan River watershed, Utah, USA. This Story Map describes key components of the water cycle (e.g., precipitation, evaporation, runoff) and provides place-based examples and illustrative images and figures to help students connect with the landscape and concepts personally. For example, the Story Map shows inset images of wet soil linked to locations on the watershed map and states that "infiltration is the process by which water on the Earth's surface enters the soil… Explore the map to find examples of infiltration in the watershed." As part of the development process, open repositories of videos and images of these water cycle components taken by the authors and from existing open web products were compiled to make this media available to students and other educators.

Several quantitative problems ask students to make basic calculations related to water storage, fluxes, and uncertainty in key components of the hydrologic cycle. These calculations are performed in a Jupyter notebook stored in a HydroShare resource embedded within the learning module for easy access (Figure 3). Students are required to create an account in HydroShare in order to run the notebook. The dynamic notebook uses the *R* programming language and is intended for students who have had no or limited programming experience. The packages and code needed to perform the calculations are provided and well notated to familiarize students with basic programming notation and key functions. This prepares them for the next section in which they are asked to modify the code slightly to use a different dataset that they select. The notebook provides a gentle and context-based introduction to R programming recognizing that learners without programming knowledge are more likely to be interested and see its value when it is applied in the context of an authentic problem (Kalelioğlu, 2015).

In one problem, students first estimate long-term average evapotranspiration rates for several watersheds using a simple water balance model, and then compute the 95% relative and absolute uncertainties in these estimates. The Jupyter notebook begins with text descriptions and equations, followed by a section with scripts needed to perform the calculations. Students must run the script to generate results, and are then prompted to add a new section and switch from calculation to text response to describe their results. They are then asked to check their understanding by comparing and contrasting the water balance and uncertainty results across watersheds in their own words. Specifically, they are prompted to indicate what catchment and climate conditions may be influencing the long-term water balances of these watersheds and then describe how specific hydrologic processes may be affected by these conditions.

In the authentic learning task, students identify a USGS stream gage of interest and work through several steps to explore the streamflow patterns and statistics at that location, including to: obtain streamflow data, delineate the upstream catchment, generate time series plots and calculate summary statistics, interpret the hydrological behaviour of the river using a flow duration curve, evaluate seasonality trends, and use histograms and probability distributions to

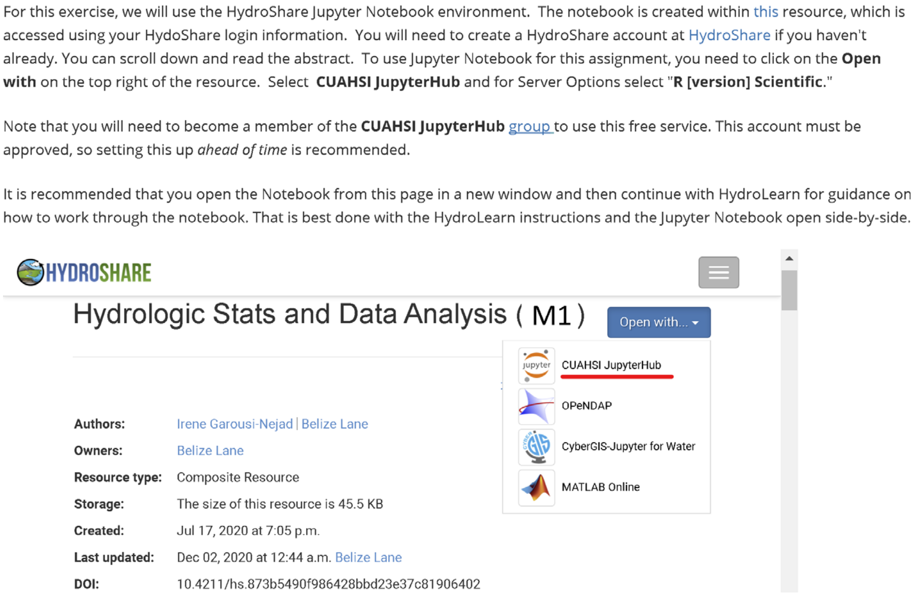


Hydrology of Logan Canyon

A *Watershed* is a hydrologic unit of space defined by ridges and areas of high elevation relative to the surrounding ground. A watershed can be thought of as a bowl where all water that falls inside of the watershed boundary drains downhill to a common outlet or to an area of low elevation.

The Logan River Watershed encompasses a large portion of the land northeast of Logan, Utah. Precipitation that falls in the watershed feeds the Logan River, and fills 1st, 2nd, and 3rd dam. The web map shown here has notes on hydrologic sites in the Logan River Watershed. If

FIGURE 2 An ESRI story map introduces key hydrologic terms and processes to students in an interactive, map-based interface using photos and videos from the local watershed of the developer's university



For this exercise, we will use the HydroShare Jupyter Notebook environment. The notebook is created within this resource, which is accessed using your HydoShare login information. You will need to create a HydroShare account at HydroShare if you haven't already. You can scroll down and read the abstract. To use Jupyter Notebook for this assignment, you need to click on the **Open with** on the top right of the resource. Select **CUAHSI JupyterHub** and for Server Options select "**R [version] Scientific.**"

Note that you will need to become a member of the **CUAHSI JupyterHub** group to use this free service. This account must be approved, so setting this up *ahead of time* is recommended.

It is recommended that you open the Notebook from this page in a new window and then continue with HydroLearn for guidance on how to work through the notebook. That is best done with the HydroLearn instructions and the Jupyter Notebook open side-by-side.

Hydrologic Stats and Data Analysis (M1)

Authors:	Irene Garousi-Nejad | Belize Lane
Owners:	Belize Lane
Resource type:	Composite Resource
Storage:	The size of this resource is 45.5 KB
Created:	Jul 17, 2020 at 7:05 p.m.
Last updated:	Dec 02, 2020 at 12:44 a.m. Belize Lane
DOI:	10.4211/hs.873b5490f986428bbd23e37c81906402

FIGURE 3 The CUAHSI JupyterHub computational Jupyter notebook used in section M1 of the HydroLearn module is stored in an embedded HydroShare resource for easy access

