

DATA MANAGEMENT AND ANALYSIS

RStudio, GitHub, QC/QA, GIS, Statistics, and Data Workflows

Environmental Analysis Program
Teaching and Research Laboratories, Seeley G Mudd
Pomona College

Marc Los Huertos

Environmental Analysis Program Coordinator



Pomona College
Claremont, California

DATA MANAGEMENT AND ANALYSIS

Data Management and Analysis

RStudio, GitHub, QC/QA, GIS, statistics, and instrument data workflows

EA Program – Biogeochemistry Lab, Pomona College

Compiled: June 26, 2026

SOP Number	Title
SOP-L801	RStudio Projects and GitHub
SOP-L802	Quality Control and Quality Assurance
SOP-L803	Data Entry and Database Management
SOP-L804	GIS and Spatial Data Management
SOP-L805	Drone Image Processing (Photogrammetry)
SOP-L806	Statistical Analysis Templates (R)
SOP-L807	Instrument Data Export and File Management



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L801

RStudio Projects and GitHub

Standard Operating Procedure

SOP Number	SOP-L801
Revision	1.07
Author(s)	Marc Los Huertos, Isaac Medina
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.
Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Acknowledgements	2
4 Definitions	2
5 Safety and Hazards	3
6 Accessing RStudio	3
6.1 Pomona College RStudio Server	3
6.2 Local Installation	3
7 Setting Up GitHub	3
7.1 Create a GitHub Account	3
7.2 Configure SSH Keys	3
8 Working with Repositories	4
8.1 Clone an Existing Repository	4
8.2 The Pull-Edit-Commit-Push Workflow	4
8.3 Branching Workflow	4
9 Writing Quarto Documents	5
9.1 Rmd vs. Qmd	5
10 Recommended Resources	5
11 Advanced Topics	5
11.1 Protecting the Main Branch	5
11.2 GitHub Issues for Task Tracking	5
11.3 .gitignore	6
12 References	6
13 Training Record	6

1. Purpose

This SOP provides instructions for setting up RStudio, connecting it to GitHub, and managing collaborative research projects using version control. R (R Core Team, 2024), RStudio (Posit PBC, 2024), and GitHub (GitHub Inc., 2024) together enable collaborative, transparent, reproducible, and accessible data analysis. For a comprehensive practical guide, see (Bryan, 2023).

2. Scope

- **Applies to:** All EA Program researchers using R for data analysis and collaborative projects
- **Does not apply to:** Local-only R scripts with no version control; Python or other language workflows
- **Location:** RStudio Server at Pomona College; local installations

3. Acknowledgements

Substantial contributions: Aparna Chintapalli and Isaac Medina. We acknowledge all students who have followed these instructions and suggested improvements.

4. Definitions

- **R:** A free, open-source programming environment for statistical computing, maintained by a global community of statisticians and programmers (R Core Team, 2024).
- **RStudio:** A user interface for R that provides tools to write code, view results, manage files, and integrate with version control. Available as a desktop application or via the Pomona College server (Posit PBC, 2024).
- **GitHub:** A web-based hosting service for Git repositories that enables collaborative code and document management (GitHub Inc., 2024).
- **Repository (repo):** A GitHub project folder that contains all files, history, and collaboration settings.
- **Version control:** A system that tracks changes to files over time, allowing multiple people to collaborate while maintaining a complete history of who changed what and when.
- **SSH key:** A cryptographic key pair used to establish a secure connection between RStudio and GitHub without entering a password each time.
- **Clone:** Creating a local copy of a GitHub repository on your computer or server.
- **Branch:** A parallel line of development that lets you work on a feature or fix without affecting the main codebase until you are ready to merge.
- **Pull:** Downloading the latest changes from GitHub to your local copy.
- **Commit:** Saving a snapshot of your changes with a descriptive message.
- **Push:** Uploading committed changes from your local copy to GitHub.
- **Pull request (PR):** A request to merge changes from one branch into another; the standard review mechanism for collaborative repositories.

5. Safety and Hazards

This is a computational procedure with no physical hazards.

6. Accessing RStudio

6.1. Pomona College RStudio Server

The recommended method for EA Program work:

1. Open a browser and go to `rstudio.pomona.edu`
2. Log in with your Pomona College credentials
3. Your files are stored on the server (no local installation needed)

6.2. Local Installation

Install in this order if working offline:

1. R: cran.r-project.org
2. RStudio Desktop: posit.co/download/rstudio-desktop
3. Git: git-scm.com

7. Setting Up GitHub

7.1. Create a GitHub Account

1. Go to github.com and create a free account
2. Use your Pomona email address
3. Choose a professional username (e.g., `firstlast` or `lastname`)

7.2. Configure SSH Keys

SSH keys let RStudio communicate with GitHub securely. Set them up once per machine (or server account).

1. In RStudio, open the Terminal tab
2. Generate a key pair:

```
ssh-keygen -t ed25519 -C "your_pomona_email@pomona.edu"
```

3. Accept the default file location; leave the passphrase blank for server use
4. Print the public key:

```
cat ~/.ssh/id_ed25519.pub
```

5. Copy the output
6. On GitHub: Settings > SSH and GPG keys > New SSH key > paste and save
7. Test the connection:

```
ssh -T git@github.com
```

A message beginning “Hi username!” confirms success.

8. Working with Repositories

8.1. Clone an Existing Repository

1. On GitHub, navigate to the repository and click **Code** > **SSH**
2. Copy the SSH URL (`git@github.com:org/repo.git`)
3. In RStudio: File > New Project > Version Control > Git
4. Paste the URL; choose a local directory; click **Create Project**

8.2. The Pull-Edit-Commit-Push Workflow

Always pull before editing to avoid merge conflicts:

1. **Pull:** Click the blue down-arrow in the Git pane (or `git pull`)
2. **Edit:** Make your changes in R scripts or `.qmd` files
3. **Stage:** Check boxes next to changed files in the Git pane
4. **Commit:** Click Commit; write a clear message (e.g., “Add soil texture analysis for sites 4-6”)
5. **Pull again:** Pull once more before pushing to catch any concurrent changes
6. **Push:** Click the green up-arrow to push to GitHub

8.3. Branching Workflow

For experimental analyses or features that are not yet ready for the main branch:

```
git checkout -b feature/my-analysis # create and switch to new branch
# ... make changes and commits ...
git push -u origin feature/my-analysis
# Open a Pull Request on GitHub when ready to merge
```

9. Writing Quarto Documents

9.1. Rmd vs. Qmd

Feature	Rmd	Qmd
Maintained by	RStudio/Posit	Quarto project (open source)
Requires R	Yes	No (also supports Python, Julia)
Chunk options	In header: <code>{r, echo=TRUE}</code>	On lines: <code># echo: true</code>
SOP system uses	Legacy	Preferred for new documents

For new EA Program documents, use `.qmd`. Existing `.Rmd` files will continue to work.

Common Chunk Options

- `#| echo: true` shows the code in the output (use for teaching)
- `#| echo: false` hides the code (use for reports)
- `#| eval: false` shows the code but does not run it
- `#| fig-cap: "Caption text"` adds a figure caption
- `#| warning: false` suppresses warning messages

10. Recommended Resources

- [RStudio YouTube Introduction](#)
- [GitHub YouTube Introduction](#)
- [Understanding the GitHub Flow](#)
- Happy Git and GitHub for the user: (Bryan, 2023) – the most comprehensive practical guide
- [Quarto Documentation](#)

11. Advanced Topics

11.1. Protecting the Main Branch

For repositories with multiple active contributors, enable branch protection rules on GitHub:

1. Go to Settings > Branches > Add rule
2. Enter `main` as the branch name pattern
3. Check **Require a pull request before merging**
4. Optionally check **Require approvals** (set to 1)

11.2. GitHub Issues for Task Tracking

Create an issue for each task, assign it to a team member, and close it when the work is merged. This creates a record of who did what and when.

11.3. .gitignore

Add file patterns that should never be committed:

```
.Rhistory  
.RData  
.Rproj.user/  
docs/  
_freeze/  
*.pdf
```

12. References

Bryan, J. (2023). *Happy Git and GitHub for the useR*. <https://happygitwithr.com/>
GitHub Inc. (2024). *GitHub Docs*. <https://docs.github.com/>
Posit PBC. (2024). *RStudio IDE*. <https://posit.co/products/open-source/rstudio/>
R Core Team. (2024). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. <https://www.R-project.org/>

13. Training Record

Before using R/RStudio/GitHub independently for EA Program research, personnel must:

1. Read this SOP in full
2. Create a GitHub account and configure SSH keys (Sections 11.1–11.3)
3. Successfully clone, pull, edit, commit, and push to the **beginnersluck** practice repository (Sections 12–13)
4. Create and render at least one **.qmd** file (Section 17)
5. Demonstrate the Pull-Edit-Commit-Push workflow under supervision
6. Be signed off by the PI or project lead



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L802

Quality Control and Quality Assurance

Standard Operating Procedure

SOP Number	SOP-L802
Revision	1.1
Author(s)	Marc Los Huertos
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.
Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Responsibilities	2
4 Definitions	2
5 Safety and Hazards	3
6 General QC Requirements	4
6.1 Every Analytical Batch	4
6.2 Additional Elements (as applicable)	4
7 Acceptance Criteria	4
8 Detection Limit Determination	5
8.1 Method Detection Limit (MDL)	5
8.2 Limit of Detection and Limit of Quantitation	5
9 Control Charts	6
10 Corrective Actions	6
11 Metadata Documentation and Reproducibility	6
11.1 Laboratory Notebook Requirements	6
11.2 Electronic Data Management	7
11.3 R-Based QC Tracking System	7
12 QC Requirements by Method Category	8
13 Waste Disposal	8
14 References	8
15 Related SOPs	9
16 Training Record	9

1. Purpose

This SOP establishes the overarching quality assurance (QA) and quality control (QC) framework for all analytical work in the EA Program laboratory. It defines the minimum QC requirements that apply across methods, describes how to calculate and interpret QC metrics, specifies detection limit procedures, and establishes the data documentation and reproducibility practices required for every analysis. Individual analytical SOPs reference this document for general QC principles and add method-specific requirements (APHA et al., 2012; U.S. Environmental Protection Agency, 2007).

Quality assurance is the system of management actions that ensures the analytical process produces results of known and documented quality. Quality control is the set of technical activities (blanks, standards, duplicates, spikes) embedded in each analytical batch to measure and control that quality in real time (Taylor, 1987).

2. Scope

- **Applies to:** All quantitative chemical, physical, and biological analyses performed in the EA Program laboratory, including those described in SOPs L002 through L999 and analyses of field-collected samples (SOPs F001 through F999)
- **Does not apply to:** Qualitative screening tests (e.g., visual soil texture by feel, [SOP-F401](#)) unless those tests include a quantitative component
- **Location:** EA Program laboratory, SGM 133

3. Responsibilities

Role	Responsibility
Lab PI (QA Manager)	Approve this SOP; review QC data quarterly; authorize corrective actions
Lab Manager	Maintain instrument logs, reagent inventory, and QC records; train new analysts
Analyst	Implement all QC requirements for each batch; flag QC failures; document corrective actions
Data Reviewer	Verify QC compliance before data are reported or published

4. Definitions

- **Accuracy:** The closeness of a measured value to the true or accepted reference value; assessed using certified reference materials (CRMs) and laboratory-fortified blanks (LFBs) (Taylor, 1987).
- **Calibration blank:** A solution containing only the analytical solvent or reagent matrix (no analyte); used to establish the instrument baseline (zero) for each calibration sequence.

