COEQWAL · KEY OUTCOME METHOD DOCUMENTATION

Reservoir storage

Reproduced as written by the Reservoir storage team, in the team's own terminology.

Key outcome name: Reservoir storage

Authors: James Gilbert[1,2*], Dino Bellugi[3], Abhinav Sharma[1], Khandaker Iftikhar Islam[1]

*Corresponding author:** James Gilbert

[1] University of California, Santa Cruz, Institute of Marine Sciences' Fisheries Collaborative Program, 1156 High Street, Santa Cruz, California 95064, U.S.A.

[2] Fisheries Ecology Division, Southwest Fisheries Science Center, National Marine Fisheries Service, National Oceanic and Atmospheric Administration, 110 McAllister Way, Santa Cruz, California, 95060, U.S.A.

[3] University of California, Berkeley, Department of Environmental Science, Policy and Management, 130 Mulford Hall #3114, Berkeley, CA 94720-3114

Background

Reservoirs typically store water during the winter and spring, when most of the precipitation and runoff occurs across California, and typically release that water later in the summer and fall for use by farms and cities. Some reservoirs, like San Luis, are situated away from a main river or stream and serve as temporary storage on the conveyance network. The amount of water stored in reservoirs at the beginning of the dry summer season, typically at the end of April, provides an indicator of the water supply available for use. Depending on their location on key rivers or conveyance networks, some reservoirs can provide water to many downstream use points while others may serve a more limited geographic extent.

For this key outcome evaluation, we select a subset of reservoirs from across the CalSim3 domain that reflect a range of hydrologic conditions, management purpose, managing agency, and configuration within the conveyance network. We evaluate the simulated storage in the selected reservoirs in comparison to the storage record observed over the last approximately 30 years. Doing so places the simulated storage patterns in the context of recent reality rather than focusing on absolute storage amounts. While comparisons to total reservoir capacity can sometimes be helpful, many reservoirs may have flood control constraints that limit spring storage to levels below the physical capacity. Using capacity as a reference standard against which to evaluate storage performance might provide misleading information in these cases. Comparison to typically-observed storage levels (e.g. historical average for a given date) avoids this issue and has an established precedent in common reservoir storage summary publications and websites (e.g. CDEC: https://cdec.water.ca.gov/resapp/RescondMain).

Each CalSim3 simulation produces a 100-year time series with a monthly time step, meaning that there are 100 end-of-April storage values for each reservoir. A simple measure of central tendency - like mean or median - provides one summary indicator of reservoir storage behavior, but may miss important information about the frequency with which a reservoir is at more extreme (full, empty) conditions. A more comprehensive measure incorporates characteristics of the full distribution to indicate how often a reservoir is at varying levels of fullness. Put another way, the average storage calculated for a reservoir that is completely full 50% of the time and completely empty the other 50% of the time would be indistinguishable from a condition where the reservoir maintains a steady half-full storage the entire period, despite these two conditions likely having very different ramifications for downstream users. The framework described in this document aims to avoid the ambiguities that might arise from assigning an outcome level based on a single percentile or central tendency value by combining information from the upper and lower parts of the April reservoir storage distributions.

The concept underlying the reservoir outcome level calculation can be illustrated using an exceedance chart, a figure commonly used to summarize CalSim3 results from a frequency perspective rather than as a direct time series. The selected time series data (end-of-April reservoir storage in this case) are ranked from highest to lowest and assigned an empirical exceedance probability (often referred to as the Weibull plotting position). Plotting the ranked storage values on the y-axis against the exceedance probabilities yields a curve that can be readily evaluated alone or in combination with other scenarios. An example of such a plot can be seen in Figure 1 below, with the solid black line representing a hypothetical reservoir storage distribution. The outcome level formulation involves first setting high and low threshold reservoir storage values (examples shown as dashed black lines in Figure 1) that divide the storage distribution into three zones - everything at or above the high threshold, everything at or below the low threshold, and everything in between. The frequency with which these thresholds are met or exceeded for a given scenario result can be determined by connecting a vertical line from where the threshold and storage curve intersect down to the x-axis. The fraction of time that the high storage threshold is met or exceeded is simply the exceedance probability or exceedance frequency at that location on the x-axis (marked by an ***A*** in Figure 1). The fraction of time that the low storage threshold is not met is determined by subtracting the exceedance probability at point B from 1.0 (or 100% if using percentages).

