Roman Chernenko Vitaly Bondar

“Water Supply Forecast Rodeo” competition report by “ck-ua” team

Abstract

In our report, we detail the methods and outcomes of our participation in the "Water Supply Forecast Rodeo" competition, where we aimed to predict water flow levels with constrained data availability. Our approach centered on using a Multi-Layer Perceptron (MLP) neural network with four layers, which proved to be the most effective within the given constraints. The network was constructed to simultaneously predict the 10th, 50th, and 90th percentile targets of water level distribution. We experimented with various network enhancements and dropout regularization but observed no substantial improvement in the model's performance.

The proposed solution is highly efficient and fast.

For our data sources, we relied on the NRCS and RFCs monthly naturalized flow, USGS streamflow, and NRCS SNOTEL data, all of which were meticulously normalized and encoded to serve as features for our training process. We propose a novel approach for using SNOTEL data by training specialized RANSAC mini-models for each site separately. For each of these mini-models the list of the used SNOTEL stations are selected by heuristic approach.

Our validation strategy involved a repeated k-fold cross-validation based on years, to both avoid overfitting and to reinforce the robustness of our model. This led us to train 25 models with distinct training years and, upon inference, we employed an ensemble of all models to determine the median value for predictions of each percentile. We also employed data augmentation due to the limited size of our training dataset, which allowed us to artificially expand our sample set.

As a result, we demonstrate that water supply forecasts may be done with the lightweight simple but robust and powerful single model for all the flow sites used in the competition.

Technical Approach

Algorithm and Architecture Selection

In light of the constrained dataset at our disposal, the decision was made to opt for compact neural networks. The utilization of small convolutional neural networks[7] was ruled out due to the unique characteristics of our data, where certain data points could be overlooked or exhibit varying distances to neighboring points. Although we initially planned to explore the feasibility of employing small transformer models[11], this avenue remained unexplored due to time constraints.

Consequently, our chosen solution revolves around a straightforward Multi-Layer Perceptron (MLP) network[8] comprising four layers with Rectified Linear Unit (ReLU)[1] activations. Our investigation revealed that predicting the residual value for the seasonal water level is a relatively simpler task for the model, and thus, this approach was incorporated into the final submission (Fig. 1).

The resultant architecture is designed to yield three outputs, enabling predictions for the 10th, 50th, and 90th percentiles of the distribution in a single execution.

Furthermore, we conducted experiments with various enhancements, including the incorporation of residual connections within the MLP, concatenation of all inputs to the last layer, adopting a densenet-like structure, and introducing additional embeddings on top of the input features. Regrettably, none of these approaches yielded significant improvements in the overall score.

Interestingly, incorporating dropout regularization[3, 9] to our solution always reduced the final score.

Additionally our team explored some bayesian based approaches for estimation of the full probability density function of the target value. But we faced many obstacles in the estimation of the posterior probability and we refused this direction.

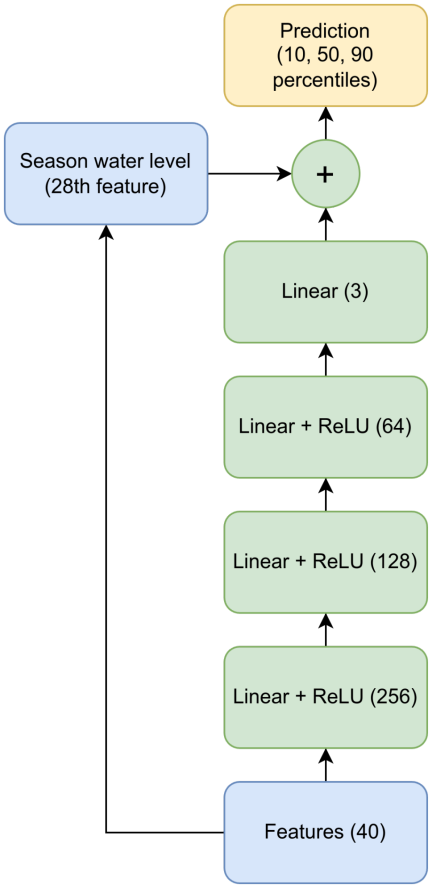

Fig. 1 Model architecture

Important aspect of our approach is fast inference. Model may handle all the Hindcast test set in 80 seconds, including features preparation.

