Streamflow data analysis with *fasstr* :: CHEAT SHEET

Getting Started

[fasstr](#), Flow Analysis Summary Statistics Tool for R, is a package for tidying, summarizing, performing hydrologic analyses, and visualizing daily streamflow data.

Install **fasstr** from CRAN using :

    install.packages("fasstr")

To use the **station_number** function argument, a Water Survey of Canada [HYDAT](#) database must be downloaded using:

    tidyhydat::download_hydat()

Function Usage

fasstr functions can be generally categorized into the following groups:

- **Tidying** - preparing data for analyses; add_* and fill_* functions.
- **Screening** - to look for outliers and missing data; screen_* functions.
- **Calculating summary statistics** - long-term, annual, monthly and daily statistics; calc_* functions.
- **Visualizing summary statistics** - plotting the various statistics; plot_* functions.
- **Computing** analyses - volume frequency analyses and trending; compute_* functions.
- **Writing** data and plots - to save your data and plots; write_* functions.

Getting Data

There are two argument options in most functions to choose a data source:

1. **data** Data frame of daily data with dates (YYYY-MM-DD), flow values, and optional groupings. 'data' is the first argument listed to allow for piping (%>%). Arguments for selecting columns in **data** data frame:
   - **dates** Dates column, default 'Date'.
   - **values** Flow values column, default 'Value'.
   - **groups** Groupings columns (optional), default 'STATION_NUMBER'.

2. **station_number** Extracts daily data from a HYDAT database using a vector of HYDAT station numbers (ex. '08NM116' or c('08NM116', '08FA002')); downloaded HYDAT required.

Example data with default column names:

STATION_NUMBER	Date	Value
08NM116	1987-04-06	6.230
08NM116	1987-04-07	6.440

Function Outputs

All outputs from are one, or lists, of the following:

- All **data tables / data frames** produced as **tibbles**.
- All **plots** are produced as lists of **ggplot2** objects.

Data Tidying

These functions add rows and columns to daily streamflow data frames to prepare for custom analyses.

fill_missing_dates()
Fill dates with missing flow values with NA.

add_date_variables(water_year_start = 1)
Add 'Year', 'Month', 'MonthName', 'WaterYear' 'DayofYear' columns. 'WaterYear' and 'DayofYear' adjust to the selected year start with water_year_start argument.

add_seasons(seasons_length)
Adds column of season identifiers called 'Season' with the length of seasons in months chosen with seasons_ length argument, seasons start in first month of year.

add_rolling_means(roll_days, roll_align)
Add columns of rolling daily flow means (ex. 7-day means).

add_basin_area(basin_area)
Add a basin area column, in square kilometres. See basin_area argument on reverse of cheat sheet.

add_daily_volume()
Add daily volumetric flows, converted from daily mean to cubic metres.

add_daily_yield(basin_area)
Add daily yields, converted from daily mean to millimetres based on upstream basin area.

add_cumulative_volume()
Add daily cumulative volumetric flows on an annual basis, in cubic metres.

add_cumulative_yield()
Add daily cumulative runoff yield flows on an annual basis, in millimetres based on upstream basin area.

Data Screening

These functions calculate and plot statistics to screen data for outliers, gaps, and missing dates.

screen_flow_data()
Calculate annual mean, maximum, minimum, standard deviation, and missing dates.
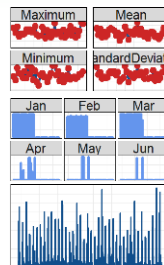
plot_data_screening()
Plot annual mean, maximum, minimum, and standard deviation.

plot_missing_dates()
Plot the number of missing dates for each month and year.

plot_flow_data(plot_by_year = FALSE, one_plot = TRUE)
Plot the daily mean data set.


