Drought Simulation Function Documentation

Overview

The `drought_simulation()` function simulates seasonal drought conditions by redistributing precipitation from spring/summer months to fall/winter months. The function can optionally apply monthly precipitation multipliers before performing the drought simulation, allowing for climate change scenarios or other precipitation adjustments.

Function Signature

```r
drought_simulation(
  input_file = "precip.csv",
  output_file = "drought.csv",
  drought_factor = 0.75,
  apply_monthly_multipliers = FALSE,
  monthly_multipliers_file = "MonthlyDeltaShifts.csv"
)
```

Parameters

Parameter	Type	Default	Description
`input_file`	string	`"precip.csv"`	Path to input CSV file containing precipitation data
`output_file`	string	`"drought.csv"`	Path for output CSV file
`drought_factor`	numeric	`0.75`	Fraction of spring/summer precipitation to retain (0-1)
`apply_monthly_multipliers`	logical	`FALSE`	Whether to apply monthly precipitation multipliers
`monthly_multipliers_file`	string	`"MonthlyDeltaShifts.csv"`	Path to monthly multipliers CSV file

Input File Requirements

Primary Precipitation Data

The input file must be a CSV containing:

- **Date column**: Named "date" (case-insensitive). Supported formats:
  - YYYY-MM-DD
  - YYYY/MM/DD

- YYYY.MM.DD
- YYYY MM DD
- YYYY-MM-DD HH:MM:SS
- YYYY/MM/DD HH:MM:SS

- **Precipitation column**: Named "precipitation" (case-insensitive)
- Values must be numeric (can include 0 for no precipitation days)

Monthly Multipliers File (Optional)

When `apply_monthly_multipliers = TRUE`, the function reads a separate CSV file with:

- **Month column**: Integer values 1-12 representing calendar months
- **PPctChange column**: Percentage change values (can be positive or negative)

Example MonthlyDeltaShifts.csv:

```csv
Month,PPctChange,Toffset
1,-10,0
2,-5,0
3,5,0
4,10,0
5,15,0
6,20,0
7,25,0
8,20,0
9,10,0
10,5,0
11,-5,0
12,-10,0
```

Algorithm Description

Step 1: Monthly Multiplier Application (Optional)

If `apply_monthly_multipliers = TRUE`:

1. Read monthly multipliers from the specified file
2. For each day with precipitation > 0:
   - Extract the month from the date
   - Apply multiplier: `shifted_precipitation = original_precipitation × (100 + PPctChange) / 100`
3. Days with zero precipitation remain unchanged

4. Missing months default to 0% change (multiplier = 1.0)

Step 2: Seasonal Classification

- **Spring/Summer**: March through August (months 3-8)
- **Fall/Winter**: September through February (months 9-12, 1-2)
- **Season Year Assignment**:
  - Spring/Summer: Same calendar year
  - Fall/Winter: September-December use current year, January-February use previous year

Step 3: Drought Simulation

1. **Calculate seasonal totals** by season year using (potentially shifted) precipitation
2. **Reduce spring/summer precipitation**: Multiply by `drought_factor`
3. **Calculate redistribution amount**: `p2shift = spring_summer_total × (1 - drought_factor)`
4. **Redistribute to fall/winter**:
   - Only to days with existing precipitation > 0
   - Proportional to each day's share of total fall/winter precipitation
   - `new_precipitation = original_precipitation + (p2shift × day_proportion)`

Output Format

The output CSV contains different columns depending on whether monthly multipliers are applied:

Without Monthly Multipliers

Column	Description
`date`	Original date
`original_precipitation`	Input precipitation values
`drought_precipitation`	Drought-adjusted precipitation
`scaling_factor`	Ratio of drought to original precipitation

With Monthly Multipliers

Column	Description
`date`	Original date
`original_precipitation`	Input precipitation values
`shifted_precipitation`	After applying monthly multipliers
`drought_precipitation`	Final drought-adjusted precipitation
`scaling_factor`	Ratio of drought to shifted precipitation

Mass Balance Conservation

The function preserves total precipitation mass:

- **Without multipliers**: `sum(drought_precipitation) = sum(original_precipitation)`
- **With multipliers**: `sum(drought_precipitation) = sum(shifted_precipitation)`

Precipitation is only redistributed between seasons, not created or destroyed.