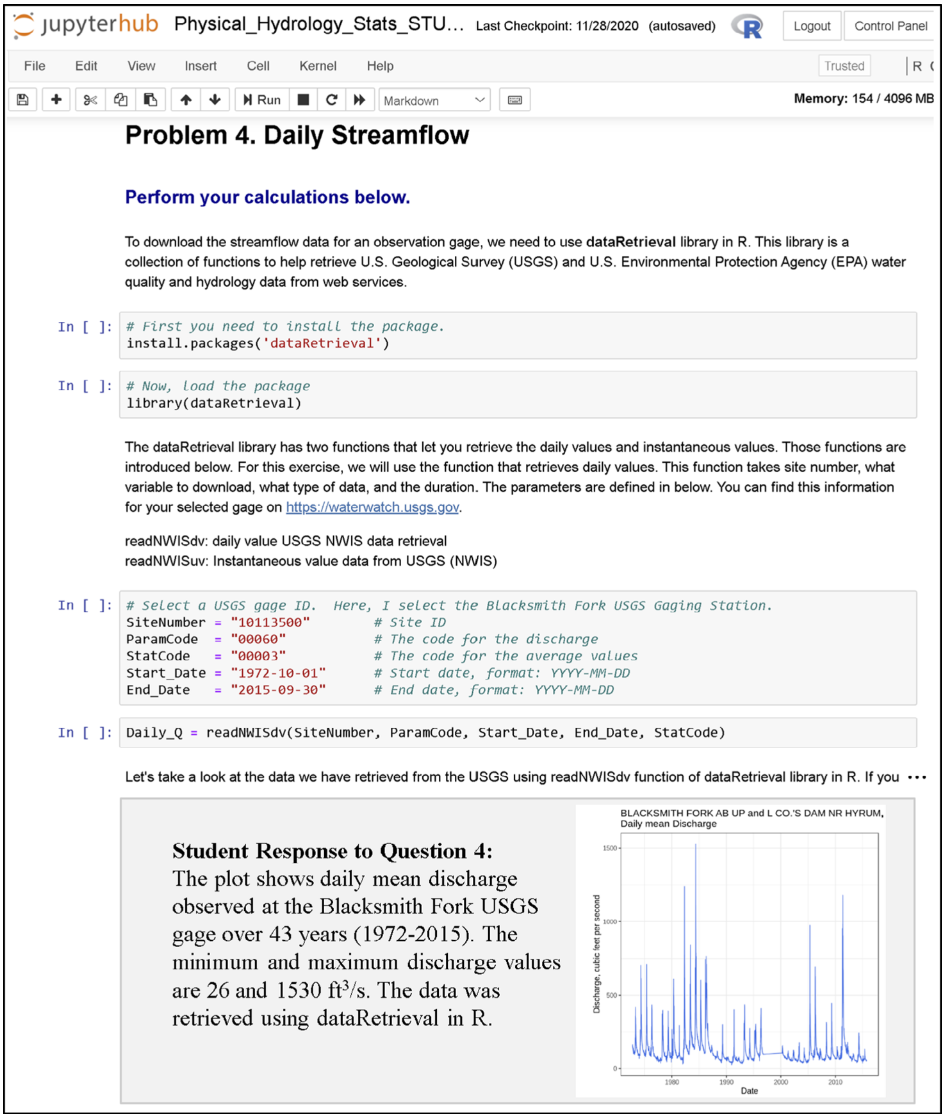
describe characteristics of the flow (Figure 4). For this activity, the Jupyter notebook walks students through each step in detail, paralleling background information and summary questions in the HydroLearn module. Complete code is provided but a few parameters (e.g., stream gage ID, start date) must be adjusted by the students to customize the script to their specific dataset. First, they are provided with a tutorial video in the learning module demonstrating how to select a stream gage, delineate a watershed, and describe basin characteristics. Then, they are directed how to use the *dataRetrieval* package in R to bring streamflow data from the USGS National Water Information System website into their cloud-based workspace for subsequent analysis without the need to download data. Finally, they explore plotting and visualization tools to generate a flow duration curve and a single plot including annual peak flow, average flow, and seven-day minimum flow. Students are also shown how to manage streamflow data using data-frames, the definition and use of Water Years (October to September in the U.S.), and how to aggregate data by year or month to calculate summary statistics (e.g., average June flow). The section concludes with CYU multiple choice questions to reinforce key concepts. Having students select a USGS stream gage of interest to perform these analyses provides students with experience in reproducible research practices and opens up opportunities for students to evaluate and compare hydrological responses in distinct climate and catchment settings. It also allows students to assess results in the context of a system they are familiar with to promote conceptual understanding.

3.2 | Section M2: Digital elevation models and GIS in hydrology

This section introduces the use of geospatial processing tools and basic terrain analysis to derive hydrologically useful information from digital elevation models (DEMs) for an example watershed, including watershed delineation and stream network generation. Similar to section M1, HydroLearn links to a Jupyter notebook that is used to complete this data-driven learning activity. Here, the notebook employs the Python programming language so students can explore the distinct utility of Python-based spatial analysis packages and visualization tools. The full set of learning activities in section M2 can be adapted for another watershed, and provide students with experience and training in open web-based tools and reproducible research practices. The specific activities are summarized below to illustrate the authentic, data-driven approach.

3.2.1 | Preparation, libraries, and getting oriented

Key concepts and input files are presented, followed by an overview of Python libraries and functions that will be used to extract hydrologic information about the study watershed and a figure illustrating expected results. Students are then prompted to open the Jupyter notebook and step through the individual sections, following instructions provided throughout the document. They are also encouraged to keep the HydroLearn module open on their browser alongside the dynamic notebook to guide the activity (Figure 5). Questions are posed periodically throughout the notebook, corresponding with questions in the HydroLearn module, to clarify key ideas for students and what they should be able to calculate or describe at any point in the notebook. Students are directed to the USGS National Water Information System (NWIS) website embedded within the module page to explore the stream gage location and extract key information including drainage area. To simplify the exercise, the USGS 10-m DEM for the Logan River watershed is provided in a linked HydroShare resource. Guidance is also provided for obtaining DEM datasets for other locations of interest to facilitate the use of these analysis tools and concepts at other locations.



Problem 4. Daily Streamflow

Perform your calculations below.

To download the streamflow data for an observation gage, we need to use **dataRetrieval** library in R. This library is a collection of functions to help retrieve U.S. Geological Survey (USGS) and U.S. Environmental Protection Agency (EPA) water quality and hydrology data from web services.

```
In [ ]: # First you need to install the package.
        install.packages('dataRetrieval')
```

```
In [ ]: # Now, load the package
        library(dataRetrieval)
```

The dataRetrieval library has two functions that let you retrieve the daily values and instantaneous values. Those functions are introduced below. For this exercise, we will use the function that retrieves daily values. This function takes site number, what variable to download, what type of data, and the duration. The parameters are defined in below. You can find this information for your selected gage on https://waterwatch.usgs.gov.

readNWISdv: daily value USGS NWIS data retrieval
readNWISuv: Instantaneous value data from USGS (NWIS)

```
In [ ]: # Select a USGS gage ID.  Here, I select the Blacksmith Fork USGS Gaging Station.
        SiteNumber = "10113500"        # Site ID
        ParamCode  = "00060"           # The code for the discharge
        StatCode   = "00003"           # The code for the average values
        Start_Date = "1972-10-01"      # Start date, format: YYYY-MM-DD
        End_Date   = "2015-09-30"      # End date, format: YYYY-MM-DD
```

```
In [ ]: Daily_Q = readNWISdv(SiteNumber, ParamCode, Start_Date, End_Date, StatCode)
```

Let's take a look at the data we have retrieved from the USGS using readNWISdv function of dataRetrieval library in R. If you •••

Student Response to Question 4:
The plot shows daily mean discharge observed at the Blacksmith Fork USGS gage over 43 years (1972-2015). The minimum and maximum discharge values are 26 and 1530 ft^3/s. The data was retrieved using dataRetrieval in R.

