Carnegie Mellon University

Drought and Water Price Simulation

Decision Analytics for Business and Policy

14 Dec 2022

Contributions

Sebastian Dodt	economic analysis, machine learning pipeline	25%
Sara Khosshal	Policy analysis, simulation problem formulations	25%
Karma Mroueh	simulation pipeline and results analysis	25%
Shantanu Samant	data cleaning, feature engineering and model selection	25%

Code and data available on GitHub:
github.com/sebdodt/drought-policy-optimization

Resources used:

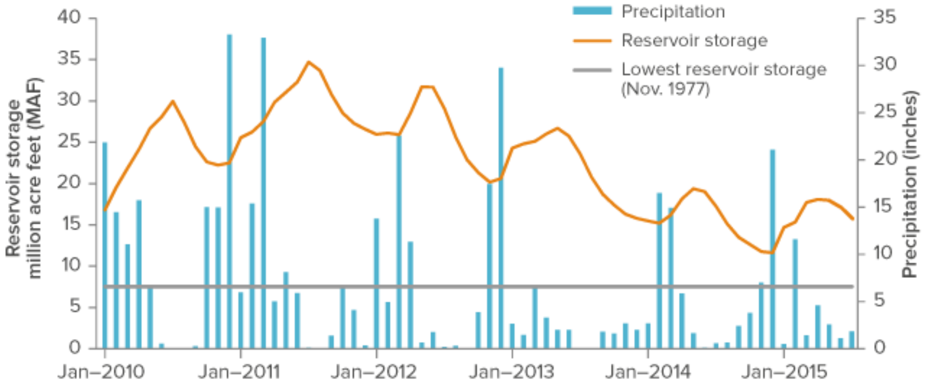
- Data from weather and water services (see data summary chapter)
- Python libraries (see requirements.txt file in our GitHub repository)
- Economic and policy studies (see Bibliography)

Problem Statement

In recent years, droughts in California have become increasingly common, with the past 22 years comprising the state's driest period in over a millennium (Klein, 2022). As water supplies decrease, California has faced significant struggles coping with the population's growing water demands.

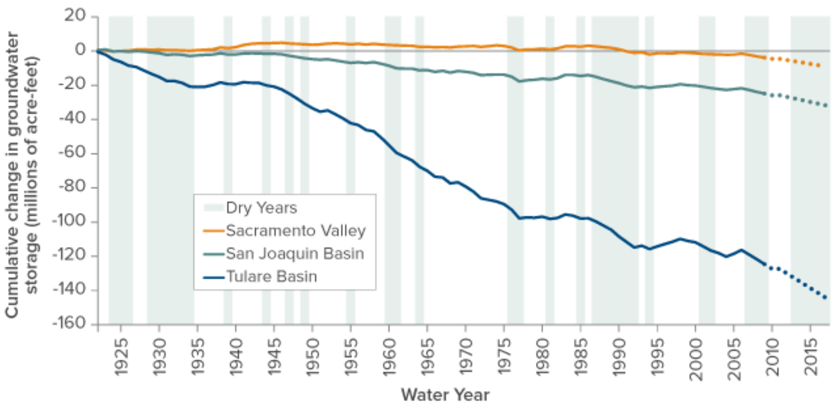
During times of low precipitation and high temperatures, California relies heavily on water reservoirs and groundwater basins as primary sources for water (Hanak et al., 2015). As droughts have intensified throughout the years, however, both these reserves have become significantly depleted (Hanak et al., 2015). Figure 1 shows how a gradual decrease in annual precipitation has led to decreases in reservoir storage, which have been continuously decreasing since 2011 (Hanak et al., 2015).

Figure 1: Precipitation and surface water reservoir levels



Groundwater basin supplies, while more resilient to the drought, have also witnessed a decline over time. Across different regions in California's southern Central Valley, particularly in the Tulare and San Joaquin Basin, there is a clear decrease in groundwater storage, and similar trends of depletion are expected to continue in years to come (see Figure 2) (Hanak et al., 2015).

Figure 2: Cumulative change in groundwater storage



The water shortage has also had a significant impact on the economy, with an estimated total of around $1.7 billion in direct and indirect costs in 2022 alone (Klein, 2022). It has also resulted in the loss of over 14,000 jobs and a record high water price of $1,144.14 per acre-foot (Klein, 2022).