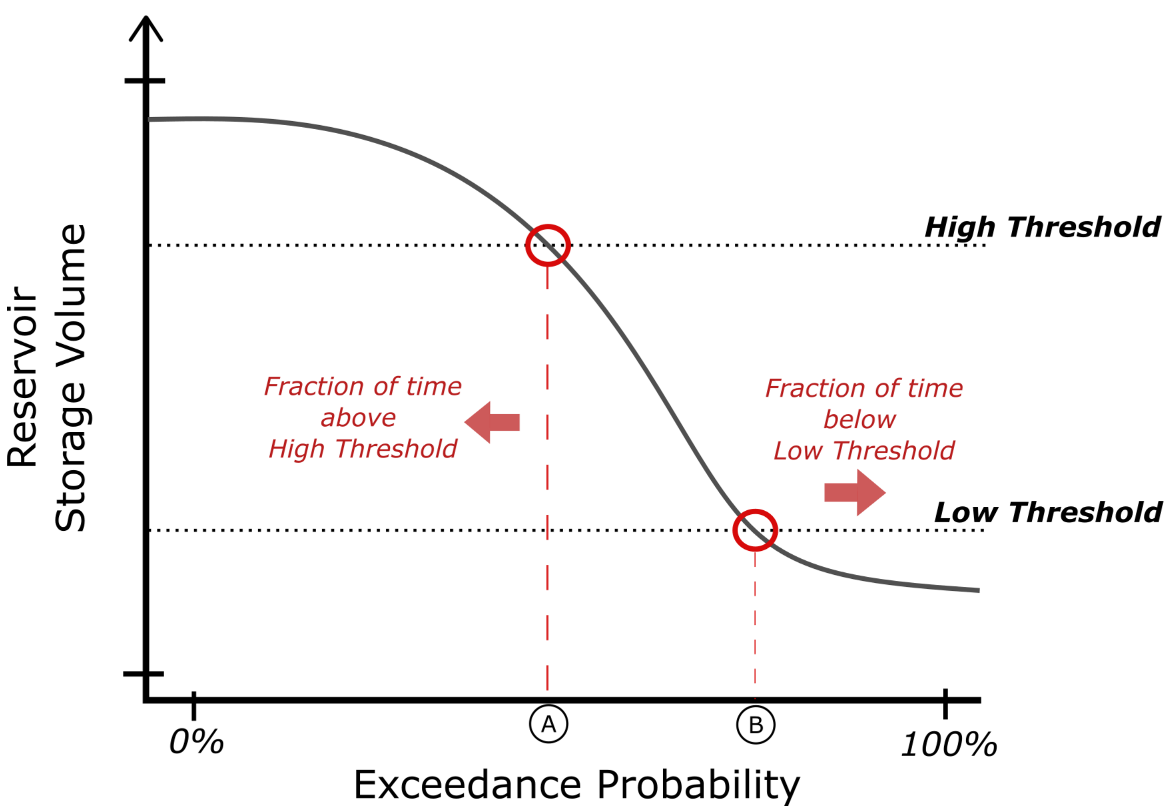

Figure 1. *Illustration of the use of thresholds to divide a reservoir storage distribution (represented using exceedance probability) into high and low zones and their corresponding exceedance and non-exceedance frequencies. The projection of the intersection of the dashed horizontal black threshold lines with the reservoir storage line (solid black line) vertically to the exceedance probability axis determines the fraction of the time (0-100%) that the high and low thresholds are met or exceeded ('A' and 'B' markers on the plot). The fraction of time that the storage is below the low threshold (area to the right of the 'B' marker) is calculated by subtracting the exceedance value at the low threshold location from 100%.*