- **Certified reference material (CRM):** A material with certified analyte concentrations traceable to a national or international metrology institute (NIST, USGS, IRMM); used to verify analytical accuracy across instruments and over time (APHA et al., 2012).
- **Control chart (Shewhart chart):** A time-series plot of QC sample results with control limits (typically ± 2 warning limits and ± 3 action limits); used to distinguish random analytical variation from systematic drift or instrument malfunction.
- **Continuing calibration verification (CVS):** A mid-range calibration standard analyzed at regular intervals (typically every 10 samples) during an analytical run; must recover within $\pm 10\%$ of the stated value; failure requires re-calibration and re-analysis of affected samples (APHA et al., 2012).
- **Detection limit – Method Detection Limit (MDL):** The minimum concentration of an analyte that can be detected with 99% confidence under stated method conditions; determined from at least 7 replicate analyses of a low-level spike following 40 CFR Part 136 Appendix B (U.S. Environmental Protection Agency, 2016).
- **Detection limit – Limit of Blank (LOB):** The highest signal expected from a blank sample; $LOB = \text{mean}_{\text{blank}} + 1.645 \times SD_{\text{blank}}$ (Armbruster & Pry, 2008).
- **Detection limit – Limit of Detection (LOD):** The lowest analyte concentration that can be reliably distinguished from the LOB; $LOD = LOB + 1.645 \times SD_{\text{low-concentration}}$; represents 95% detection probability (Armbruster & Pry, 2008).
- **Detection limit – Limit of Quantitation (LOQ):** The lowest concentration at which the analyte can be measured with acceptable precision and accuracy, typically defined as $10 \times SD$ at the LOD spike concentration; also known as the practical quantitation limit (PQL) (Armbruster & Pry, 2008).
- **Field blank:** A sample of reagent-grade water processed through the entire field collection and preservation procedure; detects contamination introduced during sampling and transport.
- **Laboratory-fortified blank (LFB):** A reagent water blank spiked with a known concentration of target analytes and processed through the full analytical procedure; monitors ongoing accuracy and extraction efficiency.
- **Laboratory-fortified matrix (LFM):** A field sample spiked with a known concentration of target analytes; detects matrix-specific suppression or enhancement of the analytical signal.
- **Method blank:** A reagent water sample carried through the full analytical procedure alongside field samples; detects contamination introduced by reagents, equipment, or laboratory environment (APHA et al., 2012).
- **Precision:** The reproducibility of repeated measurements of the same sample; expressed as relative percent difference (RPD) for duplicates or relative standard deviation (RSD) for replicate analyses (Taylor, 1987).
- **Relative percent difference (RPD):** $|A - B| / ((A + B) / 2) \times 100$; the standard measure of duplicate precision; typical acceptance criteria are RPD 10–20% depending on method and matrix.
- **Spike recovery (%):** $(\text{Measured spiked concentration} - \text{Unspiked concentration}) / \text{Added concentration} \times 100$; typical acceptance range is 80–120% for most environmental matrices (APHA et al., 2012).

5. Safety and Hazards

QC samples contain the same analytes as field samples and are subject to the same hazard considerations as the parent analytical method. Consult the relevant analytical SOP for safety information.

6. General QC Requirements

6.1. Every Analytical Batch

Every analytical batch must include at minimum (APHA et al., 2012; U.S. Environmental Protection Agency, 2007):

QC Element	Minimum Frequency	Purpose
Method blank	1 per batch	Detect laboratory contamination
Calibration blank	Start of each calibration	Instrument baseline
CVS (check standard)	Every 10 samples	Detect calibration drift
Duplicate	10% of samples	Assess precision
LFB	1 per batch	Assess accuracy

6.2. Additional Elements (as applicable)

QC Element	When Required
Field blank	Each sampling event
Matrix spike (LFM)	5% of samples or 1 per batch
CRM	1 per batch (where certified material is available)
Equipment blank	When non-dedicated sampling equipment is used

7. Acceptance Criteria

QC Element	Typical Acceptance Criteria	Action if Failed
Method blank	< MDL (analyte absent)	Investigate contamination; do not report affected samples
CVS	$\pm 10\%$ of stated value	Re-calibrate; re-analyze affected samples
Duplicate RPD	10–20% (method-specific)	Flag results; investigate and re-analyze if warranted
LFB recovery	80–120%	Flag results; check reagents and instrument

QC Element	Typical Acceptance Criteria	Action if Failed
LFM recovery	70–130%	Suspect matrix interference; consult PI
CRM recovery	Within certified uncertainty interval	Recalibrate; re-analyze

Acceptance criteria in individual analytical SOPs take precedence over these general values when they are stricter or when regulatory requirements apply.

8. Detection Limit Determination

8.1. Method Detection Limit (MDL)

The MDL is determined following 40 CFR Part 136 Appendix B (U.S. Environmental Protection Agency, 2016):

1. Select a spike concentration at $1\text{--}5\times$ the estimated MDL.
2. Prepare and analyze at least 7 replicates of the spiked solution through the full analytical method.
3. Calculate the standard deviation (s) of the replicates.
4. Multiply by the one-tailed t -value at 99% confidence ($t_{n-1,0.99}$; for $n = 7$: $t = 3.143$):

$$\text{MDL} = t_{(n-1, 0.99)} \times s$$

5. If the spike concentration is outside the range of $0.5\text{--}20\times$ the computed MDL, repeat with an adjusted spike concentration (U.S. Environmental Protection Agency, 2016).

8.2. Limit of Detection and Limit of Quantitation

The LOB, LOD, and LOQ hierarchy follows Armbruster & Pry (2008):

$$\text{LOB} = \mu_{\text{blank}} + 1.645 \sigma_{\text{blank}}$$

$$\text{LOD} = \text{LOB} + 1.645 \sigma_{\text{low}}$$

$$\text{LOQ} = 10 \sigma_{\text{low}}$$

where σ_{low} is the standard deviation of replicate analyses at the low-level spike concentration. For right-censored environmental data, apply the non-parametric methods described in Helsel (2012).

9. Control Charts

Maintain control charts for each analyte/method combination to track long-term analytical performance:

1. Plot CRM, LFB, or CVS recovery for each batch on a time-series chart.
2. Calculate and draw lines for:
 - **Center line:** Mean recovery over the last 20+ batches
 - **Warning limits:** ± 2 standard deviations from the mean
 - **Action limits:** ± 3 standard deviations from the mean
3. A result between warning and action limits is a warning; investigate but do not necessarily re-analyze.
4. A result beyond the action limits requires corrective action and re-analysis.
5. Systematic trends (7+ consecutive results on one side of the center line, or a steady drift) require investigation even if individual points are within limits (Taylor, 1987).

10. Corrective Actions

When a QC failure occurs:

1. Stop the analytical run; do not report affected samples.
2. Identify the probable cause (contaminated blank, degraded standard, instrument malfunction, matrix interference).
3. Correct the problem (replace reagents, recalibrate, perform maintenance).
4. Re-run the affected samples with fresh QC elements.
5. Verify that the repeated run passes all QC criteria.
6. Document the failure, root cause, corrective action, and re-analysis results in the QC log and lab notebook.

11. Metadata Documentation and Reproducibility

11.1. Laboratory Notebook Requirements

Every analytical session must be documented with at minimum:

- Date, analyst name, SOP number and revision
- Instrument ID and calibration status
- Reagent lot numbers and preparation dates
- Sample IDs, matrix type, and collection dates
- Calibration data (standards, equation, r^2)
- QC results (blanks, duplicates, LFBs, spikes, CRMs) with pass/fail
- Raw data file names and storage location
- Any deviations from the SOP, with justification
- Corrective actions taken for QC failures

11.2. Electronic Data Management

1. Store all raw instrument data files on the lab shared drive in a folder structure organized by project, year, and instrument.
2. Never modify raw data files. All processing must be done programmatically in R with version-controlled scripts.
3. Commit all analysis scripts to the lab GitHub repository per [SOP-L801](#) (Gandrud, 2015; Xie, 2015).

11.3. R-Based QC Tracking System

The following R framework is the EA Program standard for QC documentation and reproducibility (Gandrud, 2015; Xie, 2015):

```
library(tidyverse)
library(lubridate)
library(knitr)

session_meta <- list(
  analyst    = "Your Name",
  date       = Sys.Date(),
  sop        = "SOP-L802 v1.1",
  method_sop = "SOP-LXX vX.X",
  instrument = "Instrument ID",
  project    = "Project Name"
)

cat(sprintf("QC session: %s | Analyst: %s | Instrument: %s\n",
            session_meta$date, session_meta$analyst,
            session_meta$instrument))
```

```
calc_mdl <- function(reps, alpha = 0.01) {
  n <- length(reps); s <- sd(reps)
  t_val <- qt(1 - alpha, df = n - 1)
  list(n = n, sd = s, t_value = round(t_val, 3),
       MDL = round(t_val * s, 4), LOQ = round(10 * s, 4))
}

calc_rpd <- function(a, b) abs(a - b) / ((a + b) / 2) * 100

calc_recovery <- function(measured, expected) (measured / expected) * 100
```

Each analytical R script should:

1. Read the raw data file
2. Run all QC checks programmatically

3. Produce a formatted QC summary table
4. Generate calibration plots and control charts
5. Export a clean, QC-verified dataset as CSV
6. Render to HTML or PDF for archiving

Store templates in the project GitHub repository under `analysis/templates/` (Gandrud, 2015; Xie, 2015).

12. QC Requirements by Method Category

QC Element	Water Chemistry	Soil Extraction	Digestion (Metals)	Biological (Colilert)	Instrumental (TOC, ICP)
Calibration blank	Each cal	Each cal	Each cal	N/A	Each cal
Method blank	1/batch	1/batch	1/batch	1/batch	1/batch
Field blank	1/event	N/A	N/A	N/A	N/A
Duplicate	10%	10%	10%	10%	10%
LFB	1/batch	1/batch	1/batch	N/A	1/batch
Matrix spike	5% (or 1/batch)	5%	5%	N/A	5%
CRM	1/batch (if avail.)	1/batch	1/batch	Positive control	1/batch
CVS	Every 10 samples	Every 10	Every 10	N/A	Every 10

13. Waste Disposal

QC samples generate the same waste streams as routine samples. Dispose of QC waste (blanks, standards, spikes) according to the parent method's waste disposal procedure.