FIGURE 4 A Jupyter notebook guides students to use code to retrieve USGS streamflow data, plot time series, and answer summary questions based on their results

3.2.2 | Basic DEM hydrologic analysis

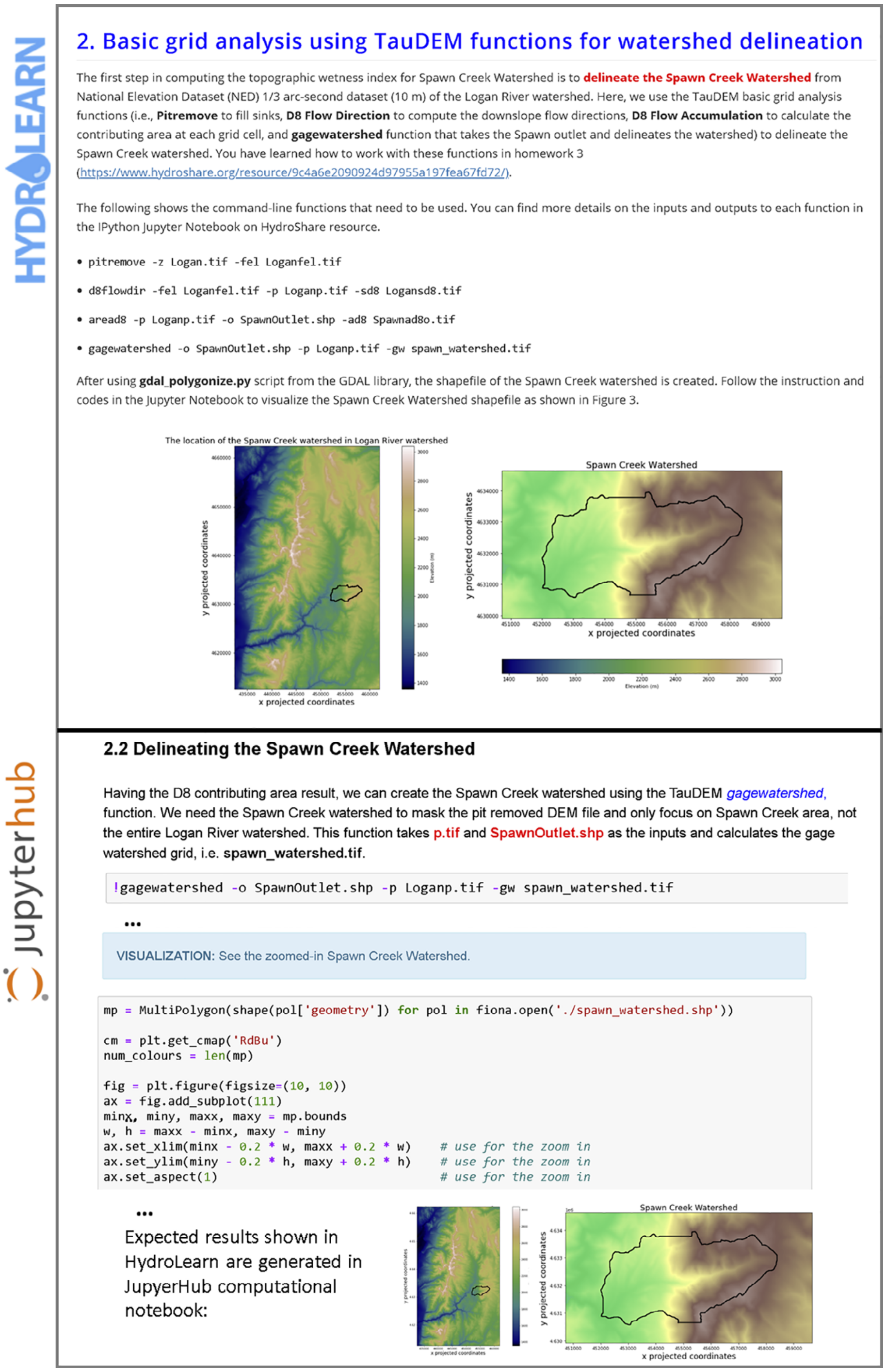
After a brief discussion of the conceptual underpinnings of the geospatial processing toolset, TauDEM, and links to useful resources, students are introduced to the Basic Grid Analysis toolset including *Pit Removal*, *D8 Flow Direction*, and *D8 Contributing Area*. The learning module again provides static code snippets and figures of expected results (Figure 5), as well as summary questions associated with the learning activity.

3.2.3 | Stream network analysis

A network analysis function within TauDEM is used to create and analyse a small section of stream network in the Logan River watershed given a DEM. The student generates a stream network and prepares an attribute table of a subset of reaches (including the link number, downstream and upstream linked reaches, contributing area, length, and corresponding watershed ID) to describe the properties of the stream network and subwatersheds. Next, the student identifies the number of grid cells in each subwatersheds, calculates the area of each subwatershed, and reconciles the values with the contributing area values from the stream network.

3.2.4 | Water balance

In the final learning activity of this section, the student delineates the watershed that was analysed in the previous steps and then uses the delineated watershed to calculate key water balance components. Streamflow data is downloaded from the USGS NWIS website for the Logan River stream gage. A web client is used to retrieve the annual 800-m precipitation for 30-year normals (1981–2010) from PRISM (Daly et al., 2000). This dataset is visualized and then clipped to the watershed extent in the Notebook. Finally, the student calculates mean annual precipitation over the watershed and reports this value along with watershed area, mean annual streamflow, and the runoff ratio.



2. Basic grid analysis using TauDEM functions for watershed delineation

The first step in computing the topographic wetness index for Spawn Creek Watershed is to **delineate the Spawn Creek Watershed** from National Elevation Dataset (NED) 1/3 arc-second dataset (10 m) of the Logan River watershed. Here, we use the TauDEM basic grid analysis functions (i.e., **Pitremove** to fill sinks, **D8 Flow Direction** to compute the downslope flow directions, **D8 Flow Accumulation** to calculate the contributing area at each grid cell, and **gagewatershed** function that takes the Spawn outlet and delineates the watershed) to delineate the Spawn Creek watershed. You have learned how to work with these functions in homework 3 (https://www.hydroshare.org/resource/9c4a6e2090924d97955a197fea67fd72/).

The following shows the command-line functions that need to be used. You can find more details on the inputs and outputs to each function in the IPython Jupyter Notebook on HydroShare resource.

- `pitremove -z Logan.tif -fel Loganfel.tif`
- `d8flowdir -fel Loganfel.tif -p Loganp.tif -sd8 Logansd8.tif`
- `aread8 -p Loganp.tif -o SpawnOutlet.shp -ad8 Spawnad8o.tif`
- `gagewatershed -o SpawnOutlet.shp -p Loganp.tif -gw spawn_watershed.tif`

After using **gdal_polygonize.py** script from the GDAL library, the shapefile of the Spawn Creek watershed is created. Follow the instruction and codes in the Jupyter Notebook to visualize the Spawn Creek Watershed shapefile as shown in Figure 3.

2.2 Delineating the Spawn Creek Watershed

Having the D8 contributing area result, we can create the Spawn Creek watershed using the TauDEM *gagewatershed,* function. We need the Spawn Creek watershed to mask the pit removed DEM file and only focus on Spawn Creek area, not the entire Logan River watershed. This function takes p.tif and SpawnOutlet.shp as the inputs and calculates the gage watershed grid, i.e. **spawn_watershed.tif**.

```
!gagewatershed -o SpawnOutlet.shp -p Loganp.tif -gw spawn_watershed.tif
```

...

