CS230

Predicting Reservoir Inflow with Augmented Tabular Data

Travis Grafton
Department of Computer Science
Stanford University
tjgraft@stanford.edu

Abstract

For this project, I developed and evaluated a deep neural network (DNN) framework for reservoir inflow forecasting. The DNN was trained using hydrological and meteorological features known to be correlative with inflow rates at headwater reservoirs. To address the bottleneck of insufficient training data and avoid overfit, two primary data augmentation techniques were used where applicable. First, monthly time-series data was disaggregated into its daily components where sufficient information existed. Second, additional data was synthesized using a conditional tabular GAN (CTGAN) framework. The DNN was evaluated using inflow rates from the Navajo Reservoir in the Upper Colorado River Basin (UCRB). Daily data produced the best overall results. Augmenting the daily data with synthetic data sometimes resulted in faster convergence but did not produce significantly more accurate results.

1 Introduction

The Colorado River Basin (CRB) is a critical water resource for the southwestern United States but does not provide consistent year-to-year streamflow and is vulnerable to ongoing droughts. Over the last century, the naturalized flow of the Colorado River had decreased by approximately 15%. Using a Variable Infiltration Capacity (VIC) model, Xiao, et al.[7] have shown that about half of the long-term decrease in natural flow is associated with general warming in the CRB. Effective water resource management is contingent on the ability to predict inflow at headwater reservoirs, and since average temperatures in the UCRB are expected to continue to rise with climate change, predictive models need to be as robust as possible while still processing a variety of interrelated statistical and dynamical inputs.

The ability of DNNs to handle complex nonlinear relationships and large numbers of features where the correlations between features and output are not always clear, as well as their ability to extrapolate based on trends within the training data, makes them an appealing choice for a predictive model. The DNN for this project was designed with this in mind and therefore takes as inputs three types of features: Satellite-based measurements of surface evaporation and soil moisture, snow water equivalents (SWE) that estimate the amount of water contained in the surrounding snowpack, and hydroclimate indices for the Pacific Ocean. Additionally, given the relative lack of training data to train an RNN, the time-series element of the data was encoded into the features by including one- and two-month lag measurements for both the above features and for inflow at Navajo Reservoir. The DNN then outputs an estimate of current flowrate based on these lagged measurements.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

2 Related work

Various machine learning frameworks have been deployed for inflow forecasting using statistical methods (i.e., relating current streamflow to antecedent meteorological, hydrological, or climatological conditions). Kalra, et al.[1] used a support vector machine (SVM) to make predictions of annual streamflow in the Gunnison River Basin and San Juan River Basin with a 1-year lead time. Their training data was based on climatological oceanic–atmospheric indices including Pacific decadal oscillation (PDO), North Atlantic oscillation (NAO), and El Niño southern oscillation (ENSO). Additionally, Li, et al.[3] also incorporated sea surface temperature (SST) oscillations when using machine learning methods, including random forest (RF), support vector regression (SVR), and extreme learning machine (ELM) in an ensemble to make drought predictions.

Woodson, et al.[6] used a random forest (RF) framework conditioned on simulated temperature data from the Community Earth System Model-Decadal Prediction Large Ensemble (CESM-DPLE) to project both decadal streamflow and reservoir pool elevations along the Colorado River and found that RF predictions outperformed Ensemble Streamflow Prediction (ESP) models. Talsma, et al.[4] recently used unsupervised learning methods to characterize drought indicators in the CRB. K-means clustering was used to identify areas in the CRB where drought indicator behavior is likely to occur. Finally, Tian et al.[5] utilized five machine learning methods, including RF and ELM, in a Bayesian Ensemble to predict streamflow in the UCRB. They found that the ensemble did not always outperform RF, but that RF tended to have worse performance on outlier events (i.e., unusually high flow), likely due to its inability to make predictions that fall outside the range of its training data.

3 Dataset and Features

The Navajo Reservoir in the upper CRB is mostly dominated by snow and has been shown to have strong lagged impacts from initial hydrological conditions [2]. This makes it ideal as a proof-of-concept for using time-lagged data to make predictions for future streamflow. Historical inflow data for the Navajo Reservoir is available from the US Bureau of Reclamation website. Monthly inflow data is available from 1962. Daily inflow data is available from 1980. Unregulated inflow, which accounts for the effects of upstream reservoirs, was used with one- and two-month lag.

The following Hydro-climatic indices for the Pacific Ocean were incorporated as additional features: PDO, Pacific/North American Index (PNA), East Central Tropical Pacific SST (NINO34), and Western Hemisphere Warm Pool (WHWP). These indices and others are available from the NOAA Physical Sciences Laboratory.

Satellite-based hydrological data was downloaded from the Global Land Evaporation Amsterdam Model (GLEAM), which is a set of algorithms that estimate components of evapotranspiration and soil moisture. GLEAM data is available on a 0.25° latitude-longitude and daily resolution. Values were extracted at the coordinates corresponding to the Navajo Reservoir daily from 1980. The following estimates were used: Actual evaporation, potential evaporation, surface soil moisture, and root-zone soil moisture. This data was also lagged by one and two months.

Lastly, SWE data was obtained from the Colorado Basin River Forecast Center (CBRFC), which uploads data from multiple stations in the San Juan Snotel Group. The records from these stations are not consistent and some years are missed, so an average across all available stations for a given year was used to represent SWE. This data was also lagged in the training data.

In total, the raw dataset used was composed of 15281 examples with 21 features each.