Usage Examples

Basic Usage (No Monthly Multipliers)

```r
# Use default parameters
result <- drought_simulation()

# Custom drought factor
result <- drought_simulation(drought_factor = 0.6)  # Retain 60% of spring/summer precip
```

With Monthly Multipliers

```r
# Apply monthly multipliers with defaults
result <- drought_simulation(apply_monthly_multipliers = TRUE)

# Custom files and parameters
result <- drought_simulation(
  input_file = "my_precipitation.csv",
  output_file = "drought_scenario.csv",
  drought_factor = 0.8,
  apply_monthly_multipliers = TRUE,
  monthly_multipliers_file = "climate_change_factors.csv"
)
```

Climate Change Scenario Example

```r
```

```r
# Simulate climate change with increased summer precipitation and drought
result <- drought_simulation(
  input_file = "historical_precip.csv",
  output_file = "future_drought_scenario.csv",
  drought_factor = 0.7,  # More severe spring/summer reduction
  apply_monthly_multipliers = TRUE,
  monthly_multipliers_file = "rcp85_precipitation_changes.csv"
)
```

Error Handling

The function includes comprehensive error checking:

- **File validation**: Checks if input files exist
- **Data validation**: Ensures required columns are present
- **Parameter validation**:
  - `drought_factor` must be between 0 and 1
  - Month values must be 1-12
- **Date parsing**: Attempts multiple date formats
- **Missing data**: Removes rows with invalid dates or precipitation values
- **Missing months**: Automatically adds missing months with 0% change

Dependencies

- **Required**: `lubridate` package for date handling
- **Auto-installation**: Function will install lubridate if not present

Console Output

The function provides detailed progress reporting:

Monthly Multipliers Summary (when applied)

```
Monthly multipliers loaded successfully:
  January (Month 1): -10% change, multiplier = 0.900
  February (Month 2): -5% change, multiplier = 0.950
  March (Month 3): +5% change, multiplier = 1.050
  ...

Monthly multiplier summary:
  Original total precipitation: 1234.5678
  Shifted total precipitation: 1345.6789
  Total change: +9.00% (+111.1111)
```

Monthly precipitation changes:
  January: -12.3456 (-10.00%)
  February: -6.7890 (-5.00%)
  ...

Drought Simulation Summary

```
=== DROUGHT SIMULATION SUMMARY ===
Input file: precip.csv
Output file: drought.csv
Monthly multipliers applied: Yes
Monthly multipliers file: MonthlyDeltaShifts.csv
Drought reduction factor: 0.75
Total records processed: 3653
Date range: 2010-01-01 to 2019-12-31
Years processed: 2010 to 2019

Seasonal breakdown:
Spring/summer days: 1837 (with precip: 892)
Fall/winter days: 1816 (with precip: 734)

Precipitation totals:
Original total precipitation: 1234.568
Shifted total precipitation: 1345.679
Drought-adjusted total precipitation: 1345.679
Mass balance difference: 0.000000

Scaling factor statistics:
Spring/summer scaling factor: 0.750
Fall/winter scaling factors - Min: 1.000 Max: 2.456 Mean: 1.234
```

Technical Notes

Season Year Logic

The function handles the fall/winter season spanning calendar years by assigning January-February to the previous year's fall/winter season. This ensures proper seasonal continuity for precipitation redistribution.

Proportional Redistribution

Fall/winter precipitation enhancement is proportional to existing precipitation patterns, maintaining realistic spatial and temporal distribution while adding the redistributed water.

Zero Precipitation Handling

- Days with zero precipitation remain zero (no artificial precipitation creation)
- Monthly multipliers only affect days with existing precipitation > 0
- Redistribution only goes to fall/winter days that already have precipitation

Precision and Rounding

The function uses high precision for calculations and reports values to appropriate decimal places based on the magnitude of the data.

Return Value

The function returns a data frame containing the output data (same structure as written to the output file) and invisibly returns this for further analysis or chaining operations.

Common Use Cases

1. **Climate Impact Studies**: Apply future precipitation projections with monthly multipliers, then simulate drought effects
2. **Water Resource Planning**: Model seasonal water availability under drought conditions
3. **Agricultural Impact Assessment**: Evaluate crop growing season precipitation changes
4. **Hydrological Modeling**: Generate drought scenario inputs for watershed models
5. **Risk Assessment**: Create extreme event scenarios for infrastructure planning