VISUALIZATION: See the zoomed-in Spawn Creek Watershed.

```
mp = MultiPolygon(shape(pol['geometry']) for pol in fiona.open('./spawn_watershed.shp'))

cm = plt.get_cmap('RdBu')
num_colours = len(mp)

fig = plt.figure(figsize=(10, 10))
ax = fig.add_subplot(111)
minx, miny, maxx, maxy = mp.bounds
w, h = maxx - minx, maxy - miny
ax.set_xlim(minx - 0.2 * w, maxx + 0.2 * w)    # use for the zoom in
ax.set_ylim(miny - 0.2 * h, maxy + 0.2 * h)    # use for the zoom in
ax.set_aspect(1)                               # use for the zoom in
```

...

Expected results shown in HydroLearn are generated in JupyerHub computational notebook:

FIGURE 5 Module learning activity instructions provided in HydroLearn are designed to be used side-by-side with the Jupyter notebook to guide students' activities. The learning module introduces key programming tools and scripts that are used for basic grid analysis along with expected results

3.3 | Section M3: Runoff generation

Section M3 is distinct from the others in that it focuses on building conceptual understanding of runoff generation processes and therefore addresses only the first two aims considered in this article: promoting active learning and harnessing open web-based software. By the end of this section, a student should be able to (1) use appropriate terms to describe the processes involved in runoff generation, (2) differentiate between infiltration excess, saturation excess and subsurface stormflow runoff generation mechanisms and when and where each is more likely to occur, and (3) justify why and how specific changes in physical watershed and climate attributes will influence the dominant runoff mechanism and storm hydrograph shape. The bulk of the content is provided through the *Introduction to Rainfall-Runoff Processes* story map. This story map is an adaptation of content from an online physical hydrology workbook developed for COMET in 2003 (Tarboton, 2003). We reconceived the workbook as an interactive Story Map including updated and compatible images, videos and animations. Many pictures in the Story Map were taken in the local watershed by the authors in an effort to ground concepts in clear examples from the field. The HydroLearn section solidifies key concepts discussed in the Story Map related to rainfall-runoff processes and promotes active learning through targeted CYU questions.

3.4 | Section M6: Simulating runoff using TOPMODEL

The culminating section of the module builds on data-driven learning skills developed in M2 and concepts covered throughout previous sections to simulate semi-distributed variable source area runoff generation in a tributary to the Logan River using TOPMODEL. TOPMODEL is a conceptual hydrologic model that uses basic topographic and soils information to estimate runoff from the saturated and unsaturated zones (Beven, 1989). The location of the interface between the two zones, quantified by the water table elevation, corresponds to the soil water saturation deficit and controls the types and amounts of flow simulated by the model. At the end of the section, students should be able to (1) compute the topographic wetness index from a DEM and describe its role and use in TOPMODEL runoff calculations; (2) apply TOPMODEL principles and equations to calculate soil moisture deficit and runoff given necessary catchment and storm characteristics; and (3) critically assess TOPMODEL assumptions and determine if and why the model is appropriate in that setting.

Section M6 first provides background information on TOPMODEL, including key equations, terms, and assumptions, which are covered in more depth in the *Introduction to Physical Hydrology* Story Map. CYU questions promote active learning and prepare students for the authentic learning activity. Similar to M2, a Jupyter notebook for M6 guides the students through the specific activity steps outlined in the module. Also similar to M2, this activity could be adapted for any watershed of interest. Finally, several summary questions help to solidify concepts introduced in the learning activity.

4 | PROMOTING DATA MANAGEMENT AND REPRODUCIBILITY PRACTICES

The HydroLearn Physical Hydrology learning module was designed to promote good data management and reproducible research practices through multiple means. It provides direct training for students in the use of findable, accessible, interoperable, and reusable (FAIR, Wilkinson et al., 2016) resources accessed through a series of learning activities. Students gain experience working with open web-based tools such as CUAHSI HydroShare and JupyterHub that are explicitly designed to share hydrology data and code to facilitate transparency and reproducibility. The authentic learning activities are also designed to be readily adapted and reproduced at other locations and include specific guidance on how to obtain datasets for other locations. This design aims to empower students to reproduce the analyses and reuse the platforms, data services, and data management practices introduced here.

5 | IMPLEMENTATION

We first used the learning module during the Fall 2019 and Fall 2020 semesters with 7 and 15 students, respectively, enrolled in a first-year graduate-level physical hydrology course taught through the Department of Civil and Environmental Engineering at Utah State University. The students were from Civil Engineering or Watershed Sciences graduate programs. None of the students had used computational notebooks before, and most had limited to no programming experience. At the end of each section of the learning module, students were asked to provide open-ended feedback on their experience. In particular, we were interested to understand student perceptions of the module and the utility of open web-based tools, what worked and did not work for them, and how their conceptual understanding of the material improved after participating. All 22 students provided feedback.

6 | RESULTS AND DISCUSSION

The learning module described in this article was intended to provide distinct benefits for hydrology students, the larger research community, and hydrology instructors. For students, the learning module aims to enhance data-driven learning through student-centred learning activities that harness emerging open data services. It also provides experience and training in open web-based tools and reproducible research practices. For the research community more broadly, this type of learning module explicitly addresses the call in hydrologic science (among other areas) for open and reproducible research and provides training in data-driven, process-oriented thinking needed to advance hydrological research. For instructors, the collaborative, modular and open nature of the HydroLearn platform and computational notebooks allows content, datasets, and scripts to be readily shared, combined, and adapted to specific instructional needs and geographic settings and used at other institutions.

While this learning module has only been implemented in two graduate courses to date, initial student perceptions and the instructor's reflections are summarized below. This section reports on lessons learned, first what we found based on student questionnaires followed by reflections from the instructor that relate to emphasis, opportunities for improvement and are intended to provide some guidance regarding implementation and customization by other instructors.

6.1 | LESSONS LEARNED

6.1.1 | Student perceptions of open web-based platforms

The students found the linked ESRI Story Maps very effective at delivering information interactively and appreciated the combination of text, videos, figures, and hyperlinks to other resources. One student indicated that they wished all of their textbooks were Story Maps, and that the 'highly accessible content makes sharing with others who may be interested in these topics much easier than information out of a textbook.' Another indicated that having videos and figures interspersed with text 'helped to not only explain but show the time and space variability of the processes.' Several students noted that the interactive aspects helped break up the text and drive home concepts.

Perceptions on the utility of HydroShare and Jupyter notebooks were variable. In M1, students with even a small amount of programming experience were initially far more receptive to these tools than students with no prior experience, but this discrepancy diminished over the semester as students established more familiarity with the platforms and programming syntax. A subset of students with no programming experience indicated early on that they thought they would appreciate these tools more once they developed basic programming skills and were now interested to do so. Others said that they appreciated knowing that Jupyter notebooks exist, even if they were "still unable to replicate or augment the code so far." In terms of the structure of the notebooks themselves, most students were grateful for the amount of code that was already provided for them, but some indicated that doing more of the coding themselves would improve learning outcomes. Some expressed frustration about technological challenges such as losing server connection and needing to log out of JupyterHub and start over.