As water prices continue to rise due to worsening climate conditions and shortages, water will become increasingly inaccessible to low-income groups in California. In this report, we predict the levels of water expected in the future, factoring in relevant variables (presented in the Data Summary), and attempt to answer the following question: "How do we allocate and price water in a manner that ensures equity and causes minimal damage to the economy, given limited supplies and growing demand?"

Analytical Formulations

Our model aims to identify a strategy for equitable water allocation. In a standard economic model, when supply decreases, the equilibrium price increases; as water becomes increasingly scarce in California, prices are likely to continue increasing, making water inaccessible to lower-income groups. To remedy this, we present the following policy recommendation: establishing a water consumption limit beyond which households are required to pay a higher price for water. The higher fee serves as somewhat of a penalty for households that consume more than what is recommended of them, as well as a financial deterrent from unnecessary consumption. Additionally, if set correctly, such a fee can remove some of the financial burden from low-income households and place it onto higher income households, who are able to cover it comfortably. In establishing such a model, it is necessary to define and identify three parameters:

- *Threshold*: the water consumption limit beyond which consumption becomes more expensive
- *Water price 1*: the optimal price to charge users, so long as their consumption is below the threshold
- *Water price 2*: the optimal price to charge users once their consumption surpasses the threshold

The model described above aims to minimize costs on the economy, while maximizing social benefits through equitable water distribution. To explore this relation, we have defined three measures of equity, which enable us to quantify and measure the "equity" created by each scenario. These three measures consist of:

1. Percentage of overall water expenses covered by the lowest income quintile
2. Proportion of total water used by the lowest income quintile

3. Percentage of average water price for the lowest income quintile relative to the highest income quintile

This report focuses on predicting water levels in California, as well as attempting to identify the ideal prices and thresholds to allocate water equitably.

Data Summary

In this project, we assume that water production companies automatically regulate water supplies during times of scarcity, based on the amount of water "naturally" available at each time frame of the month. Among the key requirements for the project, we needed a data set that encapsulated the impact of weather factors on the amount of water produced for commercial and agricultural distribution in California. Since this dataset was not readily available, we merged two different datasets: water production data from 2014 to 2022 (supply.xlsx) and weather data (weather.csv) for all counties in California since 2014 (data from Waterboards California, 2022; Visual Crossing Cooperation, 2022). Both datasets consist of time-series data (granular monthly data recordings in "supply.xlsx", and granular daily level data recordings in "weather.csv").

Pre-processing and merging the two datasets

The granularity of both datasets is changed to monthly aggregates. Monthly water production data is aggregated to represent yearly water production data in the supply.xlsx data set in order to reduce seasonality effects. In a second step, we aggregate all water supplies within one county and divide the total water production by the size of the population served.

The daily weather data from the weather.csv dataset is aggregated on the county and month-level. In our implementation of the feature engineering, we take the averages across multiple months, as well as minima and maxima and use them as features. The particular temporal splitting of this time-series data is described in more detail in the data dictionary submitted along with this report.

The two datasets are merged by county name and month.

Implementation

Machine Learning

We implemented our final pipeline in two interlinking Python modules that can be run in succession from one main file. In this section, we will describe the pipeline's different steps.

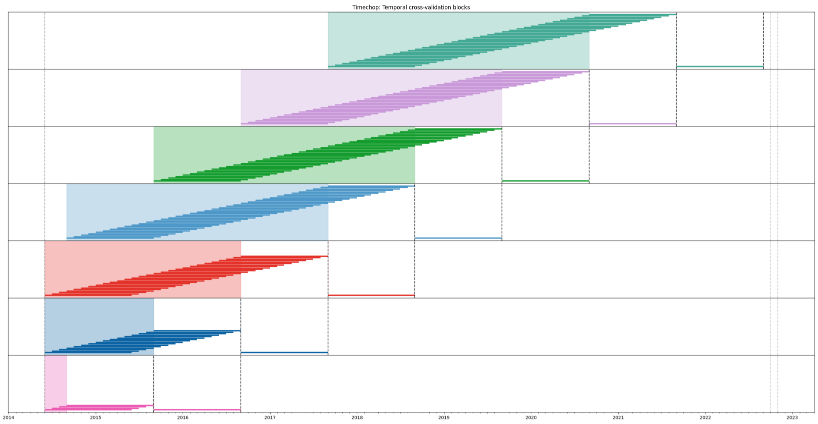
Training data creation and validation

The machine learning training uses a concept called "temporal blocked cross-validation" (Roberts et al., 2017). Since we are dealing with time series data, we cannot generate time splits and randomly cross-validate month-county combinations because data from different locations is correlated within the same time frame, known as temporal auto-correlation. If we did regular k-fold cross-validation, it would allow the model to use future data from a nearby locations, which is not available in a real deployment, to make its predictions. Randomly generating cross-validation sets would result in overestimating the testing accuracy and be a poor estimate of the true error that it would have in a setting outside of the lab.