14. References

- APHA et al. (2012)
- Taylor (1987)
- U.S. Environmental Protection Agency (2007)
- U.S. Environmental Protection Agency (2016)
- Armbruster & Pry (2008)
- Helsel (2012)
- Gandrud (2015)
- Xie (2015)

- APHA, AWWA, & WEF. (2012). *Standard methods for the examination of water and wastewater* (22nd ed., p. 1360). American Public Health Association.
- Armbruster, D. A., & Pry, T. (2008). Limit of blank, limit of detection and limit of quantitation. *Clinical Biochemistry Reviews*, 29(Suppl 1), S49–S52.
- Gandrud, C. (2015). *Reproducible research with R and RStudio* (2nd ed.). CRC Press.
- Helsel, D. R. (2012). *Statistics for censored environmental data using Minitab and R* (2nd ed.). John Wiley & Sons. <https://doi.org/10.1002/9781118162729>
- Taylor, J. K. (1987). *Quality assurance of chemical measurements*. Lewis Publishers.
- U.S. Environmental Protection Agency. (2007). *Guidance for quality assurance project plans* (EPA/240/R-02/009). Office of Environmental Information.
- U.S. Environmental Protection Agency. (2016). *Definition and procedure for the determination of the method detection limit, revision 2* (40 CFR Part 136, Appendix B). U.S. Environmental Protection Agency. <https://www.epa.gov/cwa-methods/mdl-determination-procedure>
- Xie, Y. (2015). *Dynamic documents with R and knitr* (2nd ed.). CRC Press.

15. Related SOPs

SOP	Title	Relationship
L801	RStudio and GitHub	Version control for analysis scripts
L803	Data Entry and Database	Data recording standards
F300	Surface Water Sampling	Field blank requirements
F008	Sample Labeling and COC	Chain of custody

16. Training Record

Before applying QC frameworks independently, personnel must:

1. Read this SOP in full
2. Demonstrate correct MDL calculation using the `calc_mdsl()` function on a test dataset
3. Prepare a complete QC summary table for one analytical batch under supervision
4. Set up and interpret a control chart for one analyte/method combination
5. Be signed off by the PI or lab manager



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L803

Data Entry and Database Management

Standard Operating Procedure

SOP Number	SOP-L803
Revision	0.1
Author(s)	Marc Los Huertos
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.
Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Responsibilities	2
4 Definitions	2
5 Safety and Hazards	3
6 Equipment and Materials	3
6.1 Software	3
6.2 File Storage	3
7 Procedure	3
7.1 File Naming Conventions	3
7.2 Directory Structure	4
7.3 Data Entry from Lab Notebooks	4
7.4 Variable Naming Conventions	4
7.5 Data Types and Formatting	5
7.6 Data Validation in R	5
7.7 Data Dictionaries (Metadata)	6
7.8 Version Control with Git	6
7.9 Database Management for Multi-Year Projects	6
8 Quality Control and Acceptance Criteria	7
9 Waste Disposal	7
10 References	7
11 Training Record	7
12 Appendix A: CSV Template	8
13 Appendix B: Recommended R Packages for Data Management	8

1. Purpose

This SOP describes standardized procedures for entering, organizing, validating, and managing research data in the EA Program. It covers the full data lifecycle from initial recording through archival storage, ensuring that all datasets are consistent, traceable, and reproducible (Broman & Woo, 2018; Wickham, 2014).

2. Scope

- **Applies to:** All tabular data generated in the EA Program laboratory and field campaigns, including instrument outputs, field observations, and manually recorded measurements
- **Does not apply to:** Spatial/GIS data (see [SOP-L804](#)), bioinformatics sequence data (see [SOP-L705](#)), or instrument-specific export procedures (see [SOP-L807](#))
- **Location:** EA Program laboratory (SGM 133) and RStudio Server

3. Responsibilities

Role	Responsibility
Lab PI	Approve data management plan; ensure compliance with funder requirements
Lab Manager	Maintain database templates; perform quarterly backups
Analyst / Student	Enter data following this SOP; run validation checks; document metadata
Data Reviewer	Verify data entries against lab notebooks before analysis

4. Definitions

- **Data dictionary:** A companion metadata file that describes each variable in a dataset, including the column name, description, units, data type, expected range, and source SOP; required for every EA Program dataset.
- **ISO 8601:** An international standard for representing dates and times as YYYY-MM-DD; the required date format for all EA Program data files because it sorts chronologically and is unambiguous across locales.
- **Missing value:** A data cell for which no measurement was made or the measurement was invalid; represented as blank or NA in EA Program datasets; never coded as 0 (which is a valid measurement).
- **Primary key:** A unique identifier for each row in a data table (e.g., `sample_id`); enables unambiguous linking between related tables and detection of duplicate entries.
- **Tidy data:** A data structure in which each variable has its own column, each observation has its own row, and each type of observational unit forms its own table (Wickham, 2014); the standard format for EA Program analytical datasets.

- **Version control:** A system that records changes to files over time so that specific versions can be recalled; GitHub/Git is the EA Program version control platform for all data, scripts, and SOPs.

5. Safety and Hazards

Data entry is a computational activity with no physical hazards. Ergonomic best practices (regular breaks, proper screen height) are encouraged for extended data entry sessions.

6. Equipment and Materials

6.1. Software

Item	Access	Notes
RStudio Server	rstudio.pomona.edu	Primary data processing tool
R (version 4.x)	Pre-installed on server	Required for validation scripts
Git / GitHub	github.com/marclos	Version control for all data files
Microsoft Excel / Google Sheets	Campus license	Acceptable for initial entry only
Text editor (VS Code, nano)	Various	For editing CSV files directly

6.2. File Storage

Location	Purpose	Backup Frequency
GitHub repository	Active project data	Every commit
College network drive	Archival storage	Semester end
External hard drive (lab)	Physical backup	Semester end

7. Procedure

7.1. File Naming Conventions

All data files must follow this naming pattern:

[project]_[datatype]_[YYYY-MM-DD].[ext]

For example: claremont_soilpH_2026-05-21.csv

Rules for file names:

1. Use lowercase letters only
2. Separate words with underscores (not spaces or hyphens)

3. Include the date of collection or generation in ISO 8601 format (YYYY-MM-DD)
4. Use `.csv` as the default file format for tabular data

 Tip

Never use spaces, special characters, or accented letters in file names. These cause problems across operating systems and with R's file-reading functions.

7.2. Directory Structure

Organize project data using a consistent directory tree:

```
project_name/  
  data/  
    raw/           # Unmodified instrument or field data  
    processed/     # Cleaned, validated datasets  
    metadata/      # Data dictionaries, codebooks  
  analysis/  
    scripts/       # R scripts and Quarto documents  
    output/        # Figures, tables, reports  
  README.md       # Project overview and data description
```

Raw data files in `data/raw/` must never be modified. All cleaning and transformation steps should produce new files in `data/processed/` with scripts that document every change (Wilson et al., 2017).

7.3. Data Entry from Lab Notebooks

1. Open the lab notebook to the relevant page
2. Create a new CSV file (or open the project template) in `data/raw/`
3. Enter column headers in the first row following the variable naming conventions below
4. Enter each observation as a separate row
5. Record the notebook page number, date, and analyst initials in the metadata file
6. Save and commit to Git with a descriptive message

7.4. Variable Naming Conventions

Column headers (variable names) must be:

1. Lowercase, with underscores separating words (e.g., `soil_ph`, `no3_mg_l`)
2. Free of spaces, special characters, and units in the name itself
3. Documented in an accompanying data dictionary (see Section 6.7)

Use consistent naming across related datasets. For example, always use `site_id` (not `Site`, `site`, `SiteID`, or `location`) when referring to the same sampling locations.

7.5. Data Types and Formatting

Data Type	Convention	Example
Dates	ISO 8601 (YYYY-MM-DD)	2026-05-21
Times	24-hour (HH:MM)	14:30
Missing values	Leave cell empty or use NA	NA
Numeric	No commas; use period for decimal	1234.56
Categorical	Consistent spelling; lowercase	upstream
Below detection	Flag column; numeric = MDL/2	See QC SOP

Common Mistakes

Never use 0 to represent a missing value. A zero is a valid measurement. Use NA or leave the cell blank. Never merge cells in a spreadsheet, and never use color coding as the sole carrier of information (Broman & Woo, 2018).

7.6. Data Validation in R

After entry, run the validation script to check for common errors. A minimal validation template:

```
library(tidyverse)

# Read the raw data
dat <- read_csv("data/raw/project_variable_2026-05-21.csv")

# --- Structure checks ---
glimpse(dat)                                # Column types correct?
stopifnot(nrow(dat) > 0)                    # File is not empty

# --- Missing data summary ---
dat %>%
  summarize(across(everything(), ~ sum(is.na(.)))) %>%
  pivot_longer(everything(), names_to = "variable",
               values_to = "n_missing") %>%
  filter(n_missing > 0)

# --- Range checks ---
dat %>%
  summarize(
    ph_min = min(soil_ph, na.rm = TRUE),
    ph_max = max(soil_ph, na.rm = TRUE)
  )
# Are values within physically plausible ranges?

# --- Duplicate check ---
```

```

dat %>%
  group_by(site_id, date) %>%
  filter(n() > 1)
# Flag unexpected duplicates for review

# --- Categorical consistency ---
dat %>% count(site_id, sort = TRUE)
# Look for typos (e.g., "upststream" vs "upstream")

```

7.7. Data Dictionaries (Metadata)

Every dataset must have a companion data dictionary saved as `data/metadata/[project]_datadictionary.csv` with the following columns:

Column	Description
<code>variable</code>	Column name in the data file
<code>description</code>	Plain-language description
<code>units</code>	Units of measurement (or “categorical”, “text”)
<code>type</code>	Data type: numeric, integer, character, date, logical
<code>range</code>	Expected range or valid categories
<code>source_sop</code>	SOP number for the method that generated the data

7.8. Version Control with Git

All data files and scripts must be tracked in a Git repository hosted on GitHub. Follow the procedures in [SOP-L801](#) for committing and pushing changes. Key rules:

1. Commit raw data files when they are first created (and never modify them after the initial commit)
2. Commit processed data files after each cleaning or transformation step
3. Write informative commit messages (e.g., “Add soil pH data from Apr 15 sampling” rather than “update”)
4. Use branches for experimental analyses; merge to `main` when finalized

7.9. Database Management for Multi-Year Projects

For projects spanning multiple semesters or involving multiple analysts, maintain a relational data structure:

1. Create a **sites table** (`data/metadata/sites.csv`) with location identifiers, coordinates, and site descriptions
2. Create a **samples table** (`data/metadata/samples.csv`) linking each sample to a site, date, and collector
3. Link analytical results to samples using unique `sample_id` values

This structure prevents duplication, supports merging across semesters, and enables relational joins in R (Wickham, 2014):

```
sites <- read_csv("data/metadata/sites.csv")
samples <- read_csv("data/metadata/samples.csv")
results <- read_csv("data/processed/chemistry_results.csv")

# Join all tables
full_data <- results %>%
  left_join(samples, by = "sample_id") %>%
  left_join(sites, by = "site_id")
```

8. Quality Control and Acceptance Criteria

- All data entries must be verified against the original lab notebook by a second person (or by the analyst after a 24-hour waiting period)
- Validation scripts must pass with zero unexpected errors before data are used in analysis
- The data dictionary must be complete for every variable in the dataset
- Refer to [SOP-L802](#) for analytical QC requirements that apply to the underlying measurements

9. Waste Disposal

Not applicable (computational procedure).