Several students were frustrated by the amount of time it took to work through the programming scripts. For instance, "it can be frustrating when you understand what you're trying to do but can't find the code to do it. I think those types of frustrations take away some of the benefits of the lab and cause students to worry more about coding than what we are doing with the code." Furthermore, "sometimes I spend more time looking and thinking about the code rather than the concepts." Even by the third learning activity, some students were still unsure of the utility of the computational notebooks as learning tools. "I still don't really like this format in general. It's cool to have the open source tools but I feel like I haven't gained the skill to apply it in any other context… I would prefer to build these together in class so we could learn how to do it ourselves."

Students generally had positive feedback about the use of HydroLearn to lay out learning objectives, activities and expectations, as well as its integration with CUAHSI HydroShare, JupyterHub and ESRI Story Maps. One student said HydroLearn was "easy and straightforward to use, and provided all relevant links making it convenient to access everything… the layout was such that I easily followed instructions for the learning activities and found the questions I needed to answer." Other students noted that the "variety of ways in which the material was presented allowed for better understanding" and "allowed for more focus on principles rather than just coding."

The CYU questions in particular appeared to help students solidify key ideas and support higher learning levels. For instance, in M1, students computed water balance uncertainty in several watersheds through a series of calculations and were then asked to check their understanding by comparing the different watershed results in their own words in the context of their physical catchment and climate settings. The computational notebooks allowed students to easily switch between calculations and text-based response in the same document. One student indicated that the CYU questions throughout the module "helped me focus on what was particularly important within smaller blocks of information" and "were really helpful for developing my understanding of rainfall-runoff processes." Another student articulated that "one of the things I enjoyed most about the module was that it really tested your understanding. The CYU questions in particular had relatively simple answers, but they did a good job of testing actual understanding of the concepts- especially the CYU question hints and explanations as to why the answer was correct."

6.1.2 | Instructor reflections

The effectiveness of integrating multiple open web-based platforms to enhance teaching hinged on the formalized pedagogical structure provided by HydroLearn. The emphasis on constructive alignment between teaching tools, learning activities and objectives facilitated development of activities that integrate data and tools from multiple sources while explicitly targeting mindfully crafted learning objectives across multiple levels (e.g., understand, apply, analyse). Mindful framing of questions encouraged the students to think critically about the underlying processes while learning the basics of the data analysis tools rather than getting lost in the mechanics of the calculations. Each section was followed up by in-class discussion regarding which settings the equations and models worked well in and which settings gave strange results and why that might be. These discussions provided an opportunity to guide students to critically evaluate model assumptions and requirements based on their varied personal experiences working with different datasets. Most students chose to work with watersheds that they were personally familiar with, often where they had grown up. The discussions that followed were much more in-depth and engaged than those the instructor has had following learning activities that rely on pre-canned data from a well-behaved system.

In early applications of this learning module, the intense focus by the instructor on familiarizing students with the tools may have distracted from clearly conveying the value of these tools. Several students questioned the need to learn how to use programming and computational notebooks to complete learning activities given other common software programs and GUIs already available to accomplish similar things. For example, "training us to use open software programs is great but … can hurt the learning process when compared with a program that has a user-friendly GUI … like ArcGIS." Given that students may prefer the use of tools they are already familiar with or have heard of, instructors must clearly articulate the value associated with using open web-based tools.

Data-driven learning was facilitated through the flexible data and programming language integration capacity of the open web-distributed platforms. In M1, students were taught how to access streamflow data for any stream gage of interest using a computational notebook written in *R* and work through the learning activity with no prior programming experience. As a result, the students were able to generate transparent and reproducible outcomes and compare their results to those generated by their peers for other watersheds. The notebooks accommodated use of different physical settings, parameter values, and datasets with minimal effort to facilitate exploration of how results varied across watersheds. For instance, students could simply input a different digital elevation model or change the storm depth value and re-run the notebook to update results and figures. Furthermore, using short and modular lines of code, students were able to accomplish all necessary tasks without the need for familiarity with a particular software interface.

6.2 | Outlook

Based on these lessons learned, we have several takeaways for other instructors who may choose to implement this or similar learning modules. Our takeaways are focused in three areas: (a) applying and adapting the learning module, (b) emphasizing concepts over tools, and (c) overcoming technical challenges.

6.2.1 | Applying and adapting the learning module

The collaborative nature of the HydroLearn platform allows instructors to create a new instance of the learning module that can be customized using data for a local watershed where students may better appreciate the context. Guidance is provided in the module for obtaining input datasets for other locations in learning activities where students are provided with example data to simplify the exercise. This guidance is intended to promote reproducible research practices and empower students to use the data analysis techniques and concepts covered in the module at other locations. Data retrieval guidance also supports module implementation by other instructors. Instructors can adapt learning activities associated with the Logan River watershed to other watersheds that are more relevant for their students.

Specific instruction was also provided on how to implement learning activities for catchments outside the U.S. or where USGS data is not available. Adapting the course material to locations outside the US would facilitate hydrologic comparison over a broader physio-climatic and geographic scope. However, since the module currently uses USGS scripts specifically designed to access USGS datasets, the changes required to achieve this are more than simply changing a USGS station identifier. Specifically, once a streamflow time series dataset is obtained for an international catchment of interest, the user would need to read that file into the workspace prior to working through subsequent analyses. This does require a higher level of programming knowledge than was required of students doing this module, but is in general easily doable for someone with modest R skills.

Instructors may adopt the entire or part of the Physical Hydrology learning module and modify the content to better reflect the specific goals defined in their syllabus. The module was recently implemented in this manner by two instructors at different universities, and the module adaptation process went smoothly. The fact that the module was hosted on a collaborative open platform allowed the instructor to make modifications to some sections in the module based on the specific course needs. The adaptation was also facilitated by the accessibility of the data and modelling resources that the module relies on. In the case of module sections linked to specific datasets and Jupyter notebooks stored in HydroShare, instructors may need to develop and link separate HydroShare resources referencing their preferred watershed. Alternatively, in the case of user-selected datasets such as in M1, there is no need to modify the resources at all, and each student will still have an authentic, personalized learning experience due to the modular data-driven nature of the learning activities.

HydroLearn learning modules are intended to be useful as standalone resources, but can also be implemented within a regular lecture series. For instance, some practicing engineers have self-enrolled in courses to further their own understanding outside of any university context. In the context of university courses and the Physical Hydrology module in particular, we believe the best approach is to include the module within an in-person lecture course. This is how we have implemented the module to-date. Some faculty authors of modules have chosen to implement the modules primarily outside of class, with little class time devoted to discussing the modules themselves, although lectures do relate to the content. Other faculty have given class time to allow students to work on the modules collaboratively and discuss results.

6.2.2 | Emphasizing concepts over tools

Instructors using this or similar learning modules are encouraged to focus students' efforts on how to apply the computational notebooks in different contexts rather than to fully understand or be able to generate every line of code and function themselves. The learning module is not intended to be an introduction to programming, although basic programming literacy is necessary and the module does provide some level of context-based learning of coding that serves as a motivator and entry point for non-coders to approach coding in a limited and practical way. An instructor may go so far as to clearly articulate that the students do *not* need to fully understand the code to effectively apply the notebooks—particularly if they do not have a strong programming background—just as they do not need to understand all the code that supports calculations in other software programs they may have worked with. Students should be helped to understand in general what code does, not how it does it. We posit that an instructor's main role in these modules should be to guide students regarding key concepts and how to implement, reproduce, and adapt an analysis in various settings—including the ability to identify key inputs, outputs, and parameters in the code.

