Data Coordinating Center
Biostatistics Working Guidelines

Contents

Abbreviations

Abbreviation	Definition
CDM	Clinical Data Manager
CONSORT	Consolidated Standards of Reporting Trials
DCC	Data Coordinating Center
DSMB	Data and Safety Monitoring Board
(e)CRF	(Electronic) Case Report Form
EDC	Electronic Database Capture
IRB	Institutional Review Board
MAP	Manuscript Analysis Plan
MARF	Manuscript Analysis Request Form
MOO	Manual of Operations
PHI	Protected Health Information
PI	Principal Investigator
PM	Project Manager
PUD	PUD
QC	Quality Control
RCT	Randomized Controlled Trial
SAP	Statistical Analysis Plan
TLFs	Tables, Listings, and Figures
WG	Working Guideline

Introduction

Purpose

The Biostatistics Working Guidelines (WGs) outline the common requirements, processes, and practices for providing biostatistical support to Data Coordinating Center (DCC) studies. As the name suggests, these are to be taken as general guidelines for how we accomplish our work. These guidelines were developed using the cumulative expertise of many biostatisticians and represent our current understanding of best practices for ensuring quality and consistency across projects. As we continue to develop as a biostatistics team, we will update and modify these guidelines to reflect new insights and understanding.

Biostatistician Roles and Responsibilities

The responsibilities each biostatistician on a project takes on are somewhat dynamic based on the project complexity, risk level, specific tasks required and relevant experience of each biostatistician involved, and practical considerations (e.g., funding and available personnel time).

Tasks may be delegated when appropriate given the expertise of the individual taking over and the risk level of the project. For example, for most randomized controlled trials (RCTs), the faculty biostatistician would write the statistical analysis plan (SAP) and be very involved in the primary analyses. For a lower risk project, such as descriptive analyses of pilot data, the faculty biostatistician may ask the lead to take on the tasks of both writing the SAP and performing the primary analyses and simply oversee these efforts.

Faculty Biostatistician

The faculty biostatistician is the primary contact for study design and analysis issues and authors (or is otherwise responsible for) content in the statistical sections of the protocol and the SAP. The faculty biostatistician mentors staff biostatisticians in analysis methods and oversees development and execution of analyses and the interpretation of results.

In terms of specific tasks and activities, the faculty biostatistician generally is responsible for the following, in consultation with study investigators and other members of the statistical team as needed:

- Participates in study team and statistical team meetings.
- Interacts with clinical investigators in developing and justifying details of the final study design and planned analyses.
- Provides a comprehensive review of the protocol with primary responsibility for content related to study design, hypotheses, planned statistical analyses, and power calculations.
- Writes the SAP, including detailed definitions of study endpoints, analysis populations, and planned analyses with methods for implementation.
- Reviews database content with particular attention to capture of primary, secondary, and safety endpoints and ability to follow Consolidated Standards of Reporting Trials (CONSORT) guidelines; is responsible for any formal sign off of database, if applicable.

- Oversees development and execution of analyses and the interpretation of results for all study deliverables; may take lead on analyses and written summary of results for primary manuscript(s).

Faculty involvement and oversight is especially critical for high risk studies, including RCTs. The following additional faculty responsibilities are relevant for RCTs:

- Creates the randomization plan/schedule and oversees implementation.
- Writes or is otherwise responsible for content of the Data and Safety Monitoring Board (DSMB) charter.
- Performs or oversees key analyses (safety and efficacy monitoring, futility) for the DSMB and presents results of DSMB open and closed reports during scheduled meetings.

Lead Biostatistician

The lead staff biostatistician is a key participant in the DCC study team from protocol development to final analyses, interacting with the faculty, dual and supervising biostatistician on all aspects of the project. This individual provides dataset programming and analysis support for the study, and is the main point person within the statistical team for most issues.

Given the scope of work required for the lead biostatistician, the responsibilities are outlined below by prior to and following enrollment start.

Prior to enrollment start

- Participates in study team meetings.
- Provides detailed review of key study documents including the protocol and SAP.
- Is a key contributor in database planning; provides a comprehensive review of the data elements.
- Reviews database structure, layout, and organization of forms
- Reviews variable names, labels and formats created within the database
- Participates in test data entry.
- Writes code to read in study data and create raw SAS datasets.

- Reviews test data.
- Provides detailed input on and review of edit specifications.
- Set up statistical analysis folder structure on shared drive.

During/after enrollment start

- Participates in study and statistical team meetings.
- Writes the manuscript analysis plan(s) (MAP).
- Develops mock content for reports and manuscripts.
- Creates dataset specifications for derived analysis data.
- Programs derived analysis datasets.
- Reviews data for discrepancies and potential queries.
- Implements planned analyses and verifies output.
- Participates in manuscript preparation.
- Ensures the dual biostatistician has the needed materials to perform dual programming and code review and follows up on any discrepant variables or results from dual programming until agreement is achieved.
- Ensures all materials are archived following completion of key deliverables.
- Consults with DCC study team related to preparation of a public use dataset.

The lead biostatistician utilizes the expertise and support of other team members to accomplish tasks in an effective and efficient way. In particular, the supervising and lead biostatisticians collaborate closely to ensure that the highest standard of work is being provided for a project.

Dual (QC) biostatistician

The dual biostatistician is responsible for any quality control (QC) activities related to the project. This individual provides dual programming and code review for the project as needed and contributes to statistical team discussions and decisions. It is also important to note that the roles of lead and dual may switch for some aspects of project support, particularly for projects with multiple deliverables. There may also be some tasks normally completed by the lead that are delegated to the dual biostatistician, in consultation with the supervising biostatistician. And, in some cases, the supervising biostatistician may step into the dual role on selected aspects of a project.

The dual biostatistician

- Participates in statistical team meetings.
- Reviews study materials including final protocol, SAP, and data elements collected.
- Provides detailed review of analysis dataset specifications for derived analysis datasets.
- Dual programs derived analysis datasets.
- Performs QC on project deliverables based on the risk level of the project/deliverable.

The dual biostatistician is much more involved in statistical team decisions and deliverables for high risk studies (e.g., RCTs) than for other studies. In these studies, early involvement of the dual biostatistician and an in-depth familiarity with the study protocol, database, SAP, and other study materials is essential.

Supervising biostatistician

The supervising biostatistician's primary role is to provide structure and support for a project (or set of projects/network). Specifically, this individual ensures that essential activities are coordinated and completed and that DCC standards are followed. The supervising biostatistician serves as a resource for the rest of the statistical team and provides oversight, mentoring, and support. For related projects or a network, a supervising biostatistician provides a unifying perspective and approach across the various projects. The extent of the supervising biostatistician's involvement may be more or less than is described below depending on the project complexity, risk level, and/or the experience of the DCC study team supporting the project. If a specific supervising biostatistician is not assigned, the lead biostatistician's direct manager fulfills the basic requirements of this role as needed.

When beginning a new project, the supervising biostatistician meets with the lead and dual biostatisticians to review past work experience, relevant technical expertise, and familiarity with DCC processes. For newer hires, it is also important to coordinate with the assigned mentor to ensure adequate training and orientation for any needed tasks. Throughout the project, the supervising biostatistician oversees and supports implementation of assigned tasks, and facilitates understanding of how each task fits into the overall study support and research process. The supervising biostatistician may need to engage at a more detailed level for certain tasks to ensure the highest standard of support. A suggested starting point for division of responsibilities is outlined in the table at the end of this introduction.