10. References

- Broman, K. W., & Woo, K. H. (2018). Data organization in spreadsheets. *The American Statistician*, 72(1), 2–10. <https://doi.org/10.1080/00031305.2017.1375989>
- Wickham, H. (2014). Tidy data. *Journal of Statistical Software*, 59(10), 1–23. <https://doi.org/10.18637/jss.v059.i10>
- Wilson, G., Bryan, J., Cranston, K., Kitzes, J., Nederbragt, L., & Teal, T. K. (2017). Good enough practices in scientific computing. *PLOS Computational Biology*, 13(6), e1005510. <https://doi.org/10.1371/journal.pcbi.1005510>

11. Training Record

Training is documented in the EA Program training log. Before working independently with project data, personnel must:

1. Read this SOP in full
2. Complete the RStudio and GitHub orientation ([SOP-L801](#))
3. Successfully enter, validate, and commit a practice dataset under supervision
4. Be signed off by the PI or lab manager

12. Appendix A: CSV Template

A minimal CSV template for a typical water chemistry dataset:

```
sample_id,site_id,date,time,analyst,ph,conductivity_us_cm,temp_c,notes
W001,site_a,2026-05-21,09:30,MLH,7.2,345,18.5,
W002,site_b,2026-05-21,10:15,MLH,6.8,412,19.1,slight turbidity
```

13. Appendix B: Recommended R Packages for Data Management

Package	Purpose
<code>tidyverse</code>	Data import, cleaning, and transformation
<code>janitor</code>	Consistent column names, tabulation
<code>validate</code>	Rule-based data validation
<code>pointblank</code>	Automated data quality reporting
<code>here</code>	Portable file paths across systems
<code>skimr</code>	Quick summary statistics



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L804

GIS and Spatial Data Management

Standard Operating Procedure

SOP Number	SOP-L804
Revision	0.1
Author(s)	Marc Los Huertos
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.
Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Responsibilities	2
4 Definitions	2
5 Safety and Hazards	3
6 Equipment and Materials	3
6.1 Software	3
6.2 Hardware	3
7 Procedure	3
7.1 Coordinate Reference Systems	3
7.2 File Formats	4
7.3 File Naming and Directory Structure	4
7.4 Spatial Data Acquisition	5
7.5 Spatial Data Processing in R	5
7.6 Map Production	6
7.7 Metadata Documentation	7
7.8 GPS Data Import	7
8 Quality Control and Acceptance Criteria	7
9 Waste Disposal	7
10 References	8
11 Training Record	8
12 Appendix A: Recommended R Packages for Spatial Analysis	8

1. Purpose

This SOP establishes procedures for acquiring, organizing, projecting, and archiving geospatial data used in EA Program research. It covers coordinate reference systems, file formats, metadata standards, and reproducible mapping workflows in R, ensuring that spatial data are interoperable across projects and analysts (Lovelace et al., 2019; Pebesma & Bivand, 2023).

2. Scope

- **Applies to:** All vector (point, line, polygon) and raster (satellite imagery, DEMs, interpolated surfaces) data used in EA Program projects
- **Does not apply to:** Drone image acquisition or photogrammetric processing (see [SOP-L805](#)); GPS receiver operation in the field (see relevant field SOPs)
- **Location:** EA Program laboratory (SGM 133) and RStudio Server

3. Responsibilities

Role	Responsibility
Lab PI	Approve spatial data management plan; ensure data-sharing compliance
GIS Lead	Maintain spatial data library; enforce projection and metadata standards
Analyst / Student	Follow naming conventions; document coordinate systems; archive final layers
Data Reviewer	Verify spatial metadata completeness before publication

4. Definitions

- **Coordinate reference system (CRS):** A framework that associates coordinates with locations on Earth; consists of a datum (reference ellipsoid), a projection (for 2D maps), and a coordinate system (e.g., latitude/longitude or meters); data layers must share the same CRS for spatial operations.
- **Geodatabase:** A structured container for storing multiple spatial datasets, attribute tables, and topological rules in a single file; the preferred format for complex GIS projects with many related layers.
- **GeoTIFF:** A TIFF image file with embedded geospatial metadata (CRS, extent, resolution); the standard raster format for spatial data exchange in remote sensing and GIS.
- **Raster:** A spatial data model representing continuous surfaces as a grid of equally sized cells (pixels), each with one or more attribute values; used for elevation, imagery, and interpolated surfaces.
- **Shapefile (.shp):** A common vector spatial data format introduced by Esri; stores geometry (points, lines, polygons) and attribute data in multiple associated files (.shp, .dbf, .shx,

.prj minimum); widely supported but limited by 2 GB file size and 10-character field name constraints.

- **Vector:** A spatial data model representing discrete features as points, lines, or polygons defined by coordinate pairs; used for sample locations, stream networks, and land parcel boundaries.
- **WGS84 (World Geodetic System 1984):** The global geodetic datum used by GPS; EPSG code 4326; the required CRS for all EA Program GPS-collected point data before projection into a local coordinate system.

5. Safety and Hazards

GIS work is a computational activity with no physical hazards. When using GPS receivers or conducting ground-truth fieldwork, follow [SOP-F001 Field Safety](#).

6. Equipment and Materials

6.1. Software

Item	Access	Notes
R (version 4.x)	RStudio Server	Primary analysis platform
sf package	CRAN	Vector data handling
terra package	CRAN	Raster data handling
tmap / ggplot2	CRAN	Map production
QGIS (3.x)	Free download	Visual inspection and digitizing
Google Earth Engine	earthengine.google.com	Satellite imagery processing

6.2. Hardware

Item	Location	Notes
GPS receivers (Garmin)	SGM 133 equipment cabinet	Field coordinate collection
High-resolution monitor	SGM 133	For digitizing and map review

7. Procedure

7.1. Coordinate Reference Systems

All EA Program spatial data must use one of the following coordinate reference systems (CRS). Document the CRS in every file, script, and metadata record.

CRS Name	EPSG Code	Use Case
WGS 84 (geographic)	4326	GPS data, web maps, data exchange
NAD83 / UTM Zone 11N	26911	Local analysis in Southern California
NAD83 / California Zone 5	2229	County and municipal data

Tip

When in doubt, store data in EPSG:4326 (latitude/longitude, WGS 84). Transform to a projected CRS for distance and area calculations.

When loading spatial data in R, always verify and set the CRS explicitly:

```
library(sf)

# Read a shapefile
sites <- st_read("data/spatial/sampling_sites.shp")

# Check CRS
st_crs(sites)

# Transform to UTM Zone 11N for area calculations
sites_utm <- st_transform(sites, crs = 26911)
```

7.2. File Formats

Format	Extension	Use Case	Notes
GeoPackage	.gpkg	Preferred for all vector data	Single file, open standard
Shapefile	.shp + .dbf + .shx + .prj	Legacy compatibility	Keep all component files together
GeoTIFF	.tif	Raster data	Include CRS in header
CSV with XY	.csv	Simple point data	Requires separate CRS documentation
GeoJSON	.geojson	Web applications	Always WGS 84

GeoPackage (.gpkg) is the preferred format for new projects because it stores geometry, attributes, and CRS in a single file without the limitations of shapefiles (10-character field names, 2 GB size limit) (Pebesma & Bivand, 2023).

7.3. File Naming and Directory Structure

Follow the conventions in [SOP-L803](#) with the addition of a `data/spatial/` subdirectory:

```

project_name/
  data/
    raw/
    processed/
    spatial/
      raw/          # Downloaded or collected spatial data
      processed/    # Clipped, projected, joined layers
      metadata/     # FGDC or ISO metadata files
    metadata/
  analysis/
    scripts/
    maps/           # Exported map figures
  README.md

```

7.4. Spatial Data Acquisition

Common data sources for EA Program research:

Source	URL	Data Types
USGS National Map	nationalmap.gov	DEMs, hydrography, boundaries
USGS EarthExplorer	earthexplorer.usgs.gov	Landsat, NAIP imagery
CA State Geoportal	gis.data.ca.gov	State and county layers
NRCS Web Soil Survey	websoilsurvey.sc.egov.usda.gov	SSURGO soil data
OpenStreetMap	openstreetmap.org	Roads, buildings, land use
Google Earth Engine	earthengine.google.com	Multi-temporal satellite data

When downloading external data, record the source URL, download date, and any subsetting parameters in the metadata file.

7.5. Spatial Data Processing in R

A standard workflow for processing spatial data:

```

library(sf)
library(terra)
library(tidyverse)

# --- 1. Load vector data ---
sites <- st_read("data/spatial/raw/sampling_sites.gpkg")
watershed <- st_read("data/spatial/raw/claremont_watershed.gpkg")

```

```
# --- 2. Verify and harmonize CRS ---
st_crs(sites) == st_crs(watershed)
sites <- st_transform(sites, crs = st_crs(watershed))

# --- 3. Spatial operations ---
# Clip sites to watershed boundary
sites_in_watershed <- st_intersection(sites, watershed)

# Buffer sites by 100 m
site_buffers <- st_buffer(sites_in_watershed, dist = 100)

# --- 4. Load and extract raster data ---
dem <- rast("data/spatial/raw/dem_10m.tif")
sites_in_watershed$elevation_m <- extract(dem, sites_in_watershed)$value

# --- 5. Save processed outputs ---
st_write(sites_in_watershed,
         "data/spatial/processed/sites_with_elevation.gpkg",
         delete_dsn = TRUE)
```

7.6. Map Production

All maps for reports and publications must include:

1. Title and date
2. Scale bar (use `tm_map::tm_scale_bar()` or `ggspatial::annotation_scale()`)
3. North arrow
4. Legend with clear labels and units
5. Data source attribution
6. CRS/projection noted in the caption or margin

```
library(tmap)

tm_shape(watershed) +
  tm_polygons(col = "lightblue", alpha = 0.3) +
  tm_shape(sites_in_watershed) +
  tm_dots(col = "elevation_m", size = 0.5,
         palette = "viridis",
         title = "Elevation (m)") +
  tm_scale_bar(position = c("left", "bottom")) +
  tm_compass(position = c("right", "top")) +
  tm_layout(
    title = "Sampling Sites: Claremont Watershed",
    title.size = 1.2,
    legend.outside = TRUE
  )
```

7.7. Metadata Documentation

Each spatial dataset must have a metadata record. At minimum, include:

Field	Description
Title	Descriptive name of the layer
Abstract	Brief description of content and purpose
CRS	EPSG code and full name
Extent	Bounding box coordinates
Source	Origin of data (URL, agency, field GPS)
Date created/acquired	When the data were generated or downloaded
Processing steps	Summary of transformations applied
Contact	Analyst name and email

For formal publications, use FGDC or ISO 19115 metadata standards. For internal projects, the table above saved as a text file alongside the data is sufficient.