Students may need encouragement to trust the parts of the code that they do not understand until they develop confidence through repeated evaluation of results. Instructors should, of course, still promote critical evaluation of results. Interestingly, to some extent, having the black box opened up—including all the programming scripts that are not visible in user-friendly GUIs—led to more rather than less scepticism by the students. Incorporating some videos narrated by the instructor and embedded with the module (e.g., illustrating different steps) has been shown to help support student buy-in and reduce their “shock” to the use of open platforms (Habib et al., 2018).

6.2.3 | Overcoming technical challenges

The data-driven learning activities had a relatively steep learning curve, as evident from the number of software programs, functions and packages required, as well as student feedback. The technical and technological challenges students encountered as a result required substantial technical support. Both times the learning module was implemented, the instructor had a graduate teaching assistant who provided detailed walk-throughs of the notebooks when they were first assigned and offered technical sessions and troubleshooting support for students. These additional requirements for effective implementation were a large time sink for both the instructor and teaching assistants. For subsequent applications of this or other learning modules that use open web-based platforms and programming, we encourage the use of code that is easy to understand, troubleshoot, and requires limited prior programming or operating system knowledge of students or instructors, particularly if the students have a range of backgrounds and programming experience. While we considered using only one programming language, all the functionality that we wanted to use was not equivalently available in either one of the languages. While there is an acknowledged burden associated with multiple languages, the notebooks were designed for students with little to no prior programming experience and we feel that the guidance on the differences and exposure to both languages is an important part of the learning experience.

Support mechanisms to guide learners through the data-driven procedures and provide just-in-time assistance are critical to the success of online learning activities (Habib et al., 2018; Kolodner et al., 2004). This is particularly true when multiple new tools are being presented at once, and it may be difficult to foresee where students might make mistakes or need assistance. For these reasons, the material should be presented with appropriate curricular expectations and include embedded interactive tools to support students' progression through the lessons and activities (Habib et al., 2018). The issues described above could likely be addressed in large part by incorporating additional technical support within the HydroLearn module. These user-support tools might include narrated video tutorials, additional CYU questions or check-in points, and formative feedback quizzes.

There are inevitable costs to the emphasis on new tools and software, and student feedback indicated some difficulty focusing on the key concepts and higher-level learning objectives with so much emphasis on using tools and performing calculations. Furthermore, with any technology and particularly open and open-source, there is always the challenge of future-proofing learning activities to limit the need to re-write or adjust scripts. Already, in the year since the module was first developed, several scripts had to be revised to accommodate a transition in CUAHSI JupyterHub's platform structuring. Even so, the open nature of HydroLearn allows for updates of the resources and content, as opposed to it being more difficult to update static, closed material (e.g., textbook, slides, pdfs, etc.). There are also numerous and growing options for platforms (e.g., Google Colab, GitHub) that may work as well or better than those applied in this module and have long-term support and cyberinfrastructure at a much larger scale.

7 | CONCLUSION

As an applied and interdisciplinary science, hydrology relies on direct experience with many different data sets and analysing many systems. Teaching students to explore different datasets and how well models or equations capture hydrological processes in particular settings or to solve a particular problem is essential. The learning module described in this article is a case study that demonstrates harnessing state-of-the-science open web-based technology that is increasingly utilized by the hydrology professional community to enhance physical hydrology education and prepare students to apply open and reproducible tools and practices. The data-driven learning activities allowed students to work with datasets for systems that they were particularly interested in, and enabled critical evaluation of results and assumptions. Generally, based on student perceptions and the instructor's reflections, we found that: (a) harnessing web-based platforms facilitates data-driven learning, (b) the utility of computational notebooks should be more clearly communicated, and (c) opportunities remain to enhance student learning. Challenges included some resistance to programming and unfamiliar software and time consuming technical and technological difficulties. Opportunities to further enhance data-driven learning include better articulating the benefits of using open web-based platforms upfront, incorporating additional user-support tools, and focusing methods and study questions on implementing and adapting codes to explore fundamental processes rather than tools and syntax. The profound shift in the field of hydrology toward using open data and data analysis platforms requires hydrology instructors to rethink traditional content delivery and focus instruction on using these data and data analysis tools in the preparation of future hydrologists and engineers.

ACKNOWLEDGEMENTS

This work was supported by the U.S. National Science Foundation‘s (NSF) Division of Undergraduate Education as part of Awards No. 1725989, 1726965 for the development of HydroLearn. The development of the ESRI Story Map was supported by an open educational resources award from Utah State University libraries and Garousi-Nejad's research assistantship was supported by the Utah

Water Research Laboratory. Any opinions, findings, and conclusions or recommendations expressed in this manuscript are those of the authors and do not necessarily reflect the views of the NSF.

DATA AVAILABILITY STATEMENT

The Physical Hydrology HydroLearn module described here is available at https://edx.hydrolearn.org/courses/course-v1:Utah_State_University+CEE6400+2019_Fall/about (Lane and Garousi-Nejad, 2018). Data and Jupyter Notebooks are in HydroShare (Garousi-Nejad, I. and Lane, B. 2020, 2021a, 2021b).

ORCID

Belize Lane https://orcid.org/0000-0003-2331-7038
Irene Garousi-Nejad https://orcid.org/0000-0003-2929-3946
David G. Tarboton https://orcid.org/0000-0002-1998-3479

REFERENCES

Battersby, S. E., & Remington, K. C. (2013). Story maps in the classroom. *Arcuser Magazine Spring*. https://www.esri.com/about/newsroom/wp-content/uploads/2018/11/storymaps.pdf

Beven, K. (1989). Changing ideas in hydrology—The case of physically-based models. *Journal of Hydrology*, *105*(1–2), 157–172. https://doi.org/10.1016/0022-1694(89)90101-7

Biggs, J., & Tang, C. (2011). Train-the-trainers: Implementing outcomes-based teaching and learning in Malaysian higher education. *Malaysian Journal of Learning and Instruction*, *8*, 1–19. https://files.eric.ed.gov/fulltext/EJ1137298.pdf

Bourget, P. G. (2006). Integrated water resources management curriculum in the United States: Results of a recent survey. *Journal of Contemporary Water Research & Education*, *135*, 107–114. https://doi.org/10.1111/j.1936-704X.2006.mp135001013.x

Ceola, S., Arheimer, B., Baratti, E., Blöschl, G., Capell, R., Castellarin, A., Freer, J., Han, D., Hrachowitz, M., Hundecha, Y., Hutton, C., Lindström, G., Montanari, A., Nijzink, R., Parajka, J., Toth, E., Viglione, A., & Wagener, T. (2015). Virtual laboratories: New opportunities for collaborative water science. *Hydrology and Earth System Sciences*, *19*(4), 2101–2117.