The reservoir outcome level formulation assumes that the preferable condition for a reservoir is one in which high reservoir storage occurs frequently and low reservoir storage is rare. The thresholds for setting these high and low storage conditions are extracted from historical reservoir storage data such that the best outcome level values (i.e. closest to a value of 1.0) would represent a condition in which the reservoir has some combination of more frequent high storage and less frequent low storage than has occurred in the recent past. Increments of decreasing high storage frequency and increasing low storage frequency define the boundaries for the discrete outcome level values (1-4), with the distance from those increments determining the continuous outcome level value.

Note that the formulation assumes that each of these conditions (frequency of high and low storages) are equally important – frequent high storage with frequent low storage is less preferable than a condition with the same frequency of high storage but very rare low storage. In some cases low storage at a particular reservoir may have a disproportionate negative outcome that may suggest a different balance. Such location-specific adjustments were not made for this analysis in favor of consistency of interpretation, but could be implemented as a variation of our approach.

Locations of Interest

We select a limited subset (8) of the numerous reservoirs included in CalSim3 for the calculation of outcome level values. The selected reservoirs represent a substantial fraction of storage volume for the Central Valley Project and State Water Project as well as reservoirs that provide important additional functions (flood control, offstream storage, support for Delta regulatory compliance) across the Central Valley region. The selected reservoirs are listed in Table 1 along with their CalSim3 and California Data Exchange Center (CDEC) identifiers. A map showing the location of these reservoirs across the Central Valley is shown in Figure 2.

Table 1. *Listing of reservoirs for which a spring storage outcome level value is calculated. The reservoir name is mapped, by row, to identifiers used in CalSim3 and the publicly accessible California Data Exchange Center (CDEC) database.*

Reservoir Name	Region	Agency	CalSim3 ID	CDEC Station ID
Trinity	SAC	CVP	S_TRNTY	CLE
Shasta	SAC	CVP	S_SHSTA	SHA
Oroville	SAC	SWP	S_OROVL	ORO
Folsom	SAC	CVP	S_FOLSM	FOL
New Melones	SJR	CVP	S_MELON	NML
Millerton	SJR	CVP	S_MLRTN	MIL
San Luis (State)	SJR	SWP	S_SLUIS_SWP	LUS
San Luis (Fed)	SJR	CVP	S_SLUIS_CVP	SLF