The supervising biostatistician

- Determines risk level of the project and project deliverables for QC purposes.
- Mentors, trains, and oversees lead and dual biostatisticians in their responsibilities.
- Participates directly in project support activities including review for substantive issues in the protocol, SAP, database, analysis dataset specifications, mock content, and planned analyses.
- Leads statistical team meetings and establishes agenda based on input from team; summarizes meeting decisions and action items.
- Manages project tasks and timelines and ensures accountability.
- Coordinates completion of activities for key deliverables.
- Ensures data center standards are followed and that adequate dual programming and verification is completed for analysis datasets and project deliverables.
- Manages project effort and identifies resources for additional statistical support in consultation with Biostatistics Director and/or Biostatistics Manager(s).
- Oversees project orientation, training, and timeline adjustments related to staffing changes.
- Creates documentation of key study processes related to the statistical support of a project.

The supervising biostatistician role is meant to be flexible to fit the circumstances of the project. A supervising biostatistician leverages their critical thinking and leadership skills to build tools/processes and establish statistical team dynamics that will best meet the unique needs of each project. As a core member of the DCC study team, the supervising biostatistician is familiar with details of the project. In the case of unexpected events that may threaten successful support of the study, the supervising biostatistician engages as needed to help get any essential work done.

As part of the mentoring process and to foster leadership and organizational skills, the supervising biostatistician is encouraged to identify responsibilities that can be transitioned to the lead after the major “front work” of a project is completed. The timing and extent of this transition depends on the project complexity and risk level and is determined in consultation with the faculty biostatistician. The supervising biostatistician ensures adequate training for the lead in any new responsibilities and may continue to participate in relevant activities for a time in order to mentor the lead in their new role(s).

Prior to enrollment				
Activity	**Faculty**	**Supervising**	**Lead**	**Dual**
Protocol and SAP development	Responsible	Contribute Mentor	Contribute	Familiar with final content
Randomization scheme/ implementation	Responsible	Support as requested	Support as requested	Not required
Database content review	Responsible	Lead Mentor	Contribute	Familiar with final content
Data extract testing (raw datasets)	Not required	Mentor Oversee	Lead	QC
Edit specifications review	Contribute	Mentor Oversee	Lead	Not required
Risk assessment	Contribute	Contribute Mentor	Contribute	Not required
During/after enrollment				
Activity	**Faculty**	**Supervising**	**Lead**	**Dual**
Determine risk level and QC requirements	Consult	Lead	Support	Support
MAPs	Mentor Oversee	Consult	Lead	QC
Analysis dataset specifications	Consult	Contribute Mentor	Lead	QC
Program analysis datasets	Consult	Mentor Oversee	Lead	QC
Statistical data monitoring/review	Consult	Mentor Oversee	Lead	QC
Mock Tables, Listings, and Figures (TLFs) for key deliverables	Review	Contribute Mentor	Lead	QC
Programming of key deliverables	Review Present	Mentor Oversee	Lead	QC

Study Startup

Study Startup

The startup phase of a study is a crucial period where study documentation is developed, Electronic Database Capture (EDC) system(s) is/are built, and analyses are planned. Use the DCC Pre-Enrollment Checklist to guide you through each step of this process. This WG focuses on a few specific areas of the Pre-Enrollment Checklist that require more attention. Note that not all of items in this list will necessarily be completed prior to enrollment.

Study and protocol development

Protocol changes are often difficult and lengthy processes because they must be approved by the Institutional Review Board (IRB). A thorough review of the protocol will minimize protocol changes after IRB approval. The lead biostatistician pays particular attention to the following items in the protocol:

- Specific aims
- Sample size and power calculations
- Outcomes
- Planned analyses
- Data elements (including scheduling/timing of data collection visits)
- Randomization (if applicable)

Provide feedback to the Project Manager (PM) and discuss potential issues with the study team so that any necessary changes can be incorporated.

SAP

The SAP provides detailed descriptions of the objectives, outcomes, design, variables, and analyses needed to complete the study. It will include details on the analyses for the primary manuscript(s) as well as the DSMB report (if applicable). During your review of the SAP, carefully check

- Study objectives and endpoints
- Study design

- Sample size and power calculations
- Subject and analysis population definitions
- Schedule of planned analyses
- Methods for completing analyses
- Variable definitions
- Randomization information (if applicable)

Ensure that the SAP is consistent with the protocol and provides sufficient detail to derive variables and perform all analyses for the study objectives.

DSMB materials

Studies with a DSMB require additional items, including a DSMB charter and DSMB report mock content.

DSMB Charter

The DSMB charter specifies

- The composition and responsibilities of the DSMB
- Frequency of meetings
- Specific outcomes to be monitored
- Technicalities of efficacy monitoring boundaries

The sponsor or the faculty biostatistician will write this document for the study. Carefully review the charter to ensure that all of the applicable above items are included.

DSMB Report Mock Content

Build the mock content for the report (see WG Generating Mock Content).

Data collection and validation

Data Element Review / Database Design

Review the data elements in detail to ensure elements necessary for planned analyses are included (see WG Data Element Review).

EDC and Other Data Sources

Review and test the EDC to ensure we can create raw SAS datasets for all study data (see WG Overview Raw Dataset Testing and Creation and WG External Data Sources).

Worksheets

Review data collection worksheets to ensure consistency with the EDC forms.

Data validation

The Clinical Data Manager (CDM) validates study data by writing rules. These rules check logic between variables, missingness, ranges, etc. The Edit Specifications contain all rules implemented in the study. Focus your review of the Edit Specifications on variables and forms needed for analyses, using the questions outlined in the Pre-Enrollment Checklist as a guide.

Study documents and training materials

Manual of Operations (MOO) Review

The MOO increase protocol adherence and consistency in data collection by outlining the process by which the study will be carried out at each clinical site. During your review of the MOO, ensure the accuracy and clarity of

- The Study Management section (study workflow and data management)
- Workflow and data entry guidelines (to ensure logic and consistency with the protocol)
- The randomization process (if applicable)
- Other items that may affect the interpretation of study data

Address any issues you find with the PM and study team.

Creating the Statistical Analysis folder

Create the Statistical Analysis folder (see WG Standard Folder Structure) on the P: drive. Restrict access to the Statistical Analysis folder from all non-biostatistician study members for any study with blinded or confidential results.

References

- Pre-Enrollment Checklist

Data Element Review

Preparing for data element review

As part of study startup procedures, review available study materials in preparation for data element review. Materials that you review may include:

- Study protocol
- SAP
- Manuscript Analysis Request Forms (MARFs)
- Mock DSMB report
- Other materials provided by the study investigators (e.g., example worksheets, materials from previous studies, related publications)

Identify variables needed to perform planned analyses. Consider primary, secondary, and safety analyses, as well as study flow diagrams, adverse event reporting, etc.

Data element review

Data elements may be provided in a list, a set of preliminary worksheets, or some other format. Regardless of format, do the following:

- Identify all data sources. The source of each data element needs to be clear.
- Ensure all elements/variables identified in the review of study materials are included in the list of data elements.
- Make sure all of the elements needed for derived variables are included. Walk through the derivation of variables to ensure all necessary information is collected.
- Carefully review question wording and skip logic (if provided at this stage) to ensure the data will meet the needs of the study. Note, if question wording and/or skip logic is not provided at this stage, you will need to review it before it is implemented.

Provide feedback to the PM and discuss potential issues with the study team so that any necessary changes can be incorporated.

Raw Data

Overview of Raw Dataset Creation and Testing

Review EDC materials

Before the EDC is moved into production, do the following:

- Ensure the EDC reflects all aspects of the study worksheets (e.g., same question wording, skip logic, choice sets).
- Review the documents used to create the database (e.g., OpenClinica CRF Design spreadsheets or the RedCap data dictionary). Verify that variable names and all other attributes are valid in SAS and accurately and clearly reflect what is being collected. See EDC specific tools and references for database testing for more details.