Data Sources and Feature Engineering

Due to the big difference between target volumes of different sites, all target volume values from the training dataset were normalized to 99th percentile independently for each site_id. So after normalization “0” values means no water flow and “1” values means almost maximum possible water flow during all available historical data.

The site_id value was encoded with the one-hot vector of size 26.

We used next data sources in the final version of the solution:

- NRCS and RFCs monthly naturalized flow
- USGS streamflow
- NRCS SNOTEL

NRCS and RFCs monthly naturalized flow data preprocessing and data engineering

All monthly naturalized flow data that is calculated before the issued data is loaded. All rows with empty values are dropped. The monthly flow values normalized to the same coefficients as train target volumes.

We used next features for monthly naturalized flow data:

- sum of monthly flow data from the start of waterseason to the month before issue date
- sum of monthly flow data from the start of forecast season (March of April depends of site_id) to the month before issue date
- normalized elapsed time from the start of waterseason
- normalized elapsed time from the start of forecast season

USGS streamflow data preprocessing and data engineering

All USGS streamflow data that measured before the issued date is loaded. We used only the '00060_Mean' column from .csv files that corresponds to discharge in cubic feet per second[10]. All streamflow values was converted to acre*foot/day values and then normalized in the same way as monthly naturalized flow data.

The images below on Fig. 2 represent the correlation between aggregation of USGS streamflow to target volume for two different sites. For some sites the correlation is very high and for other correlation is significant too, but noisy. The difference between aggregated USGS streamflow and target volume for some sites can be related with some river regulation as suggested in the data source description.

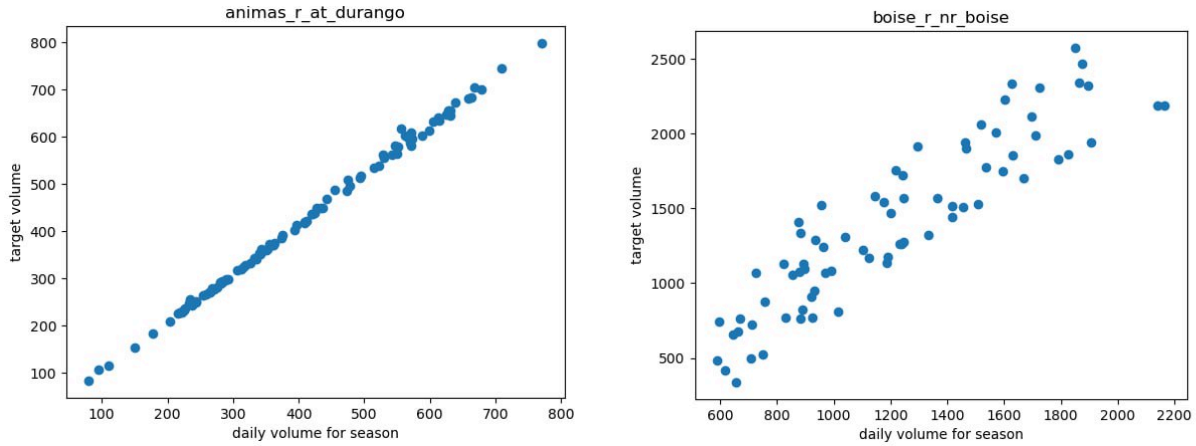

Fig.2 The correlation between aggregated USGS streamflow data and target volume

We used next features for USGS streamflow data:

- sum of streamflow daily volumes from the start of waterseason to the day before issue date
- sum of streamflow daily volumes from the start of forecast season (March of April depends of site_id) to the day before issue date
- normalized elapsed time from the start of waterseason
- normalized elapsed time from the start of forecast season

NRCS SNOTEL data preprocessing and data engineering

The main challenge of using SNOTEL data as a feature for the model related to different amounts of stations for each basin. The first attempt of using SNOTEL data is to choose the SNOTEL station that has the best correlation from aggregated precipitation to the target volume. But in this case we lost a lot of data from other SNOTEL stations.

The final solution uses a linear regression from several SNOTEL aggregated precipitation to target volume as normalization functions. The special greedy heuristic algorithm is implemented to choose the best SNOTEL subset of stations for each site independently. The same normalization function was also applied to snow-water equivalent values of SNOTEL data. The idea behind using of several SNOTEL stations that we can make better spatial coverage of the basin and surrounding land and get information from points with different weather conditions.