To create training and testing datasets, we created several time splits. For each time split, we defined a “cutoff date”. This is simulating the deployment setting where we would cutoff at the current date, use all the data we can find about the past, and make a prediction. In our pipeline, the temporal configurations can be defined in a file called `config.yaml`. The parameters we set are:

- Feature start time: The first date for which we have data from all sources. (2014-01-01)
- Feature end time: The last date for which we have data. (2022-11-01)
- Label start time: The earliest date for which we have label data (2014-06-01)
- Label end time: The last date for which we have label data (2022-10-01)
- Model update frequency: The frequency at which we would retrain our model in a real-world setting. This parameter determines the time difference between the cutoff dates (set to three months)
- Training cutoff date frequencies: Within our training dataset, we create additional splits enabling our dataset to use both the data right before the cutoff date and from earlier points.
- Max Training histories: This parameter determines how far back the training data splits are allowed to go as measured from the cutoff date (set to three years to avoid using data older than the data used around our cutoff date)
- Training label and testing label timespan: The time over which the label is aggregated (set to 12 months to reduce any seasonality effects and to predict annual water production)

Figure 3: All training and testing splits



This plot shows our seven validation sets. Each set is 12 months apart. On the right of each validation set we can see the span over which the label was aggregated for our testing set. On the left, we can see all time spans over which our training label was aggregated. There are multiple aggregations that are one month apart within each training set to allow our model to use more data. Models are trained separately for each validation set and performance on each is exported to a log csv file.

Generating labels and features

After setting the parameters, we run our pipeline, which begins by creating labels for different time splits. First, it performs data cleaning, converting date tables to datetime format and splitting values where multiple counties were served. It then matches each county name to a unique FIPS code using the `addfips` library. The library successfully matches all available counties in our dataset.

This process is repeated for the feature space. After being read in, the data was split using our self-built time splits module. We then generated features for each time split (detailed above in the data summary section). We generated all features in a loop through training and testing splits and looping through all available data columns. Two features (snow and snow depth) had missing values, which we assumed was due to there being no snow for that county at that day and impute a zero value. We then checked our label space for any zero values and dropped county-date splits for which a label is missing (less than 1% of our counties). Lastly, we stacked the training data within each cutoff date group as described in the section above.

Training and testing

We first defined the models we wanted to train. Throughout our project, we tried multiple models: Linear models, regularized regression with L1 and L2 penalty terms, Elastic Net, Bayesian Ridge, Random Forests, Gradient Boosting, and XG Boost. Because L1 (Lasso) regression with a penalty term of alpha = 1 worked best and was more time-efficient, other models are commented out but can be added into the model space at any time. All models used come from the sklearn library or the XG Boost library. After training, each model is scored on testing data whose labels come from a period that is temporally after its last training observation. The test data is stored in a multi-dimensional array and exported to a log folder. The pipeline extracts the best-performing model across all time splits and returns its name, hyperparameters, and average root mean squared error (RMSE). It also returns the RMSE of our best-performing baseline, which predicts the average value for the respective county.