Enter test data

After feedback from the review of EDC materials has been incorporated into the database design, enter test data directly into the EDC and do the following:

- Enter data on all forms.
- Test skip logic and all branches resulting from skip logic.
- Confirm that data entry fields only accept the correct data type and precision.
- Enter extreme and irregular values to test that the EDC behaves as expected.
- Ensure variables have been assigned as required or not required, as appropriate for data entry.
- Continue review of study data elements (see WG Data Element Review).

Test importing of data into SAS

After test data are entered, create raw SAS datasets using the test data combined with metadata extracted from the EDC source. See the references section for EDC-specific instructions for creating raw datasets. Write a program to import the raw data into SAS and verify the following:

- Data types, variable names, formats, labels, values, and lengths are imported correctly (view PROC CONTENTS output and open each dataset to visually inspect).

- Data are read into SAS without errors or loss of data (review of log and visually inspect each dataset).
- All forms and repeating question groups have a corresponding dataset.
- Datasets can be combined when appropriate (examine site, subject, and any within-subject level identifiers; write SAS code to test necessary merges and/or joins).
- Primary outcomes and other important or complex variables can be derived using the raw data (writing SAS code if necessary).

If applicable, also test the import of randomization data and other data sources external to the EDC (see WG External Data Sources).

Communicate issues to the CDM and repeat testing as needed until all issues are resolved.

References

- EDC specific tools and references for database testing

External Data Sources

In some cases, data must be collected from an external data source (i.e., a source that does not directly link to the SQL warehouse). Some examples are data from a randomization service, results from an outside laboratory's analysis of study specimens, and ad hoc categorizations of open text from investigators.

Test external data to the extent possible. Steps may include:

- Entering test data.
- Reviewing a sample of the data.
- Ensuring SAS datasets can be created without loss of variables, observations, or values.
- Ensuring appropriate identifiers are available and that the external data can be merged with other study data.
- Verifying datasets contain all necessary variables (e.g., randomization data may need to include treatment assigned in a blinded trial).
- Writing study-wide and manuscript-specific data checks (see WG Data Checks).

Plan for incorporating external study data early. Discuss external data sources with the study team during study startup. Work with the CDM to incorporate external data into the SQL warehouse.

In some cases, investigators will provide external data directly (e.g., an investigator makes categories based on text field entries in a spreadsheet). The study biostatistics team may decide to import that type of external data directly into SAS rather than have it added to the SQL warehouse. Import these files according to (see WG Creating Production Raw Data). Save and archive the external data files (e.g., .xlsx or .csv files) and SAS datasets with study or manuscript raw datasets (e.g., in the data/raw folder or subfolder). Update the RAW tab in the dataset specifications spreadsheet as described in (see WG Analysis Dataset Specifications).

Alternatively, the study biostatistics team may choose not to import the external data file, but to use it as a reference (e.g., an investigator makes categories based on text field entries, and you can write an algorithm to define the categories without importing the spreadsheet). In this case, treat the data file as an external reference (see WG Writing Variable Algorithms). Save and archive the file with study or manuscript documents.

When external data are complex or collecting data involves coordinating data entry from multiple sources, work with clinical data management and project management on the coordination, collection, and validation of data. Complete the steps listed above to test the external data. If you are asked to coordinate data collection, discuss with your manager and/or supervising biostatistician.

Creating Production Raw Data

Testing production raw data

After the data collection forms have been moved to the production instance of the EDC, testing is repeated. The same steps are followed as in initial testing and are completed before any subjects are enrolled in the study (see WG Overview of Raw Dataset Creation and Testing).

Creating production raw SAS datasets

Write a SAS program to create production raw SAS datasets by either creating a new program or modifying a copy of the program used to create test raw SAS datasets (see Study Startup repository in BitBucket). This new program is separate from the testing program; the testing program is retained in case future testing is needed. Write the production raw data program to accomplish the following:

- Pull raw data from the production instance of the SQL warehouse or other production source (e.g., REDCap SAS files, OpenClinica ODM file).
- Remove test records from all datasets. NOTE: You may delay the deletion of test records if data are needed to create and examine derived datasets or for initial work on analysis. However, no results are shared unless all test records have been removed from the datasets used.
- Read raw datasets into SAS libraries. If you study has multiple data sources (e.g., multiple EDCs or multiple studies within the same EDC), you may create a separate raw SAS library for each data source, consulting with the study biostatistics team if needed. If more than one SAS library will be used, write a separate SAS program to create raw datasets for each SAS library. Write an additional program to run all raw data programs if desired.

Do not modify the raw datasets beyond deleting test records or dropping useless variables. If any modifications beyond these are needed, consult with the study biostatistics team.

The program(s) that create the production raw datasets undergo QC prior to any final deliverables (e.g., DSMB reports, manuscripts). Include the program(s) as part of the QC for the deliverable (see WG Overview of QC Process, WG Determining Risk Level and QC Requirements, and WG Preparing for and Requesting Code Review).

Database Changes

When changes to a database are required (e.g., new forms/variables are added, or changes are made to existing forms/variables), work with the study team to ensure the modifications are necessary and address the problem prompting the change. For example, if the change is to question wording, verify that the new wording resolves the issue. If the change is to a choice set, verify that the new choice set accomplishes the purpose of the database change.

Fully test the database changes in a non-production environment (e.g., REDCap Test, OpenClinica Staging) before the CDM moves the changes to production. Complete the following steps:

1. Work with the CDM to agree on a timeline that allows for testing to be completed prior to moving the updates to production.

2. Complete the testing process as done during initial database testing (see WG Overview of Raw Dataset Creation and Testing), but focused on the changes. Perform the steps in review and testing in the correct order (e.g., review appropriateness of SAS variable names, labels, and formats before testing the creation of raw datasets).

3. Work with the CDM to resolve any issues identified in your review before they push the changes into the production instance of the study database.

4. Once the changes are pushed into production, repeat testing to ensure no issues arise.

Analysis Datasets

Planning Study Analysis Datasets

Thoughtful planning of analysis datasets is crucial for efficient analyses. Biostatisticians use study analysis datasets for general study activities (e.g., reports, including DSMB reports) and as a starting point for manuscript analysis datasets (see WG Analysis Datasets for Manuscript Variables).

Early in the study, plan the study analysis datasets (number of datasets, dataset order and dependencies, subject level vs not, etc.) with input from the study biostatistics team as needed. Below are steps to accomplish this:

1. **Find and review study materials.** Focus the review on data required to accomplish the needs of the study. Study materials include, but are not limited to:
   - Study protocol or similar (required)
   - SAP (if available)
   - MAPs (if available)
   - Mock DSMB content (if available)
   - Study reports or data requests (if available)
   - Scoring documents or manuals for standard assessments (if applicable)
2. **Create a list of variables.**
   - List variables in the protocol and SAP, such as analysis populations, outcomes covariates, and subgroups.
   - Include all variables required for DSMB report.
   - Include all variables used in manuscripts addressing the protocol aims of a study (i.e., the main manuscripts).
   - Include variables defined in any MAPs that are available.
   - Include study and site performance measures. Common measures are screening and enrollment rates (screened, eligible, approached, consented, randomized), follow-up rates, sample collection rate, protocol adherence rates, and measures of treatment compliance.
   - Include intermediate variables to simplify complex, mult-step algorithms.