The RANSAC regression model implementation from scikit-learn package is used for better outlier filtering.

Using the combination of SNOTEL stations makes a significant boost for the score on the local cross-validation and on the Hindcast leaderboard.

There is a significant correlation between maximum aggregated precipitation value and the target volumes as shown on charts below (Fig. 3).

We used only 'PREC_DAILY' and 'WTEQ_DAILY' features from SNOTEL data thats corresponds to Precipitation data and Snow water equivalent data respectively.

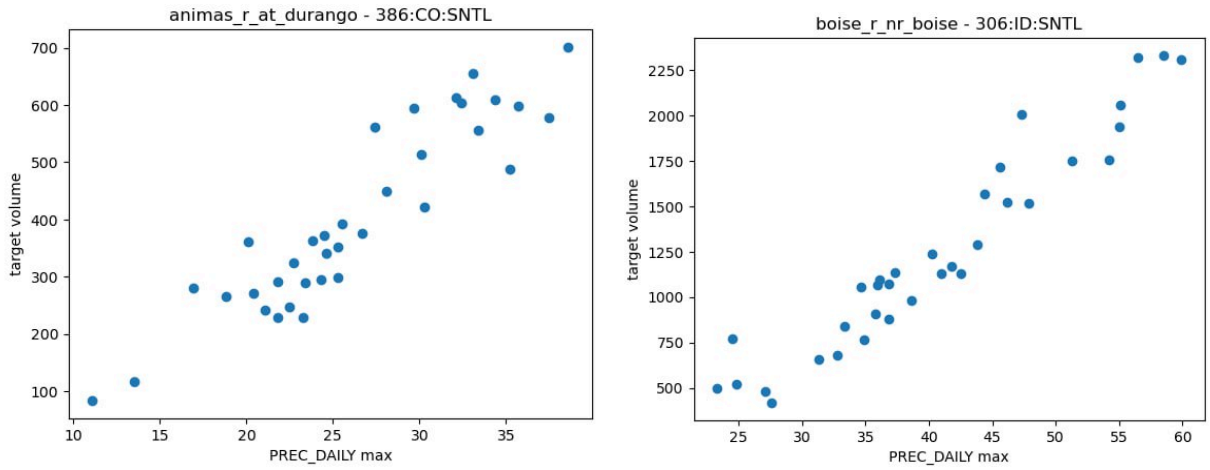

Fig. 3 The correlation between SNOTEL precipitation values and target volumes

We used next preprocessed features for NRCS SNOTEL data:

- aggregated normalized precipitation for the day before issue date
- latest normalized snow-water-equvalent value for date before issue date
- maximum of snow-water-equivalent values from the start of waterseason to the day before the issue date
- normalized elapsed time from the start of waterseason
- normalized elapsed time from the start of forecast season

Other data sources that not included in the final solution

We also tried next data sources too, but not find any significant improvement and therefore not included it into the final solution:

- CDEC Snow Sensor Network
- UA/SWANN
- CPC Seasonal Outlooks
- Palmer Drought Severity Index (PDSI) from gridMET

The UA/SWANN and PDSI datasets are available as raster georeferencing maps in different geographical projections and therefore required preprocessing to use it in our model. The main idea of preprocessing is considered in min/mean/max aggregation of raster data inside each basin polygon separately.

In work [2] showed that limited direct soil moisture observations could improve statistical forecast accuracy. We tried to use the PDSI dataset as the source of soil moisture data, but we could not achieve noticeable results with it.

We also tried to use the CDEC stations network as additional information for precipitation and snow-water-equivalent data for the basins in California where there is a lack of SNOTEL stations. But using CDEC stations does not improve the score on the cross-validation.

We also tried to use the sequence of three-month CPS seasonal outlooks from the start of waterseason to the end of season as a long-term weather forecast, but it did not change the cross-validation score either.

Several basins at north of the USA also extend to Canada and will also be useful to try to include SNOTEL-equivalent data about precipitation from Canada to improve the quality of forecasting. Seems this kind of data is available on www.nrcs.usda.gov website, but we found it too late to request it as an available data source.