7.8. GPS Data Import

When importing GPS waypoints or tracks:

1. Download data from the GPS unit to `data/spatial/raw/` in GPX format
2. Import into R using `sf::st_read("file.gpx", layer = "waypoints")`
3. Verify coordinates visually by plotting over a basemap
4. Assign the correct CRS (typically EPSG:4326 for consumer GPS units)
5. Document the GPS model, estimated accuracy, and collection conditions

8. Quality Control and Acceptance Criteria

- Verify CRS for every imported dataset before analysis
- Check that point locations fall within expected geographic bounds (e.g., all sites should be within the study area polygon)
- Validate topology for polygon layers (no gaps, overlaps, or self-intersections) using `sf::st_is_valid()`
- Compare GPS coordinates with known reference points when available
- Document the spatial accuracy and precision of each data source

9. Waste Disposal

Not applicable (computational procedure).

10. References

- Lovelace, R., Nowosad, J., & Muenchow, J. (2019). *Geocomputation with R*. Chapman; Hall/CRC. <https://doi.org/10.1201/9780203730058>
- Pebesma, E., & Bivand, R. (2023). *Spatial data science: With applications in R*. Chapman; Hall/CRC. <https://doi.org/10.1201/9780429459016>

11. Training Record

Training is documented in the EA Program training log. Before working independently with spatial data, personnel must:

1. Read this SOP in full
2. Complete the RStudio and GitHub orientation ([SOP-L801](#))
3. Demonstrate proficiency with **sf** and basic spatial operations in R under supervision
4. Produce a correctly formatted map with all required elements
5. Be signed off by the PI or GIS lead

12. Appendix A: Recommended R Packages for Spatial Analysis

Package	Purpose
sf	Vector data (points, lines, polygons)
terra	Raster data analysis
stars	Spatiotemporal raster/vector data cubes
tmap	Thematic maps (static and interactive)
leaflet	Interactive web maps
ggspatial	Spatial layers in ggplot2
mapview	Quick interactive map previews
exactextractr	Fast raster extraction by polygons
geodata	Download climate, soil, and elevation data
tidyterra	Tidy methods for terra objects



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L805

Drone Image Processing (Photogrammetry)

Standard Operating Procedure

SOP Number	SOP-L805
Revision	0.1
Author(s)	Marc Los Huertos
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.

Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Responsibilities	2
4 Definitions	2
5 Safety and Hazards	3
6 Equipment and Materials	3
6.1 Software	3
6.2 Hardware	3
6.3 Drone Platforms	3
7 Procedure	4
7.1 Image Organization	4
7.2 Ground Control Points (GCPs)	5
7.3 Photogrammetric Processing (Agisoft Metashape)	5
7.4 OpenDroneMap (Alternative Workflow)	6
7.5 Accuracy Assessment	7
7.6 Multispectral Processing	7
8 Quality Control and Acceptance Criteria	8
9 Waste Disposal	8
10 References	8
11 Training Record	8
12 Appendix A: Processing Parameter Reference	9
13 Appendix B: Troubleshooting	9

1. Purpose

This SOP describes the post-processing workflow for aerial imagery collected by uncrewed aerial systems (UAS/drones). It covers image organization, photogrammetric reconstruction (orthomosaics, digital surface models, and point clouds), ground control point integration, accuracy assessment, and data archiving (Anderson & Gaston, 2013; Iglhaut et al., 2019).

2. Scope

- **Applies to:** Photogrammetric processing of RGB, multispectral, and thermal drone imagery collected for EA Program research
- **Does not apply to:** Drone flight planning, airspace authorization, or FAA Part 107 compliance (covered in the drone operations field SOP); GIS analysis of processed outputs (see [SOP-L804](#))
- **Location:** EA Program laboratory (SGM 133) and processing workstations

3. Responsibilities

Role	Responsibility
Lab PI	Approve processing workflows; ensure data storage compliance
Drone Pilot / Operator	Deliver organized imagery with flight logs; place and survey GCPs
Processing Analyst	Run photogrammetric software; assess quality; archive outputs
Data Reviewer	Verify accuracy metrics before outputs are used in analysis

4. Definitions

- **Digital elevation model (DEM):** A raster representing bare-ground surface elevation; derived by removing vegetation and structures from a digital surface model using filtering algorithms.
- **Digital surface model (DSM):** A raster representing the elevation of all surfaces in the scene, including vegetation canopy and structures; the direct output of photogrammetric point cloud processing.
- **Ground control point (GCP):** A precisely surveyed point on the ground (surveyed with RTK-GPS or total station) that appears in multiple overlapping images; used to georeference the photogrammetric model to a known coordinate system and improve absolute accuracy.
- **Orthomosaic:** A geometrically corrected aerial image composed of many overlapping individual photos, in which scale is uniform and features can be measured accurately; produced by photogrammetric processing.
- **Photogrammetry:** The science of making measurements from photographs; in drone operations, the process of generating 3D models, DEMs, and orthomosaics from overlapping aerial images using structure-from-motion algorithms.

- **Point cloud:** A set of data points in 3D space (X, Y, Z coordinates) representing the surface of objects; generated by photogrammetric processing of overlapping images; the intermediate product from which DEMs and DSMs are derived.
- **Structure from Motion (SfM):** A photogrammetric algorithm that reconstructs 3D geometry from 2D overlapping images by identifying and matching common features across multiple images; the computational core of drone photogrammetry software (Agisoft Metashape, Pix4D).

5. Safety and Hazards

Hazards

Photogrammetric processing is computationally intensive but poses no physical hazards. However, extended processing sessions may generate significant heat from workstation components. Ensure adequate ventilation around processing workstations.

For field-related hazards during drone operations and GCP placement, refer to [SOP-F001 Field Safety](#).

6. Equipment and Materials

6.1. Software

Item	Access	Notes
Agisoft Metashape Professional	Campus license	Primary photogrammetry tool
OpenDroneMap (ODM)	Free / open source	Alternative processing
WebODM	Self-hosted	Browser interface for ODM
QGIS (3.x)	Free download	Visualization and QC
R with <code>terra</code> / <code>sf</code>	RStudio Server	Post-processing analysis
CloudCompare	Free download	Point cloud inspection

6.2. Hardware

Item	Minimum Specs	Notes
Processing workstation	32 GB RAM, dedicated GPU, SSD	More RAM for large datasets
External storage	2 TB+ portable SSD	Raw image archive

6.3. Drone Platforms

Platform	Sensor	Resolution
DJI Mavic 3 Enterprise	RGB (Hasselblad)	20 MP; 0.7 cm/px at 100 m
DJI Matrice 350 RTK	Zenmuse P1 / L2	High-res RGB or LiDAR
Custom multispectral rig	MicaSense RedEdge-P	5-band multispectral

7. Procedure

7.1. Image Organization

Before processing, organize raw imagery following a consistent directory structure:

```
project_name/  
  drone/  
    raw_images/  
      flight_2026-05-21_rgb/  
        DJI_0001.JPG  
        DJI_0002.JPG  
        ...  
      flight_2026-05-21_multi/  
    flight_logs/  
      flight_2026-05-21.csv  
    gcp/  
      gcp_coordinates.csv  
    processed/  
      orthomosaic/  
      dsm/  
      point_cloud/  
      report/  
    README.md
```

Check that all images have valid EXIF data (GPS coordinates, camera parameters). Remove obviously corrupted or blurry images before processing.

```
library(exifr)  
  
# Read EXIF from all images in a flight folder  
exif_data <- read_exif(  
  list.files("drone/raw_images/flight_2026-05-21_rgb/",  
    pattern = "\\\\.JPG$", full.names = TRUE),  
  tags = c("FileName", "GPSLatitude", "GPSLongitude",  
    "GPSAltitude", "FocalLength", "ImageWidth",  
    "ImageHeight", "DateTimeOriginal")  
)  
  
# Check for missing GPS
```

```
missing_gps <- exif_data %>%  
  filter(is.na(GPSLatitude) | is.na(GPSLongitude))  
cat(sprintf("%d images missing GPS coordinates\n", nrow(missing_gps)))  
  
# Summary  
cat(sprintf("Total images: %d\n", nrow(exif_data)))  
cat(sprintf("Date range: %s to %s\n",  
  min(exif_data$DateTimeOriginal),  
  max(exif_data$DateTimeOriginal)))
```

7.2. Ground Control Points (GCPs)

GCPs are surveyed reference points used to improve the geometric accuracy of photogrammetric outputs.

1. Place GCPs before the flight. Use high-contrast targets (black and white checkerboard, minimum 30 cm) visible from flight altitude
2. Survey GCP positions using a differential GPS or RTK receiver. Record coordinates in WGS 84 (EPSG:4326) and a local projected CRS (e.g., EPSG:26911)
3. Format GCP data as a CSV file:

```
gcp_id,longitude,latitude,elevation_m,crs  
GCP01,-117.7123,34.0966,398.2,EPSG:4326  
GCP02,-117.7118,34.0971,401.5,EPSG:4326
```

4. Use a minimum of 5 GCPs distributed across the study area. Reserve 2 or more as independent check points for accuracy assessment

7.3. Photogrammetric Processing (Agisoft Metashape)

The following workflow applies to Agisoft Metashape Professional. Adapt parameters for your specific project needs.

Step 1: Create project and import photos

1. Create a new Metashape project and save to **drone/processed/**
2. Add photos from the flight folder
3. Verify camera positions on the reference pane

Step 2: Align photos

1. Workflow > Align Photos
2. Settings: Accuracy = High; Key point limit = 40,000; Tie point limit = 10,000; Adaptive camera model fitting = enabled
3. Review the sparse point cloud and camera positions

Step 3: Import and assign GCPs

1. Import GCP coordinates from the CSV file
2. Assign each GCP to its corresponding target in at least 3 photos
3. Uncheck GCPs designated as check points
4. Run Optimize Cameras

Step 4: Build dense point cloud

1. Workflow > Build Point Cloud
2. Quality = High (or Medium for initial assessment); Depth filtering = Moderate
3. Inspect the dense cloud for artifacts; remove obvious noise manually or with CloudCompare

Step 5: Build DEM and orthomosaic

1. Workflow > Build DEM (Source: Point Cloud)
2. Workflow > Build Orthomosaic (Surface: DEM; Blending mode: Mosaic)

Step 6: Export products

1. Export DEM as GeoTIFF to `drone/processed/dsm/`
2. Export orthomosaic as GeoTIFF to `drone/processed/orthomosaic/`
3. Export point cloud as `.laz` to `drone/processed/point_cloud/`
4. Export processing report as PDF to `drone/processed/report/`

Tip

For large datasets (more than 1,000 images), process on the lab workstation rather than a laptop. Expect processing times of several hours for dense cloud generation at High quality.