Prediction

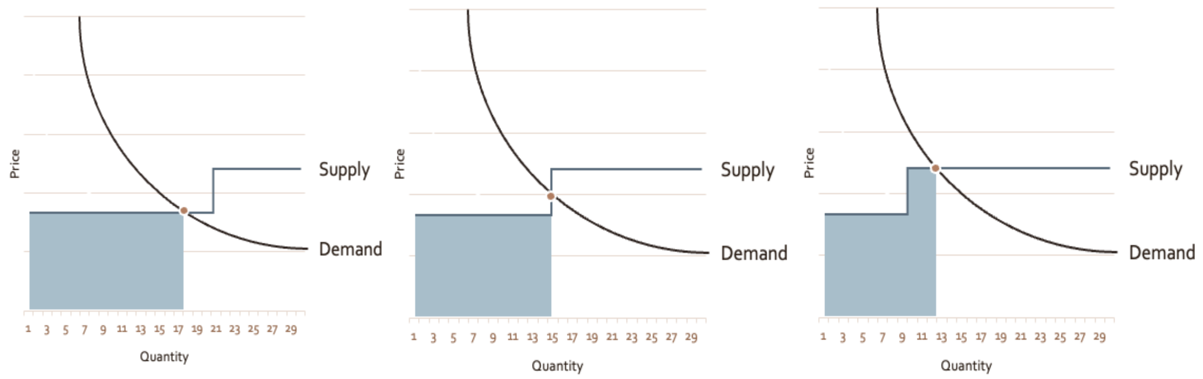
After finding the best model, we generated features for the last available time in our dataset and fed them into the model object that comes from the last time split of best model. We then generated a prediction for each county's next 12 months (measured from the last available feature data in our data sources, i.e., Oct 2023). This prediction is returned to the main module and fed into the second part of our pipeline. A policymaker could select a specific county for which to set the water price. Here, however, we are setting the price for a fictitious town for demonstration purposes, so we take the average per capita water production across all counties.

Simulation

To simulate the outcome of different price scenarios in our fictitious town, we first define the number of households: this is currently set to 5000 but can be changed by the user. The total available amount of water is defined as the amount of water per person predicted from the ML timeline multiplied by the population. Next, we defined the policies to test: *water price* 1*, water price 2,* and the *threshold*.

We then looped through all combinations of the three parameters and ran our simulation. In each loop, we defined our village as a 5000-household town that currently has the average California use of water, the average current cost of water, the median income, and income distribution of water. Using studies specifically conducted in California, we can set all parameters to receive a demand curve for each individual, where we assumed the average elasticity to be -0.435 (Bruno and Jessoe, 2021). The demand depends partly on individuals' incomes but also contains randomness and fixed amounts that represent a minimum that every household would always demand.

Figure 4: A consumer demanding of less than the threshold amount (left), exactly the threshold amount (middle), and more than the threshold amount (right)



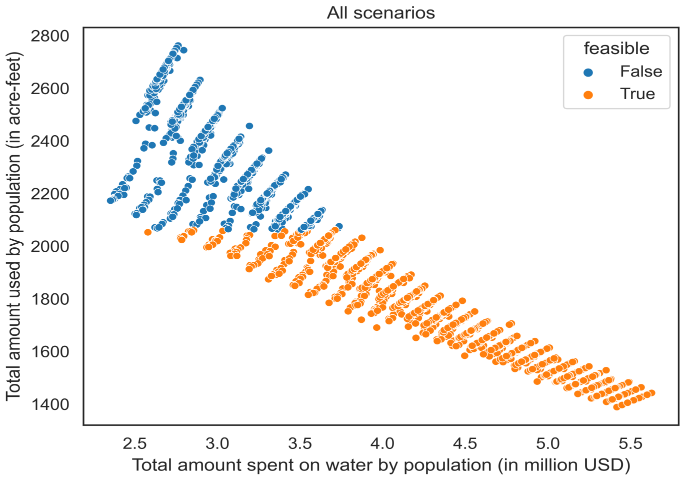
In a second step, we simulate individuals' demands given each price and threshold (Figure 4). We subsequently score each scenario through the metrics defined in the Analytical Formulations chapter. The scoring either relates to our equity goal of making water as accessible as possible for the bottom income bracket or to our efficiency goal of reducing the cost on the economy. Each combination of equity and efficiency goals is then plotted in Figure 6 though Figure 8. The plots are exported into an output folder.

Analysis

Through the implementation process defined above, we investigated the costs and the social equity produced by each of 18,000 scenarios where each scenario was a combination of a lower price, higher price, and a threshold that determined the per-household amount that was available at the lower price. The simulations we ran yielded the following plot (Figure 5), representing the variation in total amount used of water by the population as a function of total cost. Consistent with standard economic theory, we witness that demand for water falls as water prices (here, represented as total cost on the economy) rise. In addition to demonstrating the impact of pricing on water consumption, this plot also depicts the feasibility of different scenarios: dots represented in orange are feasible given the predicted water supply, while those represented in blue surpass the predicted available water supply. While these scenarios may be infeasible, we chose to model them as well to build a comprehensive model that fully and accurately portrays the relationships between different factors.