   NOTE: Do not include personal identifying information (e.g., contact information, birth date) unless it is necessary for the analysis.
3. **Create a list of datasets and group variables into datasets.**

- Organize datasets according to variable grouping that are consistent with planned analyses.
- Give each dataset a name that describes its contents.
- Give each variable a descriptive name.
- Group screening and enrollment variables in a dataset of all screened subjects. This screening dataset can be used for enrollment reports and to create a Consolidated Standards of Reporting Trials (CONSORT) flow diagram. All other datasets will contain all or a subset of these subjects.
- Plan a primary dataset that will contain key variables for the enrolled population. Include demographic and baseline characteristics, subgroups, outcomes, treatment assignment (blinded if necessary), and analysis population variables. This dataset will likely be used for the primary analysis.
- Arrange relational data in separate datasets (e.g., adverse events, laboratory results, outcomes measured over time). Include summarized data from these sources in the primary or other subject-level datasets as needed.
- Use intermediate datasets to simplify complex, multi-step algorithms.
- Consider analyses when grouping variables so that you can perform desired analyses without additional data manipulation. For example, creation of a screening report would not require joining multiple datasets.
- Do not include personal identifying information (e.g., contact information, birth date) unless it is directly used in the analysis.
- Include variables in more than one dataset as needed. Identifiers, demographics, analysis population identifiers, treatments and subgroups are some common examples of variables that you may want in multiple datasets.
- Some analyses may not require data derivation; you do not need to create a derived dataset when only raw variables will be used.
- Not all raw variables must be included in derived datasets.

4. **Determine the population, structure, and order of creation of the datasets.**

    - Determine which subjects you will include in each dataset and how to define that population.
    - Decide how many rows per subject/event/etc. each dataset will have.
    - Determine dependencies each dataset has. For example, does the adverseEvent dataset depend on enrolledSubjects, or does enrolledSubjects depend on the adverseEvent dataset?
    - Plan the order of dataset derivation.

- Start with the screenedSubjects dataset, or some other dataset that includes all subjects.
- Plan to derive intermediate datasets next. Intermediate datasets may include:
  - Datasets with multiple rows per subject. Include copies of some variables from screenedSubjects (or similar) in these (e.g., identifiers), but avoid deriving subject level variables in datasets with multiple rows per subject. Derived variables in subject level datasets can be created later on by summarizing variables from datasets with multiple rows per subject.
  - Datasets that only apply to a subset of subjects or events.
- End with primary datasets, including the dataset containing key variables for the enrolled population.
- Do not reference a derived variable before creating it.
- Do not derive the same variable more than once; you can easily copy the variable to other datasets.

- Consider changes or additions that may be made in the future. Plan your datasets so that you will be able to add new datasets and new variables without restructuring the existing ones.

5. **Review the planned study analysis datasets and variables with the study biostatistics team.** Do this before you begin programming. Outline the structure of the datasets and the lists of variables in a Word document, the analysis dataset specifications template, or in another convenient format for this review. After the study biostatistics team has provided input, move onto programming and documenting the study analysis datasets in the specifications (see WG Analysis Dataset Specifications).

Updating the structure of study analysis datasets

Changes and additions to the study analysis datasets will be needed as a study progresses.

- Follow the principles listed above as you add and modify variables and datasets.
- Discuss modifications to the dataset structure with the study biostatistics team as needed.
- As the study progresses, add or modify variables and datasets that are required to produce any report or data summary, including those requested by the study team, investigators, funding institution, or a DSMB.

- Add variables and datasets as needed when new MAPs are written.
- After the raw datasets are locked and any manuscripts addressing the protocol aims of a study are accepted, the study biostatistics team may opt to cease modifying the study analysis datasets, and instead derive new variables in manuscript analysis datasets. This is not required and is determined on a case-by-case basis.

Analysis Dataset Specifications

Analysis dataset specifications clearly define all datasets and variables to be derived for a study, manuscript, or DSMB report. The specifications also provide brief descriptions of the raw data sources used for the study. Building the analysis dataset specifications begins as soon as possible, and is done in tandem with programming the analysis datasets.

Use the most recent version of the Analysis Dataset Specifications Template in order to ensure that all necessary sections and features are available. See the references section for a link to this template, as well as an example set of analysis dataset specifications.

Contents

The following information outlines and describes the major content included in the specifications. The tabs are generally filled out in the order shown below.

Overview Tab

The Overview tab provides general information about datasets used throughout the analysis dataset specifications:

- **SAS library**: The name assigned to the dataset library in SAS (e.g., “raw”, “der”).
- **Description**: A brief description of the library (e.g., “data from OpenClinica (EntrySite database) with formats and labels applied”).
- **Location**: The folder location of the library on the P: drive (e.g., P:\PECARN\TXA\Statistical Analysis\Data\Raw). Note that this section is filled out before building any analysis datasets. At a minimum, include a row for the raw, derived lead, and derived dual dataset libraries. Add additional rows as necessary for any other SAS libraries that are needed to create the analysis datasets.

SAS library tabs

The specific SAS library tabs provide information about the contents of all SAS libraries referenced throughout the specifications. There must be one tab for each SAS library listed in the Overview tab. Details for the most common tabs (RAW and DER) are provided below. The format for any additional tabs matches either RAW or DER depending on whether the library contains raw or derived datasets.

RAW tab

The RAW tab provides information about all datasets in the RAW library:

- **SAS dataset**: The name of the dataset as defined in SAS (e.g., "Demographics").
- **Description/Form Name**: If applicable, enter the name of the EDC form in which this dataset's variables are captured (e.g., "Demographics Information - v1.0"). Otherwise, enter a brief description of the contents of the dataset (e.g., "Subject demographics").
- **Unique identifier**: The variable(s) that uniquely identify each record in the dataset.
- **SQL table name**: If different from the SAS dataset name, enter the name of the dataset as it appears in the SQL warehouse.
- **SQL server instance**: The instance of the SQL server where the dataset is stored (typically "DCCDBStudies.iicrc.med.utah.edu").
- **Data handling used**: Any special data handling completed when creating raw datasets.

This section is completed before work on the analysis datasets begins.

DER tab

The DER tab provides information about all datasets in the DER library:

- SAS dataset
- Brief description
- Population
- Dataset structure
- Unique identifier

The Excel template is designed to pull the above information from the headers of the individual dataset tabs once you enter the SAS dataset, because the information is duplicated within each tab. Fill in this tab with the SAS dataset when you begin work on the first analysis dataset, and add subsequent analysis datasets as they are built.

[DATASET NAME] tabs

An individual tab for each dataset is created to provide detailed information on the population and variables in that dataset. The tab is given the name of the dataset, and the following information outlined within the tab:

- **Dataset general information (i.e., tab header)**
  - **Dataset Specifications**: After you rename the tab with the dataset name, the first row of the tab will be automatically populated to show the dataset name (e.g., "Screen Specifications").
  - **Brief description**: Briefly summarize the content of the dataset (e.g., "adverse event information", "main subject level information").
  - **Population**: Criteria used to define the population to be included in the dataset (e.g., "randomized subjects who had an adverse event", "all subjects screened for participation in the study", etc.).
  - **Dataset structure**: Level of detail represented by individual records in the dataset (e.g., "one record per subject per adverse event", "one record per subject per study per day per lab panel", "one record per subject", etc.).
  - **Unique identifier**: Variable(s) used to uniquely identify each record or observation. This parallels the dataset structure (e.g., "StudySubjectID and AENum" when the structure is "one record per subject per adverse event", "StudySubjectID" when the structure is "one record per subject", etc.).
  - **Notes/special handling**: List notes/special handling affecting multiple variables or entire records that are not found in another section (e.g., explanation of conflicting/missing data for multiple variables/subjects, macro variables to be merged into other datasets, and screening cutoff dates for DSMB report).
- Analysis variable information (i.e., spreadsheet columns the describe the variables)
  - **Variable #**: An optional number used for logical ordering or organization of variables in the dataset.
  - **Variable name**: SAS variable name in the dataset.
  - **Label/Description**: The label assigned to the variable in SAS. Create labels that are clear and accurately describe the variables; also include units when appropriate, e.g., "Day 1 Hemoglobin (mg/dL)".