7.4. OpenDroneMap (Alternative Workflow)

For projects using the open-source pipeline:

```
# Run ODM with standard settings
docker run -ti --rm \
  -v /path/to/project/drone:/datasets \
  opendronemap/odm \
  --project-path /datasets \
  --dsm \
  --orthophoto-resolution 2.0 \
  --dem-resolution 5.0 \
  --gcp /datasets/gcp/gcp_list.txt
```

7.5. Accuracy Assessment

Evaluate positional accuracy using the reserved check points:

```
library(terra)
library(sf)

# Load the DSM
dsm <- rast("drone/processed/dsm/dsm_2026-05-21.tif")

# Load check points
check_pts <- read_csv("drone/gcp/check_points.csv") %>%
  st_as_sf(coords = c("longitude", "latitude"), crs = 4326) %>%
  st_transform(crs = crs(dsm))

# Extract DSM elevation at check points
check_pts$dsm_elev <- extract(dsm, check_pts)$value

# Calculate vertical RMSE
check_pts <- check_pts %>%
  mutate(error_m = dsm_elev - elevation_m)

rmse_z <- sqrt(mean(check_pts$error_m^2, na.rm = TRUE))
mae_z <- mean(abs(check_pts$error_m), na.rm = TRUE)

cat(sprintf("Vertical RMSE: %.3f m\n", rmse_z))
cat(sprintf("Vertical MAE: %.3f m\n", mae_z))
cat(sprintf("n check points: %d\n", nrow(check_pts)))
```

Report accuracy metrics in all publications and project reports. Typical targets for environmental applications: horizontal RMSE less than 2 times the ground sampling distance (GSD); vertical RMSE less than 3 times the GSD (James et al., 2017).

7.6. Multispectral Processing

For multispectral imagery (e.g., MicaSense RedEdge):

1. Include reflectance calibration panel images (taken before and after each flight)
2. Process each band alignment in Metashape or use MicaSense's Atlas or `micasense` Python library
3. Export calibrated reflectance orthomosaics (not raw DN values)
4. Calculate vegetation indices in R:

```
library(terra)

# Load calibrated reflectance bands
```

```
red <- rast("drone/processed/orthomosaic/red_reflectance.tif")
nir <- rast("drone/processed/orthomosaic/nir_reflectance.tif")

# Calculate NDVI
ndvi <- (nir - red) / (nir + red)
names(ndvi) <- "NDVI"

writeRaster(ndvi, "drone/processed/orthomosaic/ndvi_2026-05-21.tif",
            overwrite = TRUE)
```

8. Quality Control and Acceptance Criteria

- Photo alignment: at least 90% of images must align successfully
- Reprojection error: mean below 1.0 pixel after GCP optimization
- Check point RMSE: document and compare against project requirements
- Visual inspection: no visible seam lines, ghosting, or geometric distortion in the orthomosaic
- File integrity: verify that all exported GeoTIFFs have valid CRS headers and correct extent

9. Waste Disposal

Not applicable (computational procedure).

10. References

- Anderson, K., & Gaston, K. J. (2013). Lightweight unmanned aerial vehicles will revolutionize spatial ecology. *Frontiers in Ecology and the Environment*, 11(3), 138–146. <https://doi.org/10.1890/120150>
- Ighhaut, J., Cabo, C., Puliti, S., Piermattei, L., O'Connor, J., & Rosette, J. (2019). Structure from motion photogrammetry in forestry: A review. *Current Forestry Reports*, 5(3), 155–168. <https://doi.org/10.1007/s40725-019-00094-3>
- James, M. R., Robson, S., d'Oleire-Oltmanns, S., & Niethammer, U. (2017). Optimising UAV topographic surveys processed with structure-from-motion: Ground control quality, quantity and bundle adjustment. *Geomorphology*, 280, 51–66. <https://doi.org/10.1016/j.geomorph.2016.11.021>

11. Training Record

Training is documented in the EA Program training log. Before processing drone imagery independently, personnel must:

1. Read this SOP in full
2. Complete Agisoft Metashape (or ODM) training with a supervised practice dataset

3. Demonstrate a complete processing workflow from raw images to accuracy-assessed outputs
4. Be signed off by the PI or drone operations lead

12. Appendix A: Processing Parameter Reference

Parameter	Low Quality	Medium	High (Recommended)
Photo alignment	Low accuracy	Medium	High
Dense cloud	Low	Medium	High
DEM resolution	10x GSD	5x GSD	2x GSD
Orthomosaic	10x GSD	5x GSD	Native GSD
Approx. processing time (500 images)	30 min	2 hr	6 hr+

13. Appendix B: Troubleshooting

Symptom	Probable Cause	Corrective Action
Low alignment rate (<80%)	Insufficient overlap	Re-fly with 80%+ front/side overlap
High reprojection error	Incorrect GCP placement	Re-identify GCP markers in photos
Artifacts in orthomosaic	Moving objects (water, cars)	Mask affected areas; adjust blending
Out-of-memory error	Dataset too large for RAM	Reduce quality; split into chunks
DSM shows “doming” distortion	No GCPs; insufficient geometry	Add GCPs; increase flight line diversity



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L806

Statistical Analysis Templates (R)

Standard Operating Procedure

SOP Number	SOP-L806
Revision	0.1
Author(s)	Marc Los Huertos
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.

Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Responsibilities	2
4 Definitions	2
5 Safety and Hazards	3
6 Equipment and Materials	3
6.1 Software	3
6.2 Core R Packages	3
7 Procedure	3
7.1 Analysis Workflow Overview	3
7.2 Quarto Document Header	4
7.3 Exploratory Data Analysis Template	4
7.4 Two-Sample Comparisons	5
7.5 One-Way ANOVA Template	6
7.6 Linear Regression Template	6
7.7 Mixed-Effects Model Template	7
7.8 Multivariate / Community Analysis Template	8
7.9 Figure Formatting Standards	8
7.10 Reporting Statistical Results	9
8 Quality Control and Acceptance Criteria	9
9 Waste Disposal	9
10 References	9
11 Training Record	10
12 Appendix A: Decision Guide for Choosing a Statistical Test	10
13 Appendix B: Common Data Transformations	10

1. Purpose

This SOP provides standardized R analysis templates for common statistical procedures used in EA Program research. It establishes reproducible workflows for exploratory data analysis, hypothesis testing, regression modeling, and reporting, ensuring that all analyses are transparent, well-documented, and consistent across projects (Wickham et al., 2023; Zuur et al., 2010).

2. Scope

- **Applies to:** All quantitative statistical analyses performed for EA Program research, coursework, and publications
- **Does not apply to:** Bioinformatics sequence analysis (see [SOP-L705](#)); GIS-specific spatial statistics (see [SOP-L804](#)); instrument calibration curve fitting (see method-specific SOPs)
- **Location:** RStudio Server (rstudio.pomona.edu)

3. Responsibilities

Role	Responsibility
Lab PI	Approve analytical approach; review statistical methods in manuscripts
Analyst / Student	Select appropriate methods; follow templates; document all decisions
Data Reviewer	Verify that assumptions are checked and reported
Statistical Advisor	Consult on complex designs (mixed models, multivariate methods)

4. Definitions

- **ANOVA (Analysis of Variance):** A statistical test that partitions total variance in a response variable into components attributable to one or more categorical factors; tests whether group means differ significantly; assumes normality and homogeneity of variance.
- **Confidence interval:** A range of values computed from sample data that, under repeated sampling, contains the true population parameter with a stated probability (e.g., 95%); wider intervals reflect greater uncertainty.
- **Effect size:** A standardized measure of the magnitude of a difference or relationship, independent of sample size; examples include Cohen's d (mean differences), r^2 (explained variance), and Hedge's g; should always accompany p-values.
- **Mixed model:** A statistical model containing both fixed effects (factors whose levels are deliberately chosen) and random effects (factors whose levels represent a random sample from a larger population); appropriate when data have hierarchical or repeated-measures structure.
- **p-value:** The probability of observing a test statistic as extreme as the one obtained (or more extreme) if the null hypothesis is true; does not measure the probability that the null hypothesis is true; must be interpreted with effect size.

- **Power:** The probability of correctly rejecting the null hypothesis when it is false; increases with sample size, effect size, and significance level; should be considered in study design to ensure adequate sample sizes.
- **Residual:** The difference between an observed value and the value predicted by a statistical model; examination of residuals is the primary diagnostic for checking model assumptions.

5. Safety and Hazards

Statistical analysis is a computational activity with no physical hazards.

6. Equipment and Materials

6.1. Software

Item	Access	Notes
R (version 4.x) RStudio IDE	RStudio Server rstudio.pomona.edu	Primary analysis platform Integrated development environment
Quarto Git / GitHub	Pre-installed github.com/marclos	Report generation Version control

6.2. Core R Packages

Package	Purpose
tidyverse	Data wrangling and visualization
broom	Tidy model output
car	Diagnostic tests (Levene, VIF)
lme4	Mixed-effects models
emmeans	Estimated marginal means and contrasts
vegan	Community ecology and ordination
MuMIn	Model selection and multi-model inference
performance	Model diagnostics and assumption checks
report	Automated statistical reporting
ggplot2	Publication-quality figures
knitr / kableExtra	Formatted tables

7. Procedure

7.1. Analysis Workflow Overview

Every statistical analysis should follow this sequence:

1. State the research question and hypotheses
2. Import and validate data (see [SOP-L803](#))
3. Conduct exploratory data analysis (EDA)
4. Check assumptions for the intended statistical test
5. Fit the model
6. Evaluate model diagnostics
7. Interpret results and generate figures/tables
8. Write up methods and results in a Quarto document
9. Commit the complete analysis to Git

7.2. Quarto Document Header

Begin every analysis with a standard Quarto header:

```
---
title: "Analysis: [Project Name]"
author: "[Your Name]"
date: today
format:
  html:
    toc: true
    code-fold: true
    embed-resources: true
bibliography: references.bib
---
```

7.3. Exploratory Data Analysis Template

```
library(tidyverse)

# --- 1. Load and inspect data ---
dat <- read_csv("data/processed/project_data.csv")
glimpse(dat)

# --- 2. Summary statistics ---
dat %>%
  group_by(treatment) %>%
  summarize(
    n = n(),
    mean = mean(response, na.rm = TRUE),
    sd = sd(response, na.rm = TRUE),
    median = median(response, na.rm = TRUE),
    min = min(response, na.rm = TRUE),
    max = max(response, na.rm = TRUE)
  )
```

```
# --- 3. Distributional plots ---
ggplot(dat, aes(x = response)) +
  geom_histogram(bins = 20, fill = "steelblue", color = "white") +
  facet_wrap(~ treatment) +
  labs(x = "Response Variable (units)", y = "Count") +
  theme_minimal()

ggplot(dat, aes(x = treatment, y = response)) +
  geom_boxplot(fill = "steelblue", alpha = 0.5) +
  geom_jitter(width = 0.1, alpha = 0.4) +
  labs(x = "Treatment", y = "Response Variable (units)") +
  theme_minimal()

# --- 4. Missing data assessment ---
dat %>%
  summarize(across(everything(), ~ sum(is.na(.)))) %>%
  pivot_longer(everything(), names_to = "variable",
               values_to = "n_missing") %>%
  filter(n_missing > 0)
```