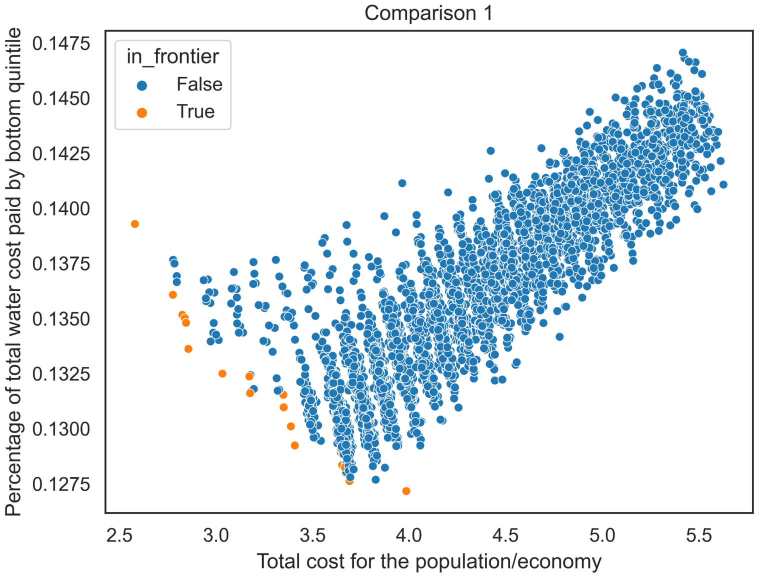
The optimal state for Figure 5 consists of all the scenarios at the top left corner, where people can purchase more water while minimal damaging the overall economy. However, under drought conditions, which are expected to persist in California, managing such demand without highly burdening the economy is infeasible.

Figure 5: Demand for Water as a Function of Price



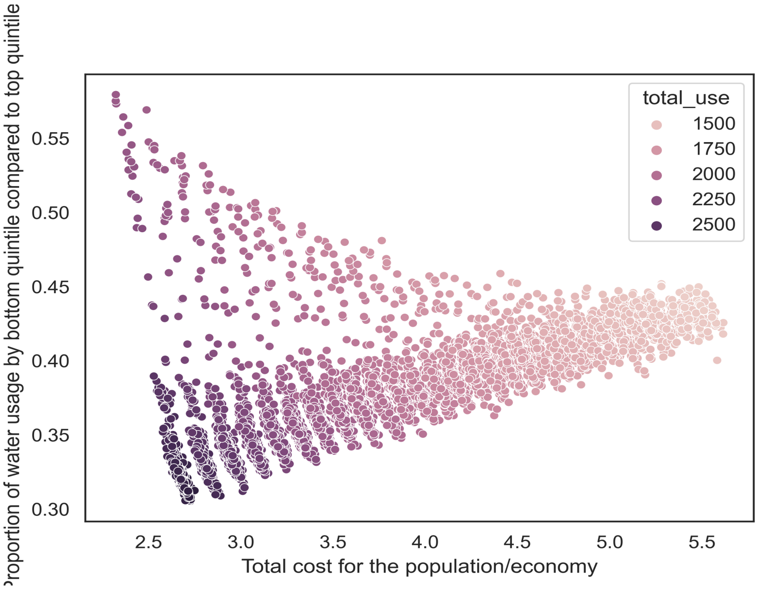
Varying the different parameters described above (*threshold, water price 1, water price 2*), we obtained the following graph (Figure 6), depicting the percentage of total costs (all costs associated with the purchase of water) covered by the bottom income quintile as a function of the overall costs on the economy.