Basic Summary Statistics

These functions calculate and plot the mean, median, maximum, minimum, and selected percentiles using the 'percentiles' argument. Can select duration of statistics (ex. 7-day) using 'roll_days' and 'roll_align' arguments.

calc_annual_stats()
plot_annual_stats()
Statistics for each year.

calc_daily_stats()
plot_daily_stats(add_year)
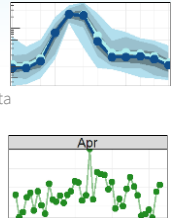Statistics for each day of the year over all years.
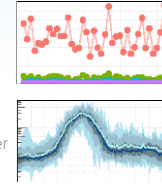
calc_longterm_daily_stats()
plot_longterm_daily_stats()
calc_longterm_monthly_stats()
plot_longterm_monthly_stats()
Statistics for all daily and monthly data for each month over all years.

calc_monthly_stats()
plot_monthly_stats()
Statistics for each month of each year.

Cumulative Statistics

These functions calculate and plot the total flows for years by volume (m³) or by area-based yield (mm) using the 'use_yield' and 'basin_area' area arguments.

calc_annual_cumulative_stats(include_seasons = TRUE**)**
**plot_annual_cumulative_stats(**include_seasons = TRUE**)**
Total annual cumulative discharge for each year (option to include seasonal totals).

calc_monthly_cumulative_stats()
plot_monthly_cumulative_stats(add_year**)**
Cumulative monthly statistics for each month over all years.

calc_daily_cumulative_stats()
plot_daily_cumulative_stats(add_year**)**
Cumulative daily statistics for each day of year over all years
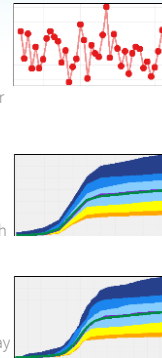
Long-term Statistics

These functions calculate and plot various long-term statistics outside of the basic summary statistics.

plot_flow_duration()
Plot flow durations curves for each month and annually over all years.
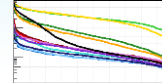

calc_longterm_mean(percent_MAD**)**
Calculate the mean discharge over all years with options to include percentages of the long-term mean.

calc_longterm_percentile(percentiles**)**
Calculate percentile flow values over all years.

calc_flow_percentile(flow_value**)**
Calculate the percentile rank of a specific flow value from flows over all years.

Annual Statistics

These functions calculate and plot various annual statistics beyond the basic summary statistics.

calc_annual_flow_timing(percent_total = c(25,33.3,25,75)**)**
plot_annual_flow_timing(percent_total = c(25,33.3,25,75)**)**
Calculate the day of year when portions of total annual flows have occurred (ex timing of half flows).

calc_annual_lowflows(roll_days = c(1,3,7,30)**)**
plot_annual_lowflows(roll_days = c(1,3,7,30)**)**
Calculate the values and day of occurrence for annual minimum flow values. Multiple 'roll_days' allowed.

calc_annual_outside_normal(normal_percentiles = c(25,75)**)**
plot_annual_outside_normal(normal_percentiles = c(25,75)**)**
Calculate the number of days per year that occur above or below "normal", "normal" period based on values provided.


plot_annual_means()
Plot annual mean flows with the x-axis centred on the long-term mean

calc_annual_peaks()
Calculate the values and day of occurrence for annual n-day (using 'roll_days') minimum and maximum flow values.

calc_all_annual_stats(annual_percentiles = c(10,90), monthly_percentiles = c(10,20), stats_days = 1, lowflow_days = c(1,3,7,30), timing_percent = c(25,33.3,50,75), normal_percentiles = c(25,75)**)**
Calculate all statistics from all calc_annual_* and calc_monthly_stats() functions.

BRITISH COLUMBIA
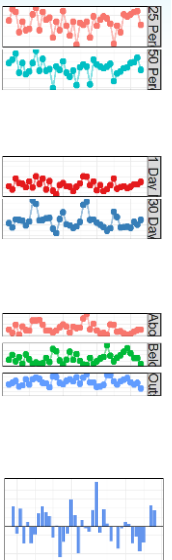
Arguments and Options

These arguments are used to customize many of the functions. Not all are listed; see function documentation for more specific argument information.