7.4. Two-Sample Comparisons

```
# --- Check normality within groups ---
dat %>%
  group_by(treatment) %>%
  summarize(
    shapiro_p = shapiro.test(response)$p.value
  )

# --- Check equal variances ---
car::leveneTest(response ~ treatment, data = dat)

# --- Parametric: two-sample t-test ---
t_result <- t.test(response ~ treatment, data = dat, var.equal = FALSE)
t_result

# --- Nonparametric alternative: Wilcoxon rank-sum ---
wilcox.test(response ~ treatment, data = dat)

# --- Effect size ---
library(effects)
cohens_d(response ~ treatment, data = dat)
```

7.5. One-Way ANOVA Template

```
# --- Fit ANOVA ---
aov_model <- aov(response ~ treatment, data = dat)

# --- Check assumptions ---
# Normality of residuals
shapiro.test(residuals(aov_model))
qqnorm(residuals(aov_model)); qqline(residuals(aov_model))

# Homogeneity of variance
car::leveneTest(response ~ treatment, data = dat)

# Residual vs. fitted plot
plot(aov_model, which = 1)

# --- ANOVA table ---
summary(aov_model)

# --- Post-hoc pairwise comparisons ---
library(emmeans)
emm <- emmeans(aov_model, ~ treatment)
pairs(emm, adjust = "tukey")

# --- Nonparametric alternative: Kruskal-Wallis ---
kruskal.test(response ~ treatment, data = dat)
```

Tip

Always check assumptions before interpreting p-values. If residuals are non-normal or variances are unequal, consider a transformation (log, square root) or switch to a nonparametric test. Document the reasoning in your analysis script (Zuur et al., 2010).

7.6. Linear Regression Template

```
# --- Fit linear model ---
lm_model <- lm(response ~ predictor1 + predictor2, data = dat)

# --- Model summary ---
summary(lm_model)

# --- Tidy output ---
library(broom)
tidy(lm_model, conf.int = TRUE)
```

```
glance(lm_model)

# --- Diagnostics ---
par(mfrow = c(2, 2))
plot(lm_model)
par(mfrow = c(1, 1))

# Check multicollinearity
car::vif(lm_model)

# --- Model performance ---
library(performance)
check_model(lm_model)

# --- Visualization ---
ggplot(dat, aes(x = predictor1, y = response)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = TRUE, color = "steelblue") +
  labs(x = "Predictor (units)", y = "Response (units)",
       caption = sprintf("R-squared = %.3f, p = %.4f",
                          summary(lm_model)$r.squared,
                          glance(lm_model)$p.value)) +
  theme_minimal()
```

7.7. Mixed-Effects Model Template

For data with nested or repeated-measures structure:

```
library(lme4)
library(lmerTest) # For p-values via Satterthwaite

# --- Fit mixed model ---
# Fixed effect: treatment
# Random effect: site (multiple samples per site)
mixed_model <- lmer(response ~ treatment + (1 | site_id), data = dat)

# --- Summary with p-values ---
summary(mixed_model)

# --- Marginal and conditional R-squared ---
MuMIn::r.squaredGLMM(mixed_model)

# --- Post-hoc comparisons ---
library(emmeans)
emm <- emmeans(mixed_model, ~ treatment)
pairs(emm, adjust = "tukey")
```

```
# --- Residual diagnostics ---  
plot(mixed_model)  
qqnorm(resid(mixed_model)); qqline(resid(mixed_model))
```

7.8. Multivariate / Community Analysis Template

```
library(vegan)  
  
# --- Prepare community matrix ---  
# Rows = sites, columns = species/variables  
comm_matrix <- dat %>%  
  select(site_id, species, abundance) %>%  
  pivot_wider(names_from = species, values_from = abundance,  
              values_fill = 0) %>%  
  column_to_rownames("site_id")  
  
# --- Diversity indices ---  
diversity_h <- diversity(comm_matrix, index = "shannon")  
richness <- specnumber(comm_matrix)  
  
# --- Ordination (NMDS) ---  
nmfs <- metaMDS(comm_matrix, distance = "bray", k = 2,  
               trymax = 100)  
stressplot(nmfs)  
  
# Plot  
ordiplot(nmfs, type = "n")  
points(nmfs, col = as.factor(env_data$treatment), pch = 16)  
legend("topright", legend = levels(as.factor(env_data$treatment)),  
      col = 1:nlevels(as.factor(env_data$treatment)), pch = 16)  
  
# --- PERMANOVA ---  
adonis2(comm_matrix ~ treatment, data = env_data,  
        method = "bray", permutations = 999)
```

7.9. Figure Formatting Standards

All figures for reports and publications must follow these conventions:

1. Use `theme_minimal()` or a consistent custom theme
2. Label axes with variable names and units
3. Use colorblind-friendly palettes (e.g., `viridis`, `RColorBrewer` “Set2”)
4. Include figure captions using the `#| fig-cap:` chunk option
5. Save publication figures at 300 DPI as both PDF and PNG:

```
ggsave("analysis/output/fig_treatment_response.pdf",
       width = 6, height = 4, units = "in")
ggsave("analysis/output/fig_treatment_response.png",
       width = 6, height = 4, units = "in", dpi = 300)
```

7.10. Reporting Statistical Results

When writing results in a Quarto document, follow APA conventions:

- Report test statistics, degrees of freedom, and p-values: “Treatment had a significant effect on soil pH ($F(2, 27) = 4.53$, $p = 0.020$)”
- Report effect sizes alongside p-values
- Include confidence intervals for parameter estimates
- Round to a consistent number of decimal places (typically 2 for test statistics, 3 for p-values)
- Use `broom::tidy()` and `knitr::kable()` for formatted tables

```
library(broom)
library(knitr)

tidy(aov_model) %>%
  kable(digits = 3, caption = "ANOVA results for soil pH by treatment")
```

8. Quality Control and Acceptance Criteria

- Every analysis must include explicit assumption checks (normality, homoscedasticity, independence) documented in the script
- All models must include diagnostic plots
- Results must be reproducible: running the script from start to finish on the same data must produce identical output
- Analyses must be committed to Git with clear commit messages
- The PI or a designated reviewer must approve the analytical approach before results are included in publications

9. Waste Disposal

Not applicable (computational procedure).

10. References

Wickham, H., cCetinkaya-Rundel, M., & Golemund, G. (2023). *R for data science* (2nd ed.). O'Reilly Media. <https://r4ds.hadley.nz/>

Zuur, A. F., Ieno, E. N., & Elphick, C. S. (2010). A protocol for data exploration to avoid common statistical problems. *Methods in Ecology and Evolution*, 1(1), 3–14. <https://doi.org/10.1111/j.2041-210X.2009.00001.x>

11. Training Record

Training is documented in the EA Program training log. Before conducting statistical analyses independently, personnel must:

1. Read this SOP in full
2. Complete the RStudio and GitHub orientation ([SOP-L801](#))
3. Complete the introductory R statistics module (see [RTricks](#))
4. Demonstrate a complete analysis workflow (EDA through reporting) on a practice dataset under supervision
5. Be signed off by the PI

12. Appendix A: Decision Guide for Choosing a Statistical Test

Research Question	Data Type	Groups	Recommended Test
Compare 2 group means	Continuous, normal	2	Two-sample t-test
Compare 2 group medians	Non-normal	2	Wilcoxon rank-sum
Compare 3+ group means	Continuous, normal	3+	One-way ANOVA
Compare 3+ group medians	Non-normal	3+	Kruskal-Wallis
Relationship between 2 variables	Both continuous	N/A	Linear regression / correlation
Multiple predictors	Continuous response	N/A	Multiple regression
Nested / repeated measures	Continuous	Any	Mixed-effects model
Community composition	Abundance matrix	Any	NMDS + PERMANOVA
Categorical association	Counts	Any	Chi-squared / Fisher's exact

13. Appendix B: Common Data Transformations

Transformation	R Code	When to Use
Log (natural)	<code>log(x)</code>	Right-skewed data; multiplicative relationships
Log (base 10)	<code>log10(x)</code>	Concentration data spanning orders of magnitude
Square root	<code>sqrt(x)</code>	Count data; moderate skew
Arcsine square root	<code>asin(sqrt(x))</code>	Proportions (0 to 1)

Transformation	R Code	When to Use
Box-Cox	<code>MASS::boxcox(model)</code>	Let the data determine the best transformation

 Caution

Always report whether data were transformed and provide results on both the transformed and original scales when possible. Transformations change the hypothesis being tested (e.g., log-transformed ANOVA compares geometric means, not arithmetic means).



Environmental Analysis Program

Pomona College
Biogeochemistry Laboratory

SOP-L807

Instrument Data Export and File Management

Standard Operating Procedure

SOP Number	SOP-L807
Revision	0.1
Author(s)	Marc Los Huertos
Date	2026-06-26
Organization	EA Program, Pomona College

FOR INTERNAL USE ONLY

This document is the property of the Environmental Analysis Program at Pomona College.
Unauthorized distribution is not permitted.

Table of contents

1 Purpose	2
2 Scope	2
3 Responsibilities	2
4 Definitions	2
5 Safety and Hazards	3
6 Equipment and Materials	3
6.1 Instruments and Export Formats	3
6.2 Transfer Media	3
7 Procedure	4
7.1 General Export Workflow	4
7.2 Instrument-Specific Export Instructions	4
7.2.1 Shimadzu UV-Vis (UV-1900i / UVProbe)	4
7.2.2 Thermo iCAP ICP-OES (Qtegra)	4
7.2.3 Shimadzu TOC-L (TOC-Control)	5
7.2.4 YSI ProDSS Field Sensor	5
7.2.5 Bio-Rad QX200 ddPCR (QuantaSoft)	5
7.3 File Transfer to RStudio Server	5
7.4 Automated Import Scripts in R	5
7.5 Data Integrity Checks	6
7.6 Archival and Backup	7
7.7 Instrument Log Cross-Reference	7
8 Quality Control and Acceptance Criteria	7
9 Waste Disposal	8
10 References	8
11 Training Record	8
12 Appendix A: File Hash Verification	8

13 Appendix B: Instrument Data Format Quick Reference

8

1. Purpose

This SOP standardizes the procedures for exporting raw data from laboratory instruments, transferring files to the data management system, and organizing instrument outputs for downstream analysis. It ensures that raw instrument data are preserved without modification, properly documented, and accessible for reproducible analysis (Gandrud, 2015).