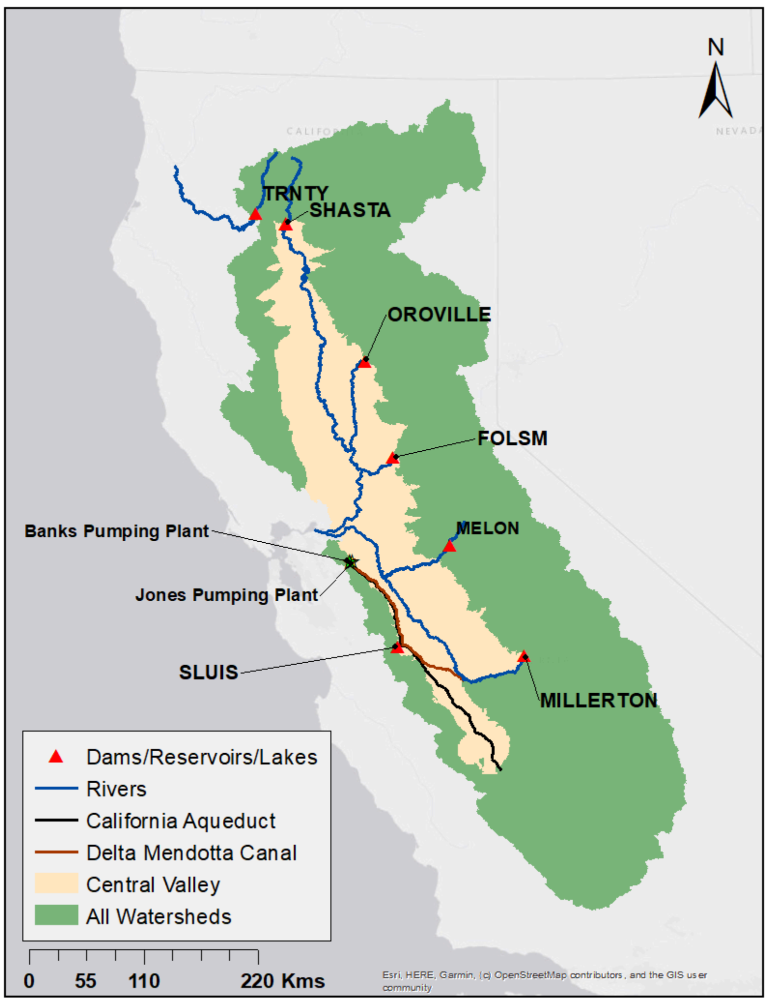

Figure 2. *Map showing the location of the 8 main Central Valley reservoirs included in this analysis. Note that the SWP and CVP portions of the San Luis reservoir are represented separately.*

Methodology

Reservoirs are assigned storage outcome levels based on the frequency with which April storage conditions satisfy predefined high and low storage performance standards. Rather than assigning outcome levels directly from storage magnitude, the method evaluates how consistently each scenario maintains favorable storage conditions over the simulation period.

Discrete key outcome level

Discrete outcome level thresholds are defined using two criteria: the minimum fraction of years with April storage at or above a high storage threshold, and the maximum fraction of years with April storage at or below a low storage threshold. These criteria are converted into a combined distance metric, where lower values indicate performance closer to the ideal condition of always exceeding the high threshold and never falling below the low threshold. Level 1 represents scenarios closest to this ideal condition, while higher outcome levels represent increasing distance from the desired storage performance.

The storage thresholds on which the subsequent outcome level calculations are based are set based on the distribution of observed April reservoir storages as provided by the CDEC database. The data for each reservoir (see Table 1) was downloaded and filtered to yield the end-of-April storage values for the 35-year period 1990–2024. We selected this period as it represents a reasonably large sample of recent hydroclimatic and operational conditions while also being uniformly available for each reservoir. The upper and lower storage thresholds for each reservoir are then set as the 50th and 20th percentiles from this filtered April storage dataset.

We calculate the fraction of years above the high storage threshold (frac_top) and the fraction of years below the low storage threshold (frac_bot) for all end-of-April storage values extracted from each scenario. These fractions are combined into a single performance distance metric, defined as D = (1 - frac_top) + frac_bot, which represents the total deviation from the ideal storage condition. Lower values are more desirable. The discrete outcome level is then assigned by comparing this distance value against predefined outcome level cutoffs derived from the storage standards. Lower D values indicate stronger storage performance and results in a lower outcome level assignment, while larger D values indicate poorer performance and result in higher outcome level assignments.

The outcome level boundaries as applied essentially prescribe the following incremental conditions for outcome level assignment:

- > Level 1: April storage should be above the upper threshold at least 50% of the time and not lower than the bottom threshold more than 20% of the time
- > Level 2: April storage should be above the upper threshold at least 33% of the time and not lower than the bottom threshold more than 25% of the time
- > Level 3: April storage should be above the upper threshold at least 25% of the time and not below the lower threshold more than 33% of the time.
- > Level 4: Not meeting the other threshold conditions, with a maximum lower bound of 2.0

Table 2. *Listing of outcome levels and the criteria to achieve those outcome levels.*

Outcome Level	Minimum Fraction of Years Above High Threshold	Maximum Fraction of Years Below Low Threshold	Equivalent Distance Cutoff (D)
1	≥ 50%	≤ 20%	(1 − 0.50) + 0.20 = 0.70
2	≥ 33%	≤ 25%	(1 − 0.33) + 0.25 = 0.92
3	≥ 25%	≤ 33%	(1 − 0.25) + 0.33 = 1.08
4	Does not meet outcome level 1–3 criteria	No upper performance requirement	> 1.08

The predefined outcome level cutoffs and storage standards are the same for both the discrete and continuous outcome level calculations.