Also we had an idea to use MODIS Snow Cover product to detect snow melting date outside of SNOTEL stations, but not implemented it due to lack of time.

Features summary

Index	Data source	Feature description
0-25		One-hot encoding of site_id values
26	montly naturalized flow	Sum of monthly flow data from the start of waterseason to the month before issue date
27		Normalized elapsed time from the start of waterseason
28		Sum of monthly flow data from the start of forecast season to the month before issue date
29		Normalized elapsed time from the start of forecast season
30	USGS streamflow	Sum of streamflow daily volumes from the start of waterseason to the day before issue date
31		Normalized elapsed time from the start of waterseason
32		Sum of streamflow daily volumes from the start of forecast season to the day before issue date
33		Normalized elapsed time from the start of forecast season
34	SNOTEL	Noolean flag that SNOTEL data is available for this site_id and year
35		Aggregated normalized precipitation for the day before issue date
36		Latest normalized snow-water-equvalent value for date before issue date
37		Maximum of snow-water-equivalent values from the start of waterseason to the day before the issue date
38		Normalized elapsed time from the start of waterseason
39		Normalized elapsed time from the start of forecast season

Table 1. Features details used for the final model

Uncertainty Quantification

Our first attempt to handle uncertainty was to predict the full probability density function (in discretized version), but we were unable to correctly estimate posterior probability for each situation.

So we decided to estimate 3 target percentiles directly by prediction of these 3 values as independent predicted values.

To force model to correctly predict needed percentiles we used next loss function:

$L_{total} = \frac{L_{10} + L_{50} + L_{90}}{3}$.

Each component of the loss is the quantile loss[5] $L_x = \frac{2}{N}\sum_{i}^{N}(x * max(0,\ t - p_x) + (1 - x) * max(0,\ p_x - t))$, where p_x - prediction for the x percentile, t - target value.

Training and Evaluation Process

We used a repeated k-fold cross-validation strategy based on years to prevent overfitting and make the model more robust[6]. In the final solution we used 5-fold cross-validation repeated 5 times. Mean values of all scores for each validation set used as a metric for local evaluation.

So we trained 25 models with different train years. At the inference we used an ensemble of all models and calculated the median value for all predictions for 10th, 50th and 90th percentiles independently. It's worth adding that using 5 models or even 1 model did not significantly degrade the score on the Hindcast leaderboard.

Due to the small size of the training dataset we used an augmentation approach to synthetically increase the number of samples. For each sample we randomly generated the issue date in the range from 1st of October to the end of the current season.

The recipe of our training procedure for the single network is next.

We used an Adam optimizer with the initial learning rate 0.001 and default beta values without the weight decay[4].

But we also used ReduceLROnPlateau scheduler with the patience period of 3 epochs and reduced factor 0.1, which means that we reduce learning rate ten times after 4 epochs without improvements in the validation loss.

All the training lasted for a maximum 120 epochs, but additionally stopped after 12 epochs without validation loss improvements.

After the training we saved the epoch checkpoint with the best validation loss as the training result.

All the hyperparameters were just slightly tuned by the average validation loss on the cross validation.

We also found it useful to check stats about error distribution dependent on the day of the water season (doy on Fig. 4). In our case we were interested only on the days after 92, which corresponds to the 1st of January, as this covers testing periods for both stages of the competition.