2. Scope

- **Applies to:** All analytical instruments in the EA Program laboratory that produce digital data files, including spectrophotometers, ICP-OES, TOC analyzers, ion chromatographs, balances with data output, pH/conductivity meters with logging, and field sensors that store data internally
- **Does not apply to:** Manual readings recorded directly in lab notebooks (see [SOP-L010](#)); drone imagery (see [SOP-L805](#)); bioinformatics sequencing data (see [SOP-L705](#))
- **Location:** EA Program laboratory (SGM 133)

3. Responsibilities

Role	Responsibility
Lab PI	Approve file management protocols; ensure backup compliance
Lab Manager	Maintain instrument computers; manage shared storage; train users on export procedures
Analyst / Student	Export data following this SOP; verify file integrity; commit to version control
Data Reviewer	Confirm that raw files match lab notebook entries and are properly archived

4. Definitions

- **Comma-separated values (CSV):** A plain-text file format in which values are separated by commas and rows by line breaks; universally readable; the preferred format for sharing tabular data between instruments and analysis software.
- **Data provenance:** The documented history of how data were collected, processed, transformed, and stored; enables reproduction of results and detection of errors; captured through lab notebooks, metadata files, and Git commit history.
- **File naming convention:** A standardized pattern for naming files that encodes date, instrument, project, and analyst information without spaces or special characters (e.g., YYYY-MM-DD_InstrumentID_ProjectCode_UserInitials_raw.csv); required for all EA Program instrument export files.

- **Instrument export file:** The raw data file generated directly by instrument software (e.g., Shimadzu LabSolutions .lcd or .txt export, Isodat .dxf export, HOBOWare .csv export); must be saved unmodified as the archival primary record.
- **Metadata:** Data describing other data; includes instrument settings, calibration records, sample identifiers, units, processing steps, and analyst notes; stored alongside raw data in a companion metadata file.
- **Raw data:** The unprocessed data as produced by the instrument or analyst at the time of measurement; must never be altered; all subsequent processing produces derived data files that preserve the original.

5. Safety and Hazards

Data export is a computational activity with no physical hazards. When interacting with instruments (connecting USB drives, accessing instrument PCs), follow the safety protocols in the relevant instrument SOP.

6. Equipment and Materials

6.1. Instruments and Export Formats

Instrument	Native Format	Preferred Export	Notes
Shimadzu UV-Vis (UV-1900i)	.spc / .csv	CSV	Export from UVProbe software
Thermo iCAP ICP-OES	.rst / .csv	CSV	Export from Qtegra software
Shimadzu TOC-L	.toa / .csv	CSV	Export from TOC-Control
Dionex ICS Ion Chromatograph	.cdf / .csv	CSV	Export from Chromeleon
Mettler Toledo balance	Direct serial output	CSV via terminal log	Use logging script
YSI ProDSS multiparameter	.csv / .dat	CSV	Download via KorDSS software
Hach DR3900 spectrophotometer	Internal memory	CSV via USB	Export from instrument menu
Bio-Rad QX200 ddPCR	.qlp / .csv	CSV	Export from QuantaSoft
PANalytical XRF	.csv / .xls	CSV	Export from SuperQ

6.2. Transfer Media

Item	Purpose	Notes
USB flash drive (lab)	Sneakernet transfer	Dedicated drive; virus-scanned

Item	Purpose	Notes
Network shared folder	Direct instrument PC transfer	If instrument PC has network access
RStudio Server upload	Upload to analysis environment	Via Files pane or <code>scp</code>

7. Procedure

7.1. General Export Workflow

The same sequence applies to all instruments:

1. **Complete the analytical run** and verify results on the instrument software
2. **Export raw data** from the instrument software to CSV format (or the most open/portable format available)
3. **Name the file** following the convention: `[instrument]_[method]_[YYYY-MM-DD]_[batch].[ext]` (e.g., `icpoes_metals_2026-05-21_batch01.csv`)
4. **Transfer the file** to the project's `data/raw/` directory using USB or network transfer
5. **Verify the transfer** by opening the file and checking row counts, column headers, and a few spot-checked values against the instrument display
6. **Record the export** in the lab notebook: file name, instrument, date, number of samples, and any export settings
7. **Commit to Git** with a descriptive message

Critical Rule

Never modify raw instrument files after export. If corrections are needed, create a new file in `data/processed/` and document the changes in your processing script. The raw file is your permanent record.

7.2. Instrument-Specific Export Instructions

7.2.1. Shimadzu UV-Vis (UV-1900i / UVProbe)

1. In UVProbe, open the data file (`.spc`)
2. File > Save As > CSV
3. Select “Wavelength and Absorbance” columns
4. Save to the USB drive as `uvvis_[method]_[YYYY-MM-DD].csv`

7.2.2. Thermo iCAP ICP-OES (Qtegra)

1. In Qtegra, navigate to the completed sequence
2. Right-click the sequence > Export Results
3. Select CSV format; include sample IDs, element concentrations, and QC flags
4. Save to the USB drive as `icpoes_[method]_[YYYY-MM-DD]_[batch].csv`

7.2.3. Shimadzu TOC-L (TOC-Control)

1. In TOC-Control, open the sample table for the completed run
2. File > Export > CSV
3. Include sample name, injection number, area, concentration, and CV
4. Save as `toc_[method]_[YYYY-MM-DD].csv`

7.2.4. YSI ProDSS Field Sensor

1. Connect the ProDSS to a computer via USB
2. Open KorDSS software
3. Select the data log for the sampling event
4. Export as CSV, including date/time, site ID, and all parameters
5. Save as `ysi_[project]_[YYYY-MM-DD].csv`

7.2.5. Bio-Rad QX200 ddPCR (QuantaSoft)

1. In QuantaSoft, open the completed experiment
2. Options > Export CSV (Well Data or Cluster Data)
3. Include well ID, sample name, target, concentration, positive/negative droplet counts
4. Save as `ddpcr_[target]_[YYYY-MM-DD].csv`
5. Also export the amplitude data for gating verification

7.3. File Transfer to RStudio Server

After exporting to USB or local machine:

```
# Option 1: scp from local machine to RStudio Server
scp icpoes_metals_2026-05-21_batch01.csv \
  username@rstudio.pomona.edu:~/project/data/raw/

# Option 2: Use the RStudio Files pane
# Click Upload > Browse > select file > OK
```

7.4. Automated Import Scripts in R

Create a standardized import script for each instrument type. This ensures consistent parsing across analysts and batches:

```
#' Import ICP-OES data from Qtegra CSV export
# '
# ' @param file_path Path to the exported CSV file
# ' @return A tibble with standardized column names
import_icpoes <- function(file_path) {
  raw <- read_csv(file_path, skip = 2) # Skip Qtegra header rows
```

```
raw %>%
  rename_with(tolower) %>%
  rename_with(~ gsub(" ", "_", .)) %>%
  mutate(
    date_analyzed = as.Date(date_analyzed, format = "%m/%d/%Y"),
    across(where(is.character) & !c(sample_id, sample_type),
           as.numeric)
  ) %>%
  # Flag values below detection
  mutate(across(
    where(is.numeric),
    ~ if_else(. < 0, NA_real_, .)
  ))
}

# Usage
icp_data <- import_icpoes("data/raw/icpoes_metals_2026-05-21_batch01.csv")
glimpse(icp_data)
```

```
#' Import TOC data from TOC-Control CSV export
#'
#' @param file_path Path to the exported CSV file
#' @return A tibble with standardized column names
import_toc <- function(file_path) {
  raw <- read_csv(file_path)

  raw %>%
    rename_with(tolower) %>%
    rename_with(~ gsub("[^a-z0-9]", "_", .)) %>%
    select(sample_name, injection, area, concentration_mg_l = conc,
           cv_percent = cv) %>%
    filter(!is.na(sample_name))
}
```

7.5. Data Integrity Checks

After importing, run basic integrity checks:

```
# --- Row count matches instrument ---
cat(sprintf("Rows in file: %d\n", nrow(icp_data)))
# Compare with number of samples in lab notebook

# --- No fully blank rows ---
stopifnot(!any(rowSums(is.na(icp_data)) == ncol(icp_data)))

# --- Sample IDs match the run sequence ---
```

```
expected_ids <- c("Blank", "Std1", "Std2", "Std3",
                 paste0("S", sprintf("%03d", 1:20)),
                 "CRM", "Dup")
setdiff(expected_ids, icp_data$sample_id)
# Should be empty if all expected samples are present

# --- Calibration standards in expected range ---
icp_data %>%
  filter(grepl("^Std", sample_id)) %>%
  select(sample_id, starts_with("fe_")) %>%
  print()
```

7.6. Archival and Backup

All raw instrument files must be archived in three locations:

Location	When	Responsible
GitHub repository (data/raw/)	Immediately after export	Analyst
Lab instrument PC	Retained on instrument	Lab Manager
External backup drive	End of each semester	Lab Manager

Tip

Large binary files (e.g., raw .spc or .qlp files that exceed GitHub's file size recommendations) should be stored on the external backup drive and the network share, not in Git. Only the exported CSV goes into the repository.

7.7. Instrument Log Cross-Reference

Every data export event must be recorded in the instrument log (see [SOP-L007](#)). The log entry should include:

1. Date and time of the analytical run
2. Analyst name
3. File name(s) exported
4. Number of samples (including QC samples)
5. Any instrument warnings or anomalies noted during the run
6. Cross-reference to the lab notebook page number

8. Quality Control and Acceptance Criteria

- Raw files must be unmodified after export (verify with file hash if needed: `md5sum filename.csv`)

- File names must follow the naming convention in Section 6.1
- Row counts and sample IDs must match the run sequence recorded in the lab notebook
- Calibration data in the export must match the calibration curve approved during the run
- Refer to [SOP-L802](#) for analytical QC criteria (blanks, duplicates, recoveries)

9. Waste Disposal

Not applicable (computational procedure).

10. References

Gandrud, C. (2015). *Reproducible research with R and RStudio* (2nd ed.). CRC Press.

11. Training Record

Training is documented in the EA Program training log. Before exporting instrument data independently, personnel must:

1. Read this SOP in full
2. Complete training on the specific instrument (see the instrument-specific SOP)
3. Demonstrate a complete export, transfer, verification, and commit workflow under supervision
4. Be signed off by the PI or lab manager

12. Appendix A: File Hash Verification

To confirm that a file has not been modified after export, compute and record an MD5 hash:

```
# On Linux / Mac / RStudio Server terminal
md5sum data/raw/icpoes_metals_2026-05-21_batch01.csv
# Record this hash in the lab notebook alongside the file name
```

```
# In R
tools::md5sum("data/raw/icpoes_metals_2026-05-21_batch01.csv")
```

13. Appendix B: Instrument Data Format Quick Reference

Instrument	Header Rows to Skip	Delimiter	Encoding	Date Format
Shimadzu UV-Vis	1	Comma	UTF-8	MM/DD/YYYY

Instrument	Header Rows to Skip	Delimiter	Encoding	Date Format
Thermo iCAP	2	Comma	UTF-8	MM/DD/YYYY HH:MM
Shimadzu TOC-L	0	Comma	Shift-JIS (convert to UTF-8)	YYYY/MM/DD
Dionex ICS	3	Comma	UTF-8	MM/DD/YYYY HH:MM
YSI ProDSS	0	Comma	UTF-8	MM/DD/YYYY HH:MM
Bio-Rad QX200	0	Comma	UTF-8	YYYY-MM-DD
PANalytical XRF	Variable	Tab or comma	UTF-8	DD/MM/YYYY