Continuous key outcome level

For the continuous outcome level, the code calculates the position of the scenario within its assigned outcome level range. This is done by measuring how far the scenario has progressed from the better boundary of its outcome level relative to the total width of that outcome level interval. A progress value near 0 indicates performance close to the stronger (lower-outcome level) boundary, while a value near 1 indicates performance close to the weaker (higher-outcome level) boundary. The final result is expressed within the following ranges:

- 1.00–1.99 for Level 1
- 2.00–2.99 for Level 2
- 3.00–3.99 for Level 3
- 4.00–4.99 for Level 4

The continuous outcome level value is calculated by adding this normalized progress value to the discrete outcome level number, producing a continuous score that retains the discrete outcome level classification while providing additional resolution within each outcome level.

More specifically, the continuous outcome level assignment is implemented using the combined storage distance metric (D), calculated as:

D = (1 - frac_top) + frac_bot

where:

- **frac_top** is the fraction of years where April storage is greater than or equal to the high storage threshold.
- **frac_bot** is the fraction of years where April storage is less than or equal to the low storage threshold.

The metric represents the distance from the ideal storage condition:

- > frac_top = 1: storage exceeds the high threshold every year.
- > frac_bot = 0: storage never falls below the low threshold.

Therefore, a value of D = 0 represents the best possible storage performance. Increasing values of D indicate increasing deviation from the desired storage condition. The worst possible value of D occurs when frac_top = 0 and frac_bot = 1, leading to D_worst = (1 − 0) + 1 = 2.0

The storage performance fractions are calculated as:

> **high_thresh** = thresholds[percentiles[0]]
>
> **low_thresh** = thresholds[percentiles[-1]]
>
> **frac_top** = (april_by_year >= high_thresh).mean()
>
> **frac_bot** = (april_by_year <= low_thresh).mean()
>
> **frac_mid** = 1.0 - frac_top - frac_bot

The **high_thresh** and **low_thresh** variables are storage levels (i.e. a reservoir storage volume) that divide the distribution of April values into top, middle, and bottom bins or categories. The thresholds can theoretically be set at any valid storage level for a reservoir, but are defined for this analysis based on a percentile of the observed 1990-2024 April storage record. The 'percentiles' included in the code above are the mechanism by which those percentiles are set. For this analysis, the outcome level boundaries are defined using storage standards that consist of three pairs of values:

- > the minimum required fraction of years above the high storage threshold
  - > *These are the "top" values used to set the outcome level cutoffs in the code below*
- > the maximum allowable fraction of years below the low storage threshold
  - > *These are the "bot" values used to set the outcome level cutoffs in the code below*

For Level 1, the "top" value should be the highest and the "bot" value the lowest of the set, indicating that results that meet this condition have the highest fraction of high reservoir storage along with the lowest fraction of low reservoir storage. Each subsequent outcome level incrementally reduces the "top" value and increases the "bot" value. These standards are converted into equivalent ***D*** cutoffs – with values as selected for the actual outcome level (called 'tiers' in the code) calculations shown second:

> ***General formulation:***
>
> tier1_cutoff = (1 - top1) + bot1
>
> tier2_cutoff = (1 - top2) + bot2
>
> tier3_cutoff = (1 - top3) + bot3
>
> ***As applied:***