  - **Format/Values**: The format applied to the variable in SAS. Include the format name and, if applicable, unformatted and formatted values. If the format is new (i.e., not found in the raw data), indicate this by listing "(DER)" or "(NEW)" after the format name to identify format that need to be dual programmed. Clearly indicate variable types (e.g., "numeric", "character") unless they are clear from the format. Here is an example format:

    ACTIVE.
    (DER)
    1 = Active
    2 = Inactive

- **Dataset(s) used**: List the specific input datasets needed to derive the variable in this field (e.g., "Raw.Demog", "der.Screen").

- **Algorithm**: List the algorithm for deriving the variable in this field. See WG Writing Variable Algorithms for requirements and guidelines on writing variable algorithms.

- **Notes and special handling**: Provide additional information and considerations for variable derivation (e.g., hard coding reasons) for a particular variable when needed.

- **Edits and comments**: This column is used for communication between the biostatisticians as datasets are being worked on. Use this field to mention specific changes or to provide feedback on variable definitions.

Maintenance of analysis dataset specifications

Make any edits to the analysis dataset specifications (e.g., new variables, algorithm/ label/format changes, header changes, etc.) clear and easy to identify so that the dual biostatistician can easily review and implement the changes. The following will facilitate collaboration and facilitate communicating edits:

1. Allow multiple users to edit the Excel analysis dataset specifications document by selecting "Review" → "Share Workbook" → "Allow changes by more than one user at the same time".

2. Include when an edit was made, who made it, and what was changed.

3. Use text color, strikeout, and highlighting to track specific edits without losing the original specification; these edits are made directly in the field that requires updating.

4. Have the dual biostatistician review the changes and provide feedback, if needed. Once the feedback has been implemented, the comments and special formatting to show edits are removed (unless the feedback helps in resolving discrepancies found while dual programming).

Keep the dataset specifications consistent with the current state of the analysis datasets. When you modify the analysis datasets, make the same modifications to the specifications. Archive specifications associated with major deliverables (see WG Archiving).

References

- Analysis Dataset Specifications Template
- Analysis Dataset Specifications Example
- Best Practices for Dataset and Variable Naming

Writing Variable Algorithms

For each variable in an analysis dataset, the information in the algorithm column in the analysis dataset specifications (see WG Analysis Dataset Specifications) is specific enough to allow someone with only the database documentation, dataset specifications, and SAP/MAP to reproduce the variable. This WG outlines general directions for writing variable algorithms.

Algorithm Guidelines

- There are multiple options for expressing algorithms:
  - Pseudo code (variable names and logical expressions):

    1 if Age $<$ 0.5 years
    0 otherwise
  - English prose (sentences):

    Count the number of records in der.Biospecimen that exist for the participant. If the participant has no records in der.Biospecimen, set to zero.
  - Combination of pseudo code and English prose:

    If Main.BaseNeuroEpYN = "Yes", then difference in years between Birthdate and Main.FirstNeuroEpisodeDate
    Missing otherwise

  Choose the option that makes the algorithm most readable and easiest to follow. Write the algorithm in a way that parallels the definition in the source (e.g., definition in the SAP). This will allow the dual biostatistician to verify the algorithm is defined correctly in the analysis dataset specifications.

- Do not include explicit SAS code. It is acceptable to indicate which SAS function and necessary options to use.

- Use formatted values instead of coded values for all variables except the current variable being defined. For instance, to define the variable *randomized*, use

  randomized = 1 when consent is "yes" and rand is "yes"
  rather than
  randomized = 1 when consent is 1 and rand is 1

- If applicable, include external references for algorithms (e.g., survey scoring documentation). Some algorithms require additional documentation as a supplement in order to organize essential information from references and include sufficient detail to program the variable. If the algorithm has been completely written out in the specifications, add a note about the reference in the notes and special handling column. If the algorithm is not completely written out in the specifications, but the derivation is fully explained in the additional documentation, include a link to the reference or a note indicating the location of the reference in the algorithm column rather than in the notes and special handling column. The reference is reviewed in tandem with the algorithm by the dual biostatistician to confirm it has been written correctly.

- Any applicable hard coding instructions (i.e., code to modify a variable's value for a specific subject) are provided in the algorithm column. The justification for the hard coding is provided in the algorithm or notes/special handling column of the analysis dataset specifications WG Analysis Dataset Specifications).
- If a variable is being retained in the analysis dataset as an exact copy from a raw dataset, use the word "copy" in place of an algorithm.
- If the only change being made in the derived dataset to a raw variable is a label or format change, then the new derived variable can have the same name as the original raw variable. Write the algorithm to reflect this minor change (e.g., "copy, but change format"). If more than the label and format are changed, assign a new variable name.
- If the only change being made is a new variable name being assigned, write the algorithm as, "copy of xyz".
- If more than one dataset is needed to derive the variable, reference variable names preceded by the respective dataset name (e.g., use Eligibility.ScreenDate when ScreenDate is taken from the Eligibility dataset).
- Explicitly define which cases are defined missing within the algorithm so it is clear which cases are expected to have missing values.
- If the algorithm for a new variable is long or complex, break it up by defining intermediate variables that together make the final variable easier to define.

Creating Analysis Datasets

Programming analysis datasets can be time consuming. As a result, adequate planning and development of a timeline are needed to meet study deliverable deadlines. Begin programming the analysis datasets as soon as possible, in tandem with creating the analysis dataset specifications (see WG Analysis Dataset Specifications). Use data from test subjects to begin programming the analysis datasets when real subject data are limited or unavailable (e.g., prior to enrollment, before follow up data are available).

Program the analysis datasets in SAS and keep code well organized, readable, and commented (see WG General Programming Standards and WG SAS Specific Programming Standards). To keep code organized, maintain a separate SAS program for each analysis dataset. While an analysis dataset program may reference other programs, it only creates a single derived dataset. While programming, it is important to delete datasets in the work directory often, especially before or after creating each analysis dataset. This is done so that errors will not be masked or created by reuse of temporary dataset names. The code below is an example of how to clear datasets in the work directory:

```
proc datasets library = work kill;
quit;
```

Store derived format datasets and catalogs in the same directory as the analysis datasets (see WG Standard Folder Structure). Create a single program, separate from any analysis dataset, containing the code to create the derived format dataset and catalog. The derived format dataset and catalog only contain formats that are not direct copies of those contained in the raw format dataset. The following code is an example of how to create the format dataset and catalog:

```
proc format lib = der cntlout = der.formats;
value NewFormat
1 = 'Format 1'
2 = 'Format 2'
;
run;
```

Include code in your formats program to delete the formats dataset and catalog before the formats are re-created. This is done so that only the formats needed are included in the formats catalog. The following code is an example of deleting the format dataset and catalog:

```
proc datasets library = der memtype = catalog nolist;
```

```
delete formats;
quit;

proc datasets library = der memtype = data nolist;
delete formats;
quit;
```

Data Checks

Data checks are used to identify raw data issues that could affect analyses and to provide feedback to the study team. Identifying issues in analysis variables and tracing the issue back to raw variable(s) is a helpful way to find raw data issues. Write data checks early and continue adding data checks throughout the study.

Review the SAP/MAP(s) and other study documentation to find important analysis variables. Next, identify raw variables that are needed to derive these important analysis variables. Develop a list of potential issues with the raw and analysis variables you have identified. This is your initial list of data checks. You will identify additional data checks as you create the analysis datasets, and review the analysis variables for data issues.