Date Filtering and Options

ignore_missing Logical value indicating whether dates with missing values should be included in the analysis. If TRUE then a statistic will be calculated regardless of missing dates. If FALSE then only statistics with no missing dates will be returned.

water_year_start Numeric value indicating the starting month (1 through 12) of years to filter/group data instead of calendar years, designated by calendar year in which year ends; default 1.

start_years and **end_years** Numeric values of the first and last year to consider for analysis. Leave blank to include all years of data provided.

exclude_years Numeric vector of years to exclude from analysis; ex. c(1991:1993, 1995). Leave blank to include all years of data provided.

complete_years Logical value indicating whether to only include years with complete data in analysis. Only in selected analyses; default FALSE.

months Numeric vector of months to include in analysis; default 1:12.

Data Analysis Options

roll_days Numeric value (or values for some functions) of the number of days to apply a rolling mean; default 1.

roll_align Character string identifying the direction of the rolling mean from the specified date, either by the first ('left'), last ('right'), or middle ('center') day of the rolling n-day group of observations; default 'right'.

use_yield Logical value indicating to use area-based yield, in mm, instead of volumetric for cumulative analysis functions; default FALSE. Requires basin_area.

basin_area Drainage basin area, in square km, to use when use_yield = TRUE. Three options: 1) leave blank if column of HYDAT station numbers; 2) single numeric value to apply to all observations.; 3) list each basin area for each station c("08NM116" = 795, "08NM242" = 10) to supply an area or override the HYDAT supplied area.

percentiles Numeric vector of percentiles to calculate, ex. c(5,25,75,95). Set to NA if none required.

Tibble Options

transpose Logical value indicating whether to transpose rows and columns of results; default FALSE.

Plotting Options

log_discharge Logical value to indicate plotting the discharge axis on a logarithmic scale; default FALSE.

include_title Logical value to indicate adding the group/station number to the plot, if provided.

add_year Numeric value indicating a year of daily flows to add to the daily and long-term statistics plot.

Writing Functions

These functions help save the outputted objects (tibbles and lists of plots) from the **fasstr** functions.

· ·

write_flow_data()
Write a streamflow dataset as a .xlsx, .xls, or .csv file. Can extract and save HYDAT data with this function.

write_results(digits = 10)
Write a data frame as a .xlsx, .xls, or .csv file. Can save a data frame and round digits of all numeric columns.

write_plots(plots, folder_name, plot_filetype, combined_pdf)
Write plots from a list object into a directory or PDF document. By default will save all plots in a folder. To create a PDF of all plots, set combined_pdf = TRUE.

write_objects_list(list, folder_name, table_filetype, plot_filetype)
Write all tables and plots contained in a list object into a folder. Saves only data frames and ggplot2 objects.

Annual Trending Analysis

This function computes and plots prewhitened, non-parametric annual trends on streamflow data.

· ·

This function calculates prewhitened, non-parametric annual trends using the 'zyp' package. It calculates various annual metrics using the calc_all_annual_stats() function and then calculates and plots the trends. See the zyp package, function documentation, and the trending vignette for more information on the analysis.

Function

compute_annual_trends()
Calculate prewhitened nonlinear annual trends on streamflow data.

Arguments

zyp_method Prewhitening method, either 'yuepilon' or 'zhang'. See zyp methodology for more information.

include_plots Logical value indicating if annual trending plots should be included. Default TRUE.

zyp_alpha Numeric value of the significance level (ex. 0.05) of when to plot a trend line. Leave blank for no line.

Outputs

$Annual_Trends_Data A tibble of annual data from the `calc_all_annual_stats()` function used for trending

$Annual_Trends_Results A tibble of annual trending results, including significance, confidence intervals, trend values, etc.

$Annual_* A ggplot2 object for each annual statistic trended, with the slope plotted if significance is greater than 'zyp_alpha' provided.
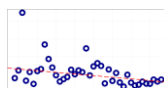
Volume Frequency Analyses

These functions compute and plot volume frequency analyses on annual low or high streamflow data.