Figure 6: Equity Metric I: Water cost of bottom quintile



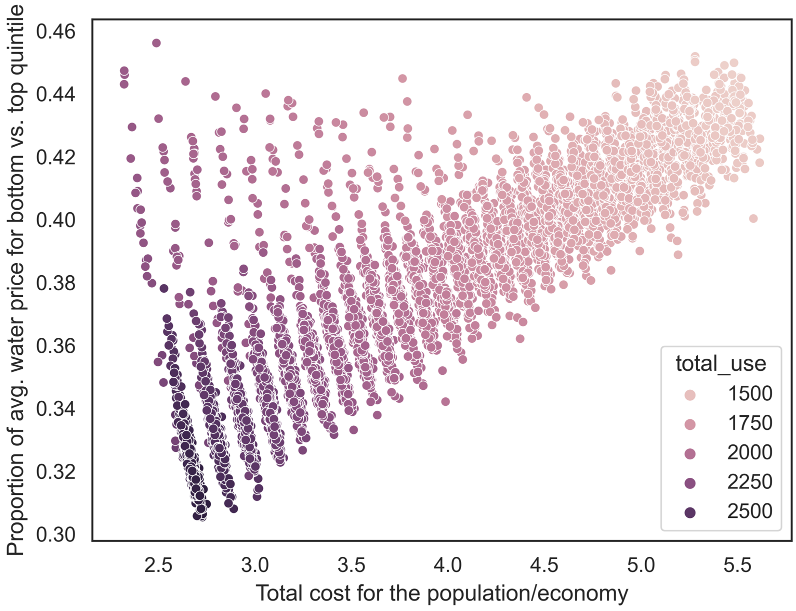
The dots in orange represent the scenarios on the efficient frontier: scenarios in which the cost on the economy is minimized, while equity is maximized. The orange dot at the top left corner represents the most cost-efficient scenario for the entire population, yet it is placing a larger financial burden on the bottom income quintile. The most equitable scenario, on the other hand, is depicted by the orange dot with coordinates (4.0; 0.1275): more damage to the economy, but less financial burden on the bottom income quintile. Figure 7, presented below, shows us the proportion of water used by the bottom quintile relative to the top quintile in different scenarios (consisting of a different *threshold, water price 1,* and *water price 2)*, as well as the associated over all costs on the economy.

Figure 7: Equity Metric II: Water Usage by bottom quintile compared to top quintile



We would want to choose the option that maximizes total use (subject to the water supply constraint), while minimizing total costs on the economy and ensuring equitable water allocation. In this scenario, water allocation is most equitable when water consumption per capita remains consistent across different income quintiles. Looking at this case, there appears to be an optimal scenario that satisfies both sides, which is found in the top left corner.

Figure 8: Equity Metric III: Comparison of per gallon water prices for different quintiles



One of our goals is to minimize the financial burden placed on lower-income households in purchasing water by “subsidizing” *water price* 1. One way in which this can be achieved is to place more of the financial burden on higher-income households: by setting a higher *water price 2* that will apply for everyone consumer more than the threshold amount, we can compensate for any lost revenue generated by the subsidization of *water price 1*. Figure 8 demonstrates the scenarios possible at different combinations of parameters, as a function of total cost on the economy. Here again, we hope to minimize the total cost on the economy, while ensuring that more of the financial burden falls onto the top income quintile rather than the bottom. Our optimal situation would therefore lie in the bottom half of the graph but would depend on the amount of water consumed (once again depicted by different purple shades).

Conclusion

Climate change and a growing population are likely to further limit the availability of water in California. With growing demand and decreasing supplies, the equilibrium price will increase.

It is therefore the responsibility of policymakers in California to ensure that water remains accessible to low-income groups. When determining the optimal parameters, policymakers must consider the trade-off between equity and economy and decide which matters most, as demonstrated by our analysis. As data scientist, we do not think that it is our place to define the best trade-off between the economic and equity metrics highlighted above. Our pipeline therefore serves as a menu for policymakers to consider, but it does not provide an “optimal” policy, as we believe that there is a certain degree of subjectivity and human consideration necessary to tackle this problem.

Bibliography

Bruno, E. M. & Jessoe, K. (2021) Using Price Elasticities of Water Demand to Inform Policy. *Annual Review of Resource Economics*. [Online] 13 (1), 427–441.

Hanak, E. et al. (2015) What If California's Drought Continues? *Public Policy Institute of California*.

Klein, K. (2022) Drought has already cost close to $2 billion and 14,000 jobs, and it's likely not over yet. *KVPR - NPR For Central California*.

Roberts, D. R. et al. (2017) Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*. [Online] 40 (8), 913–929.

Visual Crossing Cooperation (2022) Weather Data & API. *Global Forecast & History Data*. [online]. Available from: https://www.visualcrossing.com (Accessed 10 October 2022).