Below is a list of suggested data checks and items to review:

- Missingness for variables and forms
  - expected forms exist
  - data in repeating groups are consistent with lead-in question
  - sub-questions are consistent with the main questions
- Cross tabulations to verify data consistency
  - variables are consistent with other similar, dependent or associated variables (e.g., randomized subjects are also eligible subjects)
  - analysis variables are consistent with alternative definitions (when raw data are collected in multiple places)
- Date/time checks
  - range of time intervals compared to expected range
  - event timelines (e.g., ED triage date $\leq$ ED discharge date $\leq$ death date)
  - other logical checks (e.g., new subjects entered in Jan 2018 have screening date of Jan 2017 – this is an example of entering the wrong year for screening date)
- Frequency distributions or summary statistics
- Cross tabulations of key analysis variables by site
- Subject-level review of variable values

- Open text fields (the study team usually has a process for reviewing all open text fields, but the biostatistician may also review these fields to identify possible improvements in the team process)

- Obvious checks that come up during development of dataset specifications that are not directly addressed through queries

Write a SAS program to run the data checks and create a list of raw data issues. You may find raw data issues using analysis variables, but the output of your data checks program describes the issue, displays raw variable names and values to demonstrate the issue, and displays identifiers for records with the issue. See example output below. Note that if a data check never identifies a discrepancy in the data, you do not need to print the data issue in the output file. Send the raw data issues to the study team by saving the output in a location accessible to all study team members and providing a link to that location. Do not send the data checks output by email, as it may contain sensitive information or PHI (e.g., date of birth in the example below). Update the SAS program as potential data issues are identified over the course of the study. Store your data checks SAS program as in the "Reports and data requests" folder, naming the subfolder "Study Data Checks" or something similar.

Example output indicating raw data issues.

Missing age-related data or age out of range			
StudySubjectID	ScreeningDate	DOB	AgeYears
525252	14JAN17	16MAR18	-1.17
646464	20JAN18	.	.

Run your data checks periodically throughout the study. This helps the study team identify the need for new data validation rules, database changes, training, or other areas for potential improvement. Also, run your data checks prior to reporting external results that meet the criteria for medium or high risk (see WG Determining Risk Level and QC Requirements).

Checking Derived Variables

To promote quality and minimize coding errors, check your code as you create analysis variables and datasets. At a minimum, check the following:

- Data restructuring (e.g., merging, stacking, transposing, or sub-setting)
  - After each operation, verify that the number of rows is as expected.
  - When merging records, verify records link on ID variables as expected.
  - Verify the structure of the resulting dataset is as expected.
  - Verify variable names, lengths, labels, values, and formats are as expected when combining datasets.
- Deriving variables
  - Inspect the distribution of each derived variable.
  - Verify that missing values are accounted for.

Examples

The required tasks are each repeated below, along with examples of how you might perform them. The method used to perform the task is not dictated by this WG. Use any convenient or familiar method that accomplishes the task.

Data restructuring (e.g., merging, stacking, transposing, or sub-setting)

- After each operation, verify that the number of rows is as expected.
  - Open the "Properties" of the dataset within the SAS Explorer Window, or open and view the dataset, or check the SAS log before and after the operation.

    ```
    NOTE: There were 518 observations read from the
    data set WORK.INDATA1.
    NOTE: There were 523 observations read from the
    data set WORK.INDATA2.
    NOTE: There were 519 observations read from the
    data set WORK.INDATA3.
    NOTE: The data set WORK.MERGED has 523 observations
    and 8 variables.
    ```

- When merging records, verify records merge/join on ID variables as expected.
  - View the dataset in the SAS Explorer Window, a PROC PRINT, and/or use the "IN = " dataset option to review a merged dataset.

```
data mergedData;
* the in= option creates a variable to identify
each record's source;
   merge parentForm(in = ParentForm)
   childForm (in = ChildForm);
   by subject_id;
      *create non-temporary variables;
      hasParentForm = ParentForm;
      haschildForm = childForm;
run;

* verify every child form also has a parent form;
proc freq data = mergedData;
table HasParentForm*HasChildForm / missing norow nocol nopercent;
run;
```

- Verify the structure of the resulting dataset is as expected.
  - After transposing a dataset, or after performing merges or joins, view or print the dataset to ensure that columns, rows, variable names and other attributes are as you expect them to be.
- Verify variable names, lengths, labels, values, and formats are as expected when combining datasets.
  - Inspect any variables that had the same variable name prior to combining. Did the procedure truncate data values due to the variables having different lengths? Which label and format were applied? View the dataset, use PROC FREQ to look at the variable, and/or look at output from PROC CONTENTS before and after the procedure.

Deriving variables

- Inspect the distribution of each derived variable.
  - For categorical variables derived based on other categorical variables, review frequencies of the new variables crossed with the input variables.

```
proc freq data = <data>;
table NewVariable * input1 * input2/ list missing;
```

```
run;
```

- For categorical variables derived from continuous variables, you may inspect the new variable and the input variables in a PROC MEANS statement or a plot.

```
* this will show you the observed end-points of each category;
proc means data = <data> min max n nmiss;
class NewCategoricalVariable;
var ContinuousInput;
run;

* this will show you the observed end-points of each category;
proc sgplot data = <data>;
scatter x = NewCategoricalVariable y = ContinuousInput;
* OR;
vbox ContinuousInput /group = NewCategoricalVariable;
run;
```

- For continuous variables derived from continuous variables, you may simply view a selection of row and check the calculation.

```
proc print data = <data>(obs = 20);
   var ContinuousInput1 ContinuousInput2 NewContinuousVariable;
run;
```

- Verify that missing values are accounted for.
  - Copied variables with no missing values initially may end up with missing values due to a merge/join in which not all records from one dataset have a match in the other. Examine the copied variables to ensure that there are no unexpected missing values. You might do this by viewing the variable using SAS explorer, PROC PRINT, etc., or viewing the distribution using PROC FREQ. If you have several copied variables from a single source, it is more efficient to inspect missing values as a group.
  - For derived variables, this may be done while inspecting them in the step above. Look at a selection of rows with missing data and walk through the calculation by hand to understand why they are missing.

```
proc print data = <data>;
where NewVariable <= .z;
* This selects rows where NewVariable is missing,
even if special missing values are used.
.z is the largest special missing value.;
var <input variables and indicators from in= statements>;
```

```
* Should any of these result in a non-missing value of NewVariable?;
run;
```

DSMB

DSMB Timeline and Responsibilities

Prior to screening cutoff

Preparation for the DSMB report begins early on in the study. The following tasks begin as soon as possible:

- Creating mock content (see WG Generating Mock Content).
- Programming and dual programming study analysis datasets (see WG Analysis Datasets).
- Comparing analysis datasets to resolve data issues early.
- Programming study data checks (see WG Data Checks).
- Programming and dual programming tables, listings, and figures (TLFs) (see WG DSMB Results).
- Comparing TLFs to resolve issues early.

Create mock content and analysis dataset specifications, program study datasets, and program data checks concurrently. Mock content will inform the creation of variables, labels and formats. Compare analysis datasets as they are programmed and dual programmed (QC will be performed on the final analysis datasets after data lock).

Continue adding data checks as you program TLFs. Write checks that ensure results will not contain data discrepancies or avoidable missing values when data are final. Work with the study team to resolve discrepancies.

Although QC will be performed on the final DSMB report results, preliminary comparison of TLFs is done after these are programmed and dual programmed. It may take several months to complete each of these critical tasks. Before the DSMB schedules a meeting date, use the schedule of interim analyses in the study SAP to anticipate when the meeting will occur, and set deadlines to complete the tasks outlined above. After the DSMB schedules a meeting date, use the DSMB preparation timeline to set and track deadlines for tasks that are not yet complete.

Create a timeline

After a DSMB meeting is scheduled, create a timeline (see Template and Example DSMB timelines in Resource Hub). The lead biostatistician creates the timeline with input from the study team. Other study team members (e.g., the PM) may lead the creation of a portion of the timeline.