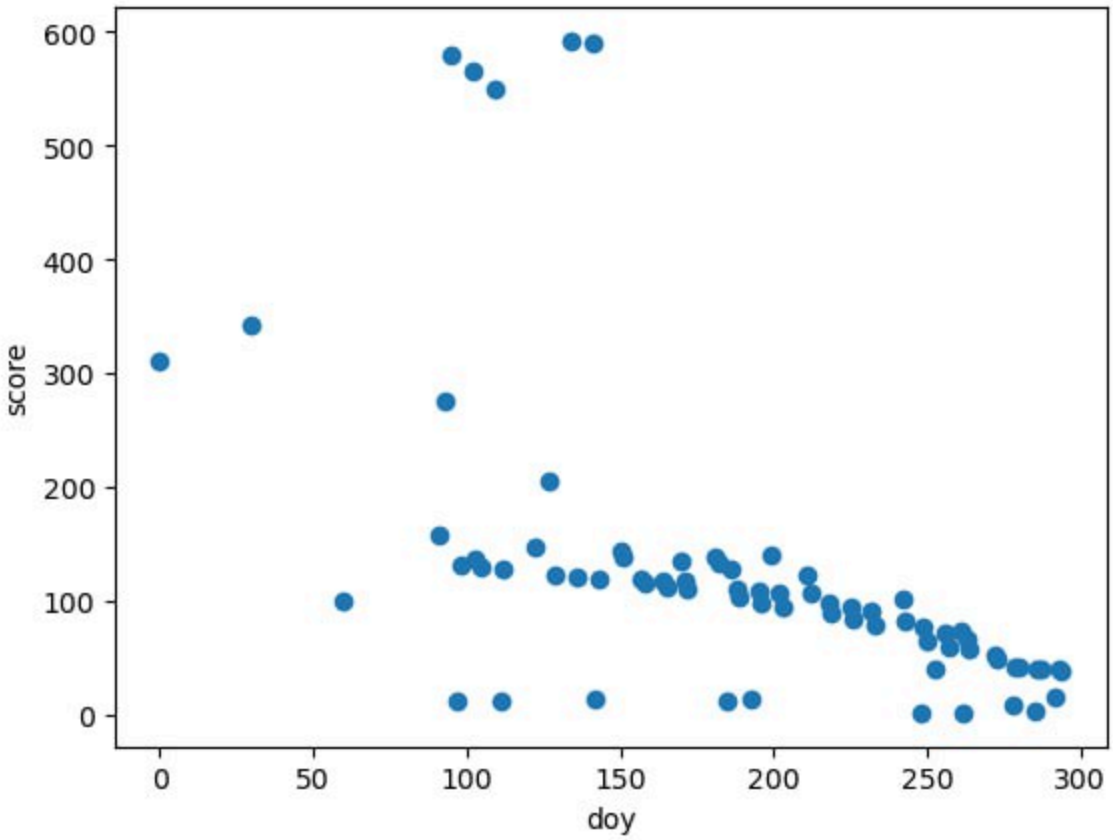

Fig. 4 The distribution of validation scores depends of elapsed days from start of waterseason

Machine Specifications

We used an ordinary PC with Ubuntu OS to train the model. Due to using the lightweight model the GPU is not required for the training and inference process.

The solution is implemented using the Python language and several open-source libraries (PyTorch, NumPy, pandas, scikit-learn, pytorch-lightning, tqdm)

References

1. Agarap A., Deep learning using rectified linear units (relu). arXiv preprint arXiv:180308375. 2018.
2. Harpold, Adrian & Sutcliffe, Kent & Clayton, Jordan & Goodbody, Angus & Vazquez, Shareily. (2016). Does Including Soil Moisture Observations Improve Operational Streamflow Forecasts in Snow-Dominated Watersheds?. JAWRA Journal of the American Water Resources Association. 53. 10.1111/1752-1688.12490.
3. Hinton G. E., Srivastava N., Krizhevsky A., Sutskever I., Salakhutdinov R.R., Improving neural networks by preventing co-adaptation of feature detectors, arXiv preprint arXiv:1207.0580, 2012.
4. Kingma D.P., Ba J., Adam: A Method for Stochastic Optimization, International Conference on Learning Representations (ICLR), 2015.
5. Koenker R., Galton, Edgeworth, Frisch, and prospects for quantile regression in economics, 1998.
6. Kohavi R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence, 1995.
7. LeCun Y., Boser B., Denker J.S., Henderson D., Howard R.E., Hubbard W., Jackel L.D., Back-propagation applied to handwritten zip code recognition. Neural Computation, 1989, 1(4):541–551.
8. Rumelhart D., Hinton G.E., Williams R.J., Learning Internal Representations by Error Propagation, Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundation. MIT Press, 1986.
9. Srivastava N., Hinton G.E., Krizhevsky A., Sutskever I., Salakhutdinov R., Dropout: a simple way to prevent neural networks from overfitting, The journal of machine learning research 15 (2014), 1929–1958.
10. U.S. Department of the Interior | U.S. Geological Survey. Title: USGS Water Data for the Nation Help. URL: https://help.waterdata.usgs.gov/
11. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I., Attention is All you Need, Advances in Neural Information Processing Systems 30, 2017.