· ·

These functions perform volume frequency analyses by fitting annual minimums or maximums to Log-Pearson Type III or Weibull probability distributions. These functions plot probabilities of data using chosen plotting methods and calculates frequency quantiles (ex. 7Q10) based on fitting data to selected distributions and fitting methods. See function documentation for more information.

Functions

compute_annual_frequencies()
Annual frequency analysis from daily streamflow data; calculates minimums or maximums of selected roll_days.

compute_frequency_quantile()
Annual frequency analysis from daily streamflow data; calculates minimums or maximums of selected roll_days and return_period. Quantile value is returned.

compute_hydat_peak_frequencies()
Annual frequency analysis from instantaneous peak data (minimum or maximum) for stations from HYDAT. Data selected using station_number argument.

compute_frequency_analysis()
Conduct a frequency analysis with custom data.

Arguments

use_max Rank data from high to low rather than low to high (for peak analyses); default FALSE.

use_log Log-transform event data before analysis; default FALSE.

prob_plot_positions Plotting positions used to plot the probabilities; 'weibull' (default), 'hazen', or 'median'.

prob_scale_points Probabilities to be plotted on the x-axis; default c(.9999, .999, .99, .9, .5, .2, .1, .02, .01, .001, .0001).

fit_distr Distribution used to fit the data; one of Log-Pearson Type III, 'PIII' (default), or Weibull, 'weibull.'.

fit_dist_method Method used to fit the data to the distribution; one of method of moments , 'MOM' (default), or maximum likelihood estimation, 'MLE'.

fit_quantiles Quantiles to be estimated from the fitted distributions (event probabilities); default c(.975, .99, .98, .95, .90, .80, .50, .20, .10, .05, .01).

plot_curve Plot the computed curve on the plot; default TRUE.

Outputs

$Freq_Analysis_Data Tibble of computed or extracted data used in analysis.

$Freq_Plot_Data Tibble of plotting coordinates used in the frequency plot.

$Freq_Plot ggplot2 object of the frequency plot with return periods and probabilities.

$Freq_Fitting fitdistplus::fitdist objects of fitted distributions.

$Freq_Fitted_Quantiles Tibble of fitted quantiles with probabilities and return periods.
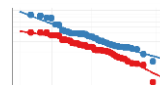
Computing Full Analyses

These functions calculate a suite of data and plots from many of the **fasstr** functions.

· ·

These functions calculate many of the data and plot analyses from the fasstr functions, producing tables and plots organized by analysis types. See the function documentation for more information.

Functions

compute_full_analysis()
Computes a suite of analyses from fasstr functions and produces assorted tables and plots organized in lists grouped by time period and analysis type.

write_full_analysis()
Writes the compute_full_analysis() objects into an Excel workbook and accompanying plot files.

Arguments

analyses Numeric vector of the analyses to include; default is all (1:7). Include those analyses with which statistics are desired: 1: Screening, 2: Long-term, 3: Annual, 4: Monthly, 5: Daily, 6: Trending, 7: Low-flow Frequencies.

Writing Arguments

file_name Name of Excel workbook, and plots folder if necessary, to save analysis results.

Outputs

$Screening List of table and plot objects to review and screen data.

$Longterm List of table and plot objects from long-term statistics, including summary statistics and flow duration.

$Annual List of table and plot objects from annual statistics, including summary and cumulative statistics, and other annual metrics.

$Monthly List of table and plot objects from monthly statistics, including summary and cumulative statistics.

$Daily List of table and plot objects from daily statistics, including summary and cumulative statistics.

$Trending List of table and plot objects from an annual trending analysis.

$Lowflow_Frequencies List of table and plot objects from a low-flow frequency analysis.

Writing Outputs

Excel '.xlsx' workbook containing tables and plots from selected analyses, and a folder of plots if Daily and/or Trending analyses are computed.

Further Help

There are several vignettes that provide more information and examples of the many fasstr functions:

- *Users Guide*
- *Trending Analysis Guide*
- *Frequency Analysis Guide*
- *Full Analysis Guide*
- *fasstr Internal Workflows*

See https://bcgov.github.io/fasstr/ or view them in R using **browseVignettes("fasstr")**