> Tier1_top_bound (best) = (1 - 1) + 0 = 0.0
>
> tier1_cutoff = (1 - 0.50) + 0.2 = 0.70
>
> tier2_cutoff = (1 - 0.33) + 0.25 = 0.92
>
> tier3_cutoff = (1 - 0.25) + 0.33 = 1.08
>
> Tier4_low_bound (worst) = (1 - 0) + 1.0 = 2.0

Generalized code for the continuous outcome level calculation is available in Appendix S-1. Generalized code for outcome level assignment is available in Appendix S-2. All code for this implementation is available via GitHub at https://github.com/canruso/COEQWAL_V3, with the core functions available in coeqwalpackage/tier.py.

An example of the outcome level values as calculated for the baseline scenario 20 is shown as a map below, with the colors of each reservoir and label indicating the discrete outcome level assignment.

Figure 3. *Map of example reservoir outcome level values for the 8 locations of interest across the Central Valley, for the baseline scenario 20 (s0020). The reservoir outline color and label indicate the outcome level value (green = Level 1, blue = Level 2, orange = Level 3, red = Level 4).*

Guidelines for Interpretation

The continuous and discrete outcome level values can be interpreted in essentially the same manner: a value closer to 1 means that, in comparison to recent history, the reservoir tends to be fuller more frequently and near empty less frequently. Outcome level values of increasing numerical magnitude indicate incremental reductions in the frequency of high April storage and increases in the frequency of low April storage. In general, the span of continuous outcome level values between 1 and 2.99 may be viewed as bracketing the recent historical performance, with values within the outcome level 1 category (1-1.99) representing results on a spectrum from much better than recent observed conditions (closer to 1.0) to similar to recent performance (closer to 2). Outcome level values between 2.0 and 2.99 would similarly indicate gradations in performance declining from the recent conditions to moderately worse than recent (i.e. the reservoir is not quite as full as frequently as in recent decades). Level 3 and Level 4 conditions are then gradations of increasingly lower and/or more sustained low storage conditions that tend to represent a substantive shift from the recent historical record. More specifically, a Level 4 result would indicate the reservoir storage is below the lower threshold substantially more often than it is above the top threshold – in other words, the reservoir is nearly empty more often than it is full.

One important consideration with this approach is that the calculated outcome levels may be distorted in cases where the observed and simulated reservoir storage patterns do not closely align. This is the case for Millerton Reservoir on the San Joaquin River: on average, simulated April storage tends to be about 50 TAF lower than the observed historical storage while the interquartile range of the simulated and observed end-of-April storage are both approximately 150 TAF. The plot in Figure 4 illustrates this pattern -- the blue (CalSim3 simulated) and orange (observed record) lines notably diverge in the month of April.

This bias between the model and observations makes it difficult to use the threshold and frequency comparisons on which our framework is based without additional corrective factors or different methods customized for the specific reservoir. Because the outcome level methodology described herein ultimately does provide an accurate summary of the simulation results for Millerton Reservoir (simulated April storage is consistently lower than observed and will thus tend to have Level 3 or Level 4 values for many scenarios), we opted to keep the formulation uniform across all reservoirs. For Millerton Reservoir, however, the interpretation of the outcome level results must consider that model error as well as operational changes will control the outcome level results.