Each study's timeline will be different. Use the sample timeline and example timelines to develop a timeline for the DSMB report. The timeline includes tasks related to creating the report as well as tasks related to sending the report and any supplemental material needed for the DSMB meeting (see WG Sending DSMB Reports). In addition to tasks found in the sample and reference timelines, include tasks unique to your study (see SAP, protocol, etc.) Set deadlines for the completion of each task. Designate a person or group that is responsible for the completion of each task.

Screening cutoff and data entry deadlines

Subjects screened prior to the screening cutoff date will be included in the DSMB report. Subjects screened on/after the screening cutoff date will not be included in the DSMB report, but may be included in a supplemental report that uses data current at the time of the DSMB meeting.

The data entry deadline is the deadline for entering study data required to generate the DSMB report. Study research personnel are instructed to enter study data and resolve all data discrepancies (i.e., queries) relevant to subjects enrolled prior to screening cutoff. This may also include data entered by the study team (e.g., adverse event term coding).

When setting the screening cutoff and data entry deadlines, consider the timing of primary and safety outcomes. For example, if you plan to report on an outcome that is collected at 28 days post-enrollment, allow a month for data entry deadlines, plus a few more days. This allows the 28-day outcome data to be collected and entered, and for discrepancies to be resolved prior to the data entry deadline. Allow time for sites and the study team to perform any data cleaning, coding, adjudication, or other tasks that are required prior to creating a snapshot of the data (see Interim Data Snapshot below).

The study team, study investigators, and research staff are all made aware of the screening cutoff date and the data entry deadline by the PM or CDM. The PM and/or CDM sends reminders to research personnel regarding data entry and queries.

Run data checks often before the planned interim data snapshot, and notify the study team of discrepancies.

Interim data snapshot

After the data entry deadline, run data checks again to make sure the data are complete and clean enough to be used in the DSMB report. Allow time to address any remaining data issues after the data entry deadline. Once the study team determines that data are ready, a snapshot is created. Plan enough time after the data snapshot and before the DSMB meeting to generate results, write the report, and perform QC prior to sending the report and holding the meeting. This could be 4-6 weeks or more prior to the DSMB meeting.

The DSMB report

A few weeks before sending the report, compile materials that will accompany the report (e.g., DSMB charter, protocol/summary of protocol, SAP).

When the data snapshot is available, the lead and dual biostatisticians produce the final TLFs for the DSMB report. The lead biostatistician notifies the faculty biostatistician so that he or she may write the report using final TLFs. The lead and dual biostatisticians begin QC using the QC tracking spreadsheet (see WG QC Tracking Spreadsheet). The lead and dual biostatistician notify the faculty biostatistician of any changes resulting from QC, and notify him/her when all results are matching.

The lead and dual biostatisticians review the draft DSMB report. During this review, the lead and dual biostatisticians verify all results in the report text, and verify any TLFs not previously verified (see WG Dual Programming Results and WG Verification of Deliverable). Verification of the report is documented in the QC tracking spreadsheet. The lead and dual biostatisticians provide feedback to the faculty biostatistician, who makes necessary revisions. The report may require multiple rounds of review and revisions.

After the report is verified and finalized, the study biostatisticians provide the report to the DSMB (see WG Sending DSMB Reports). The study biostatisticians also provide any planned supplemental materials (e.g., enrollment or safety data) at the time of the DSMB meeting.

The DSMB meets to discuss the study (see WG DSMB Meetings).

Lead and dual biostatisticians archive DSMB report materials (see WG Archiving Overview and Timeline).

Subsequent DSMB reports

Create a timeline for each DSMB meeting to review data. Subsequent DSMB timelines are similar to the first timeline. When creating subsequent timelines, allow for revisions to mock content, dataset and result programs, and for revisions to the report. Also, allow time to perform necessary dual programming and verification of revised content. Consult with the study team to make any necessary adjustments to the screening cutoff, data entry, and/or internal study team deadlines for subsequent meetings.

References

- DSMB/DMC Materials

Generating Mock Content

Mock content for a DSMB report describes the presentation of the data to be included in the report. It is the responsibility of the lead or dual biostatistician to create the mock content.

Mock content is created either before or in parallel with analysis datasets and specifications. Review the SAP, raw data documentation (e.g., annotated eCRF), and other study documentation when creating the mock content. Example mock content or reports from other studies may be helpful.

DSMB reports have two main sections: Open Session and Closed Session. The Open Session is the component of the meeting that is open to the DSMB members, the study team, and the Principal Investigator (PI). The Closed Session is only open to the DSMB members and the study biostatisticians. Any content that summarizes data by treatment arm is in the Closed Session. Ensure that the mock content clearly differentiates between the content in the Open versus Closed Session. It also includes a table of contents and shell TLFs.

Mock content is detailed enough to allow a biostatistician familiar with the study data and SAP to create the DSMB report. As the DSMB is the intended audience of the mock content, it is written so that they can understand what is being presented. When drafting the shell TLFs, do the following:

- Include titles and footnotes as they are intended to appear in the DSMB report.
- Identify the population of each TLF, typically in the title or footnote.
- Use placeholder values (i.e., XX (XX%)) to indicate how the data will be displayed. Actual study data are NOT included.
- Denominators for percentages are clear. Denominators could be described in a footnote, provided in the summary (xx/yy zz%), or implied by indentation.
- The mock content contains the actual variable labels and choice sets as intended for the DSMB report.
- Add footnotes to describe analysis methods when necessary.

Send mock content to the study biostatisticians, including the faculty biostatistician, for review. The study PI and the DSMB members may review the mock content at the direction of the faculty biostatistician.

Mock up new TLFs before creating TLFs with study data. After TLFs with study data are created, updates are incorporated directly in the TLFs with study data (rather than updating the mock content).

DSMB Results

Generating results

The lead biostatistician for the DSMB report writes programs to produce all TLFs based on the mock content approved for the DSMB report. Each program generates only one TLF, except in cases where content is extremely similar between multiple TLFs. One program can create more than one TLF in these circumstances. The output of these programs is publication quality so that it can be used in the DSMB report as is, without manual manipulation. The study biostatisticians agree upon the format of the output (e.g., pdf, rtf, tex, etc.). The lead biostatistician for the DSMB report writes programs to generate any other results used in the text of the report.

The dual biostatistician for the report dual programs the results found in all TLFs and results used in the text of the DSMB report (see WG Dual Programming Results).

Reviewing results

The lead and dual biostatisticians review analysis results while the results are being programmed. To ensure that the analyses were performed correctly,

- Check the SAS log for errors, warnings, and other important notes.
- Examine TLFs to confirm results are logical and correctly displayed.
- Review the output with the faculty biostatistician to confirm results are reasonable.

Reviewing the results may result in changes to the TLFs and may prompt additional data checks (see WG Data Checks).

Generating final results

The final version of the output must be created using derived datasets generated from the interim data snapshot. The lead biostatistician for the report ensures the SAS log shows locked data were used to generate the raw data used to create the derived datasets and results.

Sending DSMB Reports

Methods for sending DSMB reports

DSMB reports may be shared with the DSMB members and other appropriate persons either electronically or by sending physical copies. The method used is decided by the faculty biostatistician. With either method, care must be taken to minimize the risk of any unauthorized person receiving or accessing DSMB report materials. Only DSMB members and study biostatisticians receive access to closed session materials; all other attendees are only given access to open session materials.