4 Methods

A DNN was constructed using TensorFlow. Because the 21 input features of the algorithm cover a wide range of values, a normalization layer was used, followed by 18 densely connected hidden layers with 64 hidden units for the first 12 hidden layers, 30 for the next 2, 20 for the next, 10 for the next two, and 1 unit for the output layer. ReLU activations were used for each hidden layer. Dropout with

a frequency of 0.1 was added for regularization. Stochastic gradient descent with Adam optimization was used for training. Mean absolute error was used as the loss function:

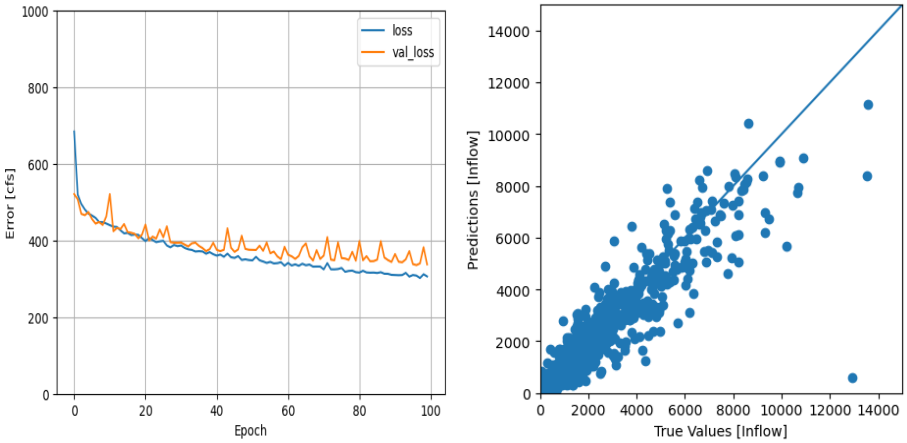
$$loss = abs(y_{true} - y_{pred}) \tag{1}$$

Over mean squared error (MSE). This is due to the fact that MSE tends to emphasize outlying data points, which were expected to be common in the training and test sets.

Despite using time lags of one and two months, the data used was split into daily granularity instead of the monthly. Essentially, breadth of data was sacrificed for depth of data, and 700 examples over 60 years were converted to 15000 examples over 40 years.

5 Experiments/Results/Discussion

The 15281 examples were split into a training set of 13753 and a test set of 1528. When training, 0.2 of the training set was used for validation. Training the DNN on unaugmented data for 100 epochs resulted in a loss of 334 on the test set.



Training the model up to 700 epochs resulted in loss of 296 on the test set.

To generate synthetic data, I utilized the CTGAN framework as described by Xu, et al.[8] An experimental CTGAN for synthesizing tabular data is available as part of the Synthetic Data Vault (SDV) library maintained by DataCebo. CTGAN is capable of handling discrete, continuous, and categorical tabular data. However, to speed the training process, I "categorized" the flowrates into discrete values that could easily be 1-hot encoded by floor dividing them (i.e., a flowrate of 345.5 cfs becomes the discrete value 3).

After training CTGAN for 300 epochs on the 13753 training examples, it was used to generate 100000 additional synthetic training examples. Training the DNN on the total combined synthetic and real examples for 100 epochs resulted in a loss of 398.75 on the test set, substantially worse than the loss from training only on real examples.

Training the DNN on all 13753 real examples and 16247 synthetic examples for 100 epochs resulted in a loss of 354.83 on the test set. Training for an additional 400 epochs resulted in a loss of 327.67 on the test set.

Training the DNN on all 13753 real examples and 6247 synthetic examples for 100 epochs resulted in a loss of 316.09 on the test set. Training for 1200 epochs total resulted in a loss of 302.65 on the test set.

6 Conclusion/Future Work

Ultimately, a DNN shows promise even when used with a relatively low number of training examples. However, current synthetic tabular data generation does not appear to enhance training. As models

like CTGAN continue to develop, viable next steps could be generating synthetic time-series data and training on a recurrent neural network.

References

[1] Ajay Kalra, William P. Miller, Kenneth W. Lamb, Sajjad Ahmad, and Thomas Piechota. Using large-scale climatic patterns for improving long lead time streamflow forecasts for gunnison and san juan river basins. *Hydrological Processes*, 27(11):1543–1559, May 2012.

[2] Randal D. Koster, Sarith P. Mahanama, Ben Livneh, Dennis P. Lettenmaier, and Rolf H. Reichle. Skill in streamflow forecasts derived from large-scale estimates of soil moisture and snow. *Nature Geoscience*, 3(9):613–616, 2010.

[3] Jun Li, Zhaoli Wang, Xushu Wu, Chong-Yu Xu, Shenglian Guo, Xiaohong Chen, and Zhenxing Zhang. Robust meteorological drought prediction using antecedent sst fluctuations and machine learning. *Water Resources Research*, 57(8), 2021.

[4] Carl J. Talsma, Katrina E. Bennett, and Velimir V. Vesselinov. Characterizing drought behavior in the colorado river basin using unsupervised machine learning. *Earth and Space Science*, 9(5), 2022.

[5] Di Tian, Xiaogang He, Puneet Srivastava, and Latif Kalin. A hybrid framework for forecasting monthly reservoir inflow based on machine learning techniques with dynamic climate forecasts, satellite-based data, and climate phenomenon information. *Stochastic Environmental Research and Risk Assessment*, 36(8):2353–2375, 2021.

[6] David Woodson, Balaji Rajagopalan, Sarah Baker, Rebecca Smith, James Prairie, Erin Towler, Ming Ge, and Edith Zagona. Stochastic decadal projections of colorado river streamflow and reservoir pool elevations conditioned on temperature projections. *Water Resources Research*, 57(12), 2021.

[7] Mu Xiao, Bradley Udall, and Dennis P. Lettenmaier. On the causes of declining colorado river streamflows. *Water Resources Research*, 54(9):6739–6756, 2018.

[8] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. 2019.