Waterboards California (2022) Water Boards Data and Databases. *Water Conservation Portal*. [online]. Available from: https://www.waterboards.ca.gov/resources/data_databases/ (Accessed 10 October 2022).

Data Dictionary

The water supply data was obtained from Waterboards California. (2022). Water Boards Data and Databases. *Water Conservation Portal*. https://www.waterboards.ca.gov/resources/data_databases/

The weather data was obtained by querying the Weather API of the following source: Visual Crossing Cooperation. (2022). Weather Data & API. *Global Forecast & History Data*. https://www.visualcrossing.com

We feature engineered the data to produce the following set of independent and dependent variables.

Table 1: Data dictionary for the combined dataset

Label: “Dependent variable”				
Column Name.	**Unit**	**Data type**	**Description**	**Extra Notes**
Y	Gallons/Capita	float	Represents the average amount of water produced by each county for each person in the county	This label is generated by dividing the total amount of water produced in each county by dividing it by the population size of the county.

Features: “Independent Variables”	
Feature name	**Description**
county	Represents one of 50 counties in California that have water production facilities (out of 58). **Extra note:** processed through one-hot encoding, this variable is converted into 58 dummy variable features.
1) tempmax_avg 2) tempmax_max 3) tempmax_min	For a given county across a particular time span (e.g., 3 years, depending on the configurations set), these variables indicated the average daily temperature minimum, as well as the highest and lowest temperature maximum of any month.
1) tempmin_avg 2) tempmin_max 3) tempmin_min	For a given county across a particular time span (e.g., 3 years, depending on the configurations set), these variables indicated the average daily temperature minimum, as well as the highest and lowest temperature minimum of any month.

1) feelslikemax_avg 2) feelslikemax_max 3) feelslikemax_min	For a given county across a particular time span (e.g., 3 years, depending on the configurations set), these variables indicated the average daily feels like maximum, as well as the highest and lowest temperature maximum of any month.
1) dew_avg, 2) dew_max, 3) dew_min	For any given county these three features store the average, maximum and minimum values, respectively for daily recorded dew level indicators for any given month
1) feelslikemin_avg 2) feelslimemin_max 3) Feelslikemin_min	For a given county row, these features store the average, maximum and minimum of “daily recorded minimum feels-like- temperature value” of a month in that county
1) humidity_max 2) humidity_min 3) humidity_avg	For a given county row, these features store the maximum, minimum and averages of “daily recorded humidity index value” of a month in that county
1) precip_avg 2) precip_max 3) precip_min	For a given county row, these features store the maximum, minimum and averages of “daily recorded precipitation levels” of a month in that county
1) precipprob_avg 2) precipprob_max 3) precipprob_min	For a given county row, these features store the maximum, minimum and averages of “daily recorded precipitation probability levels” of a month
1)precipcover_avg 2)precipcover_max 3)precipcover_mon	For a given county row, these features store the maximum, minimum, and average of “daily recorded precipitation coverage levels” of the respective month. Extra note: precipitation coverage is a measure that records the percentage of the day throughout which rainfall fell. A precipcover value of 0.5 denotes that rainfall occurred for 50% of the entirety of the day.
1) snow_avg 2)snow_max 3)snow_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded snowfall index” of that month. Null values are imputed with zero.
1)snowdepth_avg 2)snowdepth_max 3)snowdepth_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded snow depth levels” of a specific month. Null values are imputed with zero.
1)windgust_avg 2)windgust_max 3)windgust_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded wind gust speeds” of a month.
1)windspeed_avg 2)windspeed_max 3)windspeed_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded wind speeds” of a month.

1)sealevelpressure_avg 2)sealevelpressure_max 3)sealevelpressure_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded sea level pressures (in millibars)” of a month.
1)cloudcover_avg 2)cloudcover_max 3)cloudcover_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded cloud cover indexes (in oktas)” of a month.
1)visibility_avg 2)visibiliy_max 3) visibility_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded visibility indexes (in meters)” of a month.
1)solarradiation_avg 2)solarradiation_max 3)solarradiation_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded solar radiation in watts per sq. meter” of a month.
1)uvindex_avg 2)uvindex_max 3)uvindex_min	For a given county row, these features store the maximum, minimum, and average of the “daily recorded solar radiation in watts per sq. meter” of a month.

Note that monthly variables are aggregated using a temporal blocked cross-validation system described in implementation section of our report.