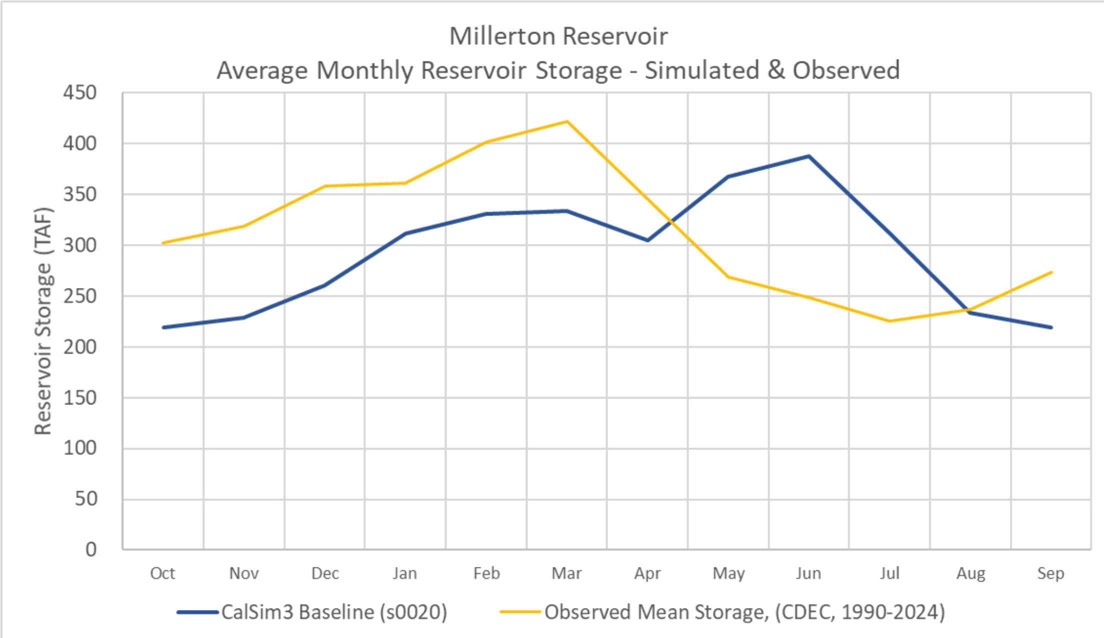

Figure 4. *Comparison of mean monthly storage at Millerton Reservoir. Simulated baseline CalSim3 results (blue line) are shown as a typical example for the model. The observed historical record (orange line), calculated over the period 1990-2024, shows the difference in spring and summer months.*

The reservoirs for which outcome levels are calculated represent a large part of the reservoir capacity in the Central Valley but this selection is not comprehensive. Some reservoirs of significance to local and/or non-project (not part of CVP or SWP) uses and many small reservoirs have not been included in this analysis. This is a potential limitation of the current approach that could be addressed through an expansion of the scope and collection of additional data. However, we view the current set of reservoirs to be a concise yet reasonable reflection of major storage and management actions throughout the Central Valley.

A major assumption underlying this methodology is that certain percentiles of the recent historical data are sufficient for setting reference levels against which to judge simulated outcomes. We selected the 50th percentile (median) observed historical values to define the threshold for high storage as this generally corresponds to relatively high storage volumes for many reservoirs. It may seem somewhat misleading that a median would correspond to a high value, but we note that spring reservoir storage often approaches capacity (either physical or operational) such that the overall distribution is highly skewed, with high storage conditions normally occurring more frequently than average or low storage.

Similarly, we selected the 20th percentile as the level to define the threshold for low storage. This roughly corresponds to the storage level encountered during droughts over recent decades, but is very much an approximation. In some reservoirs a more specifically defined threshold may make more sense - e.g. an April storage historically tied to meeting a certain level of downstream deliveries, ability to support temperature control, or a level at which widespread allocation cuts are made - if it is desired that the outcome level values are more directly calibrated to specific outcomes. Again, we opted for a more uniform approach with the formulation described above so that interpretation could be generalized across all reservoirs a bit more easily.

References

CDEC Data source: https://cdec.water.ca.gov/

Code for calculating reservoir storage outcome levels: https://github.com/canruso/COEQWAL_V3

Appendix

Appendix S-1. Generalized code for continuous outcome level calculation.