Chen, M., Voinov, A., Ames, D. P., Kettner, A. J., Goodall, J. L., Jakeman, A. J., Barton, M. C., Harpham, Q., Cuddy, S. M., DeLuca, C., Yue, S., Wang, J., Zhang, F., Wen, Y., & Lü, G. (2020). Position paper: Open web-distributed integrated geographic modelling and simulation to enable broader participation and applications. *Earth-Science Reviews*, *207*, 103223. https://doi.org/10.1016/j.earscirev.2020.103223

Choi, Y.-D., Goodall, J. L., Sadler, J. M., Castronova, A. M., Bennett, A., Li, Z., Nijssen, B., Wang, S., Clark, M. P., Ames, D. P., Horsburgh, J. S., Yi, H., Bandaragoda, C., Seul, M., Hooper, R., & Tarboton, D. G. (2021). Toward open and reproducible environmental modeling by integrating online data repositories, computational environments, and model application programming interfaces. *Environmental Modelling & Software*, *135*, 104888. https://doi.org/10.1016/j.envsoft.2020.104888

Clark, M. P., Schaefli, B., Schymanski, S. J., Samaniego, L., Luce, C. H., Jackson, B. M., Freer, J. E., Arnold, J. R., Moore, R. D., Istanbulluoglu, E., & Ceola, S. (2016). Improving the theoretical underpinnings of process-based hydrologic models. *Water Resources Research*, *52*(3), 2350–2365. https://doi.org/10.1002/2015WR017910

Daly, C., Taylor, G. H., Gibson, W. P., Parzybok, T. W., Johnson, G. L., & Pasteris, P. A. (2000). High-quality spatial climate data sets for the United States and beyond. *Transactions of the ASAE*, *43*(6), 1957–1962. https://pubag.nal.usda.gov/catalog/27078

Eagleson, P. S. (1991). Hydrologic science: A distinct geoscience. *Reviews of Geophysics*, *29*(2), 237–248. https://doi.org/10.1029/90RG02615

Essawy, B. T., Goodall, J. L., Voce, D., Morsy, M. M., Sadler, J. M., Choi, Y. D., Tarboton, D. G., & Malik, T. (2020). A taxonomy for reproducible and replicable research in environmental modelling. *Environmental Modelling & Software*, *134*, 104753. https://doi.org/10.1016/j.envsoft.2020.104753

Gallagher, M.A., Byrd, J. L., Habib, E., Tarboton, D., & Willson, C. S. (2021). *HydroLearn: Improving students' conceptual understanding and technical skills in a civil engineering senior design course*. Paper presented at ASEE Annual Conference & Exposition.

Garousi-Nejad, I., B. Lane (2020). Hydrologic statistics and data analysis (HL1). *HydroShare*. http://doi.org/10.4211/hs.873b5490f986428bbd23e37c81906402.

Garousi-Nejad, I., B. Lane (2021a). Digital elevation models and GIS in hydrology (M2). *HydroShare*. https://doi.org/10.4211/hs.9c4a6e2090924d97955a197fea67fd72.

Garousi-Nejad, I., B. Lane (2021b). Calculating runoff using TOPMODEL (M6). *HydroShare*. https://doi.org/10.4211/hs.ea30176c717d4b7baeb85c16427f1d6f.

Goodall, J. L., Merwade, V., Zaslavsky, I., Tarboton, D. G., Horsburgh, J. S., & Ames. D. P., & Couch, A (2017). Chapter 7: Hydrologic information systems. In V. P. Singh (Ed.), *Handbook of applied hydrology* (pp. 7-1–7-9). McGraw Hill.

Habib, E., Deshotel, D., & Williams, D. (2018). Unlocking the educational value of large-scale, coastal-ecosystem restoration projects: Development of student-centered, multidisciplinary learning modules. *Journal of Coastal Research*, *34*(3), 738–751. https://doi.org/10.2112/JCOASTRES-D-17-00064.1

Habib, E., Deshotel, M., Lai, G. L., & Miller, R. (2019). Student perceptions of an active learning module to enhance data and modeling skills in undergraduate water resources engineering education. *International Journal of Engineering Education*, *35*(5), 1353–1365. https://par.nsf.gov/biblio/10189082

Habib, E., Ma, Y., Williams, D., Sharif, H. O., & Hossain, F. (2012). HydroViz: Design and evaluation of a web-based tool for improving hydrology education. *Hydrology and Earth System Sciences*, *16*(10), 3767–3781. https://doi.org/10.5194/hess-16-3767-2012

Horsburgh, J. S., Aufdenkampe, A. K., Mayorga, E., Lehnert, K. A., Hsu, L., Song, L., Jones, A. S., Damiano, S. G., Tarboton, D. G., Valentine, D., Zaslavsky, I., & Whitenack, T. (2016). Observations data model 2: A community information model for spatially discrete earth observations. *Environmental Modelling & Software*, *79*, 55–74. https://doi.org/10.1016/j.envsoft.2016.01.010

Horsburgh, J. S., Morsy, M. M., Castronova, A. M., Goodall, J. L., Gan, T., Yi, H., Stealey, M. J., & Tarboton, D. G. (2016). HydroShare: Sharing diverse environmental data types and models as social objects with application to the hydrology domain. *JAWRA Journal of the American Water Resources Association*, *52*(4), 873–889. https://doi.org/10.1111/1752-1688.12363

Horsburgh, J. S., Tarboton, D. G., Maidment, D. R., & Zaslavsky, I. (2008). A relational model for environmental and water resources data. *Water Resources Research*, *44*(5). https://doi.org/10.1029/2007WR006392

Hutton, C., Wagener, T., Freer, J., Han, D., Duffy, C., & Arheimer, B. (2016). Most computational hydrology is not reproducible, so is it really science? *Water Resources Research*, *52*(10), 7548–7555. https://doi.org/10.1002/2016WR019285

Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.Org. *Computers in Human Behavior*, *52*, 200–210. https://doi.org/10.1016/j.chb.2015.05.047

Kandlbinder, P. (2014). Constructive alignment in university teaching. *HERDSA Review of Higher Education*, *36*(3), 5–22. https://www.herdsa.org.au/herdsa-review-higher-education-vol-1/5-22

Kerski, J. (2019). A geomorphology field trip in Northwest Indiana story map. *ArcGIS Story Map*. https://storymaps.arcgis.com/stories/79348b0fd4474d8c87819dc95ef36c1e

Kolodner, J. L., Owensby, J. N., & Guzdial, M. (2004). Case-based learning aids. In D. Jonassen & M. Driscoll (Eds.), *Handbook of research on educational communications and technology (2nd ed.)* (pp. 829–861). Routledge.