In generalized code, the continuous outcome level calculation is implemented as follows. Outcome levels are called 'tiers' in the code below:

Tier1_width = Tier1_cutoff

Tier2_width = Tier2_cutoff – Tier1_cutoff

Tier3_width = Tier3_cutoff – Tier2_cutoff

Tier4_width = D_worst – Tier3_cutoff

```
D = (1 - frac_top) + frac_bot

dist1 = D

dist2 = D - tier1_cutoff

dist3 = D - tier2_cutoff

dist4 = D - tier3_cutoff

IF dist1 <= tier1_width:

ELSE IF dist2 <= tier2_width:

discrete_tier = 2

progress = dist2 / tier2_width

ELSE IF dist3 <= tier3_width:

discrete_tier = 3

progress = dist3 / tier3_width

ELSE:

discrete_tier = 4

progress = dist4 / tier4_width

progress = np.clip(progress, eps, 1 - eps)
```

> tier = discrete_tier + progress

The progress value represents the location of the scenario within its assigned outcome level interval:

- > A progress value near 0 indicates storage performance close to the stronger boundary of the assigned outcome level.
- > A progress value near 1 indicates storage performance close to the weaker boundary of the assigned outcome level.

The final continuous outcome level value is calculated by adding this normalized progress value to the discrete outcome level assignment. This preserves the original Level 1–4 classification while providing additional resolution within each outcome level.

Appendix S-2. Pseudocode for outcome level assignment procedure.

The reservoir storage outcome level assignment procedure can be illustrated with the following pseudocode. In the code, outcome levels are referred to as 'tiers':

FOR each reservoir scenario:

STEP 1: Select April storage conditions

Extract April storage values from the scenario:

April storage by year

STEP 2: Calculate storage performance fractions

Determine:

frac_top =

fraction of April values

where storage >= high_threshold

frac_bot =

fraction of April values

where storage <= low_threshold

frac_mid =

1 − frac_top − frac_bot

STEP 3: Compute storage distance from ideal

The ideal condition is:

frac_top = 1

frac_bot = 0

Compute the combined storage distance:

D = (1 − frac_top) + frac_bot

STEP 4: Define tier boundaries

From the user-specified tier standards:

(top1, bot1)

(top2, bot2)

(top3, bot3)

Compute the cumulative distance cutoffs:

Tier1_cutoff =

(1 − top1) + bot1

Tier2_cutoff =

(1 − top2) + bot2

Tier3_cutoff =

(1 − top3) + bot3

Define the worst possible storage condition:

frac_top = 0

frac_bot = 1

D_worst = (1 − frac_top) + frac_bot

> = (1 − 0) + 1

= 2.0

Compute the width of each tier interval:

Tier1_width =

Tier1_cutoff

Tier2_width =

Tier2_cutoff − Tier1_cutoff

Tier3_width =

Tier3_cutoff − Tier2_cutoff

Tier4_width =

D_worst − Tier3_cutoff

STEP 5: Assign discrete tier

IF D <= Tier1_cutoff:

discrete tier = 1

ELSE IF D <= Tier2_cutoff:

discrete tier = 2

ELSE IF D <= Tier3_cutoff:

discrete tier = 3

ELSE:

discrete tier = 4

STEP 6: Compute within-tier progress

Progress represents the normalized location

within the assigned tier interval.

progress = 0:

strongest performance within the tier

progress = 1:

weakest performance within the tier

IF Tier 1:

progress =

D / Tier1_width

ELSE IF Tier 2:

progress =

(D − Tier1_cutoff)

/

Tier2_width

ELSE IF Tier 3:

progress =

(D − Tier2_cutoff)

/

Tier3_width

ELSE:

progress =

(D − Tier3_cutoff)

/

Tier4_width

Clamp progress to remain strictly within (0,1).

STEP 7: Compute continuous tier

IF continuous tiers are requested:

continuous tier =

discrete tier + progress

ELSE:

continuous tier =

discrete tier

Store:

Scenario

frac_top

frac_mid

frac_bot

storage tier

All code for this implementation is available via GitHub at https://github.com/canruso/COEQWAL_V3, with the core functions available in coeqwalpackage/tier.py.