Lane, B. A. & Garousi Nejad, I. (2018), Physical hydrology. *HydroLearn*. https://edx.hydrolearn.org/courses/course-v1:Utah_State_University+CEE6400+2019_Fall/about

Litzinger, T., Lattuca, L. R., Hadgraft, R., & Newstetter, W. (2011). Engineering education and the development of expertise. *Journal of Engineering Education*, *100*(1), 123–150. https://doi.org/10.1002/j.2168-9830.2011.tb00006.x

Lyon, S. W., & Teutschbein, C. (2011). Problem-based learning and assessment in hydrology courses: Can non-traditional assessment better reflect intended learning outcomes? *Journal of Natural Resources and Life Sciences Education*, *40*(1), 199–205. https://doi.org/10.4195/jnrlse.2011.0016g

Marshall, J. A., Castillo, A. J., & Cardenas, M. B. (2013). Assessing student understanding of physical hydrology. *Hydrology & Earth System Sciences*, *17*(1), 829–836. https://doi.org/10.5194/hess-17-829-2013

Merck, M. F., Gallagher, M. A., Habib, E., & Tarboton, D. (2021). Engineering Students' perceptions of mathematical modeling in a learning module centered on a hydrologic design case study. *International Journal of Research in Undergraduate Mathematics Education*, 1–27. http://doi.org/10.1007/s40753-020-00131-8

Merwade, V., & Ruddell, B. L. (2012). Moving university hydrology education forward with community-based geoinformatics, data and modeling resources. *Hydrology and Earth System Sciences*, *16*(8), 2393–2404. https://doi.org/10.5194/hess-16-2393-2012

National Research Council. (1991). *Opportunities in the hydrologic sciences*. The National Academies Press. https://doi.org/10.17226/1543

Ngambeki, I., Thompson, S. E., Troch, P. A., Sivapalan, M., & Evangelou, D. (2012). Engaging the students of today and preparing the catchment hydrologists of tomorrow: Student-centered approaches in hydrology education. *Hydrology & Earth System Sciences Discussions*, *9*(1), 707–740. https://doi.org/10.5194/hessd-9-707-2012

R Core Team (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

Rajib, M. A., Merwade, V., Kim, I. L., Zhao, L., Song, C., & Zhe, S. (2016). SWATShare – A web platform for collaborative research and education through online sharing, simulation and visualization of SWAT models. *Environmental Modelling & Software*, *75*, 498–512. https://doi.org/10.1016/j.envsoft.2015.10.032

Rosenberg, D. E., Filion, Y., Teasley, R., Sandoval-Solis, S., Hecht, J. S., Van Zyl, J. E., McMahon, G. F., Horsburgh, J. S., Kasprzyk, J. R., & Tarboton, D. G. (2020). The next frontier: Making research more reproducible. *Journal of Water Resources Planning and Management*, *146*(6), 01820002. https://doi.org/10.1061/(ASCE)WR.1943-5452.0001215

Ruddell, B. L., & Wagener, T. (2015). Grand challenges for hydrology education in the 21st century. *Journal of Hydrologic Engineering*, *20*(1), p. A4014001. https://doi.org/10.1061/(ASCE)HE.1943-5584.0000956

Ryberg, K. R., & Vecchia, A. V. (2012). *An R package for retrieval, analysis, and anomaly calculation of daily hydrologic time series data*, version 1.0 (No. 2012–1168; Open-File Report). US Department of the Interior, US Geological Survey. https://doi.org/10.3133/ofr2012116

Sanchez, C. A., Ruddell, B. L., Schiesser, R., & Merwade, V. (2016). Enhancing the T-shaped learning profile when teaching hydrology using data, modeling, and visualization activities. *Hydrology and Earth System Sciences*, *20*(3), 1289–1299. https://doi.org/10.5194/hess-20-1289-2016

Slater, L. J., Thirel, G., Harrigan, S., Delaigue, O., Hurley, A., Khouakhi, A., Prosdocimi, I., Vitolo, C., & Smith, K. (2019). Using R in hydrology: A review of recent developments and future directions. *Hydrology and Earth System Sciences*, *23*(7), 2939–2963. https://doi.org/10.5194/hess-23-2939-2019

Stagge, J. H., Rosenberg, D. E., Abdallah, A. M., Akbar, H., Attallah, N. A., & James, R. (2019). Assessing data availability and research reproducibility in hydrology and water resources. *Scientific data*, *6*(1), 190030. https://doi.org/10.1038/sdata.2019.30

Tarboton, D. G. (2003). *Rainfall-runoff processes*. Utah State University.

Tarboton, D. G. (2018). Terrain Analysis Using Digital Elevation Models (TauDEM). *Utah Water Research Laboratory, Utah State University*. http://hydrology.usu.edu/taudem

Tarboton, D. G., Idaszak, R., Horsburgh, J. S., Heard, J., Ames, D., Goodall, J. L., Band, L., Merwade, V., Couch, A., Arrigo, J., Hooper, R., Valentine, D., & Maidment, D. (2014). HydroShare: Advancing collaboration through hydrologic data and model sharing. In D. P. Ames, N. W. T. Quinn, & A. E. Rizzoli (Eds.), *Proceedings of the 7th International Congress on Environmental Modelling and Software*. International Environmental Modelling and Software Society (iEMSs). https://scholarsarchive.byu.edu/iemssconference/2014/Stream-A/7/

Tesfa, T. K., Tarboton, D. G., Watson, D. W., Schreuders, K. A. T., Baker, M. E., & Wallace, R. M. (2011). Extraction of hydrological proximity measures from DEMs using parallel processing. *Environmental Modelling & Software*, *26*(12), 1696–1709. https://doi.org/10.1016/j.envsoft.2011.07.018

Vogel, R. M., Lall, U., Cai, X., Rajagopalan, B., Weiskel, P. K., Hooper, R. P., & Matalas, N. C. (2015). Hydrology: The interdisciplinary science of water. *Water Resources Research*, *51*(6), 4409–4430. https://doi.org/10.1002/2015WR017049

Wagener, T., Kelleher, C., Weiler, M., McGlynn, B., Gooseff, M., Marshall, L., Meixner, T., McGuire, K., Gregg, S., Sharma, P., & Zappe, S. (2012). It takes a community to raise a hydrologist: The modular curriculum for hydrologic advancement (MOCHA). *Hydrology and Earth System Sciences*, *16*(9), 3405–3418. https://doi.org/10.5194/hess-16-3405-2012

Wagener, T., Savic, D., Butler, D., Ahmadian, R., Arnot, T., Dawes, J., Djordjevic, S., Falconer, R., Farmani, R., Ford, D., Hofman, J., Kapelan, Z., Pan, S., & Woods, R. (2021). Hydroinformatics education—The water informatics in science and engineering (WISE) Centre for doctoral training. *Hydrology and Earth System Sciences*, *25*, 2721–2738. https://doi.org/10.5194/hess-25-2721-2021

Wagener, T., Weiler, M., McGlynn, B., Gooseff, M., Meixner, T., Marshall, L., McGuire, K., & McHale, M. (2007). Taking the pulse of hydrology education. *Hydrological Processes*, *21*(13), 1789–1792. https://doi.org/10.1002/hyp.6766

Ward, A., Herzog, S., Bales, J., Barnes, R., Ross, M., Jefferson, A., N. Basu, L. Yoder, T. Covino, E. Habib, Maertens, J. (2020). Educational resources for hydrology & water resources. *HydroShare*. http://www.hydroshare.org/resource/148b1ce4e308427ebf58379d48a17b91

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J. W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., … Mons, B. (2016). The FAIR guiding principles for scientific data management and stewardship. *Scientific data*, *3*(1), 160018. https://doi.org/10.1038/sdata.2016.18

How to cite this article: Lane, B., Garousi-Nejad, I., Gallagher, M. A., Tarboton, D. G., & Habib, E. (2021). An open web-based module developed to advance data-driven hydrologic process learning. *Hydrological Processes*, *35*(7), e14273. https://doi.org/10.1002/hyp.14273