Electronic sharing of DSMB report

Determine which additional materials need to be sent to DSMB members along with the report. These may include the DSMB charter, the protocol or a summary of the protocol, or any other documents agreed upon by the study team or requested by the DSMB members. The DSMB report is saved in two separate documents, one for each session (open and closed). The document for the closed session may include all open session materials so that closed session participants need only access one file. All DSMB files are shared with DSMB participants according to the DSMB timeline, often approximately 1 week before the meeting. A secure online system such as eRoom, Sharepoint or Florence eBinder is used to share the report files when possible so that access to the materials can be easily restricted to appropriate individuals. Consult with the faculty biostatistician to determine if report files shared via a secure online system also need to be encrypted with password protection. If a secure online system is not feasible, the report files may be emailed. Any closed session report that is emailed must be encrypted by password protection. Do not send the passwords to the files in the same emails as the documents themselves. Consult with the faculty biostatistician to determine if the open session report file(s) also need password protection.

Sending physical copies of DSMB report

Determine the number of total physical copies needed and of these, how many are open and closed session. As with electronic sharing, additional materials may need to be included with the DSMB report. The PM or study admin staff can prepare and print these additional documents and materials. Additionally, request that study admin staff prepare binders with tabs to hold the report and boxes for shipping. The study admin staff can prepare everything except the actual DSMB report. When it is time to print, study biostatisticians are on hand to ensure that DSMB reports are not viewed by unauthorized persons as they are being printed. Study biostatisticians insert the DSMB reports into the binders and seal the completed binders in the shipping packages for any materials being shipped to DSMB members. Once the packages are sealed, they are delivered to study admin staff for shipping. The packages are shipped according to the DSMB timeline, often so that they arrive to DSMB members approximately one week before the meeting. All physical copies that are not shipped are stored in a locked cabinet or drawer when not in use or are shredded.

DSMB Meetings

Prior to DSMB meeting

The faculty biostatistician presents the DSMB report to the DSMB members, but may request that the staff biostatisticians address or explain specific TLFs. Review the DSMB report with the study biostatisticians, and decide who will present each section of the report.

During DSMB meeting

If possible, all study biostatisticians attend the DSMB meeting. While a specific biostatistician may be assigned to take notes, all biostatisticians present are responsible for taking careful notes on

- Items discussed and questions/concerns of the DSMB members
- Clarity/accuracy of TLF contents (including titles and footnotes)
- Requested changes, additions, and/or deletions to the DSMB report
- Decisions made
- Other items and insights

Keep open and closed session notes separated, as closed session notes are not shared with anyone but the study biostatisticians.

After DSMB meeting

Review the notes with the study biostatisticians, and develop plans to implement changes requested by the DSMB members. Communicate decisions made to the study team at the discretion of the faculty biostatistician. A meeting summary will be distributed to the members of the DSMB. The author of the meeting summary is determined on a study-by-study basis. Provide any necessary comments or revisions to the meeting summary when it is sent to you for review.

Manuscripts

Manuscript Responsibilities and Timeline

Responsibilities

The lead biostatistician for a manuscript is responsible for providing the statistical support for that manuscript. This person is not necessarily the lead biostatistician for the study, but is usually either the lead or dual biostatistician for the study. The lead biostatistician for a manuscript follows up with the faculty biostatistician as needed.

The dual biostatistician for a manuscript is responsible for completing QC related to the manuscript. This person is usually the dual or lead biostatistician for the study.

The writing team typically consists of the lead biostatistician, faculty biostatistician, and investigator(s).

Timeline

A study typically has multiple manuscripts in various stages that biostatisticians work on concurrently. Base deadlines for a manuscript are in the manuscript timeline below. Adjust deadlines based on the needs of the study and in consultation with the faculty, supervising, and/or managing biostatisticians.. For example, drafting one MAP may be delayed while you are drafting another MAP for the same study.

Step	Timeline
Manuscript is added to the Manuscript list	As early as possible during enrollment, after the investigator(s) approve the manuscript idea.
MARF is submitted	As early as possible after the manuscript is added to the Manuscript List.
MARF is reviewed and investigators notified	Within 1–3 days of receipt of the MARF, or within a timeline determined by the study biostatisticians.
MAP is drafted and sent to the investigators	Send the first draft as soon as possible within a timeline determined by the study biostatisticians or managing biostatistician.
MAP updated and revised until it is approved	Within a timeline determined by the study biostatisticians or managing biostatistician.

Variables (lead) are programmed	Begin during MAP discussions, directly after investigators or faculty biostatistician review variable definitions, and after any necessary discussions about the variable definitions. Finish within 1-3 months after MAP approval or within a timeline determined by the study biostatisticians or managing biostatistician.
Descriptive summaries are programmed and reviewed internally	Concurrently with variable programming by lead biostatistician.
Data checks are implemented	Concurrently with variable programming by lead biostatistician.
Variables are dual programmed	After variables are programmed by lead biostatistician, before QC is complete, and within a timeline determined by the study biostatisticians or managing biostatistician.
Descriptive summaries are sent to the investigators	Timeline determined by faculty biostatistician.
Results are programmed as described in the MAP	2–4 weeks after MAP is approved and variables are programmed, or within a timeline determined by the study biostatisticians or managing biostatistician.
Results are sent to the investigators	Timeline determined by faculty biostatistician.
Manuscript is drafted, MAP and results are updated as needed, manuscript is edited and finalized	Timeline determined by writing team.
Results are verified, MAP is updated, QC is completed	1–2 weeks after receiving final manuscript draft.
Manuscript is submitted	Timeline determined by writing team.
Manuscript is revised and resubmitted until accepted, MAP is updated and QC is completed based on manuscript edits	Within timelines dictated by the journal.

Manuscript List

The progress on the manuscript(s) for a study is tracked within the study's Manuscript List in Sharepoint. The lead biostatistician for a manuscript is responsible for maintaining the entry for that manuscript in the list, and ensuring that all fields and contents are up to date.

Manuscript list entry fields

To create a new entry for a manuscript, select "new" near the top left of the manuscript list. Select "folder" from the dropdown that appears. Assign the folder name as the short title of the manuscript. Once the folder appears in the manuscript list, select the "Library" tab at the top left of the page, then select "Quick Edit" to modify the other fields for that manuscript's record. Once you are done, select "Stop editing this list" to save your changes. Note that there are validation rules written to check the logic between the fields entered; if needed, these can be suppressed using the "Override Validation" field. Dates when tasks are completed are updated regularly (e.g., during stats or study team meetings, upon completion of a task). While the comments field can be used as needed for a given study or network, you may consider using it as a place to house a detailed status of the manuscript. This would include (1) the date the status was last changed, (2) what has been completed, and (3) the next step (as well as the party responsible). A detailed comment may sound like this: "11/18/2019: Henry sent the log-rank p-value for Figure 1 to Thomas on 08/12/2019. The next step is for Thomas to finalize the manuscript and send it back to Henry for final review prior to submission."

Manuscript list entry attachments

The MARF, MAP, and shared results including descriptive summaries and manuscript drafts are maintained and versioned within the manuscript folder.

Attachment Section Structure

[] indicates a folder. () indicates instructions for naming an item.
Organize the attachments section of a manuscript entry as follows:

- MARF (short name).docx
- MAP (short name).docx

- (Short name of the manuscript).docx
- [Results]
  - [Descriptive Summaries]
    - [Obsolete]
  - [Other folders as needed - e.g., Tables, Figures, Data review)]
  - [Obsolete]

Give files in the [Results] folder and sub-folders descriptive names that contribute to the organization of the folder. Consider including the term "Table", "Figure", or "Listing" to indicate the type of output, and a brief description of the contents. If a table or figure becomes obsolete because the analyses take a new direction, move the file to the [Obsolete] folder.

Security

Everyone in the network has access to the entries as well as their corresponding attachments in the manuscript list. You may have sensitive output that is not to be shared with a study group/network at large until the manuscript is published. Consequently, do not store these items in the list until the manuscript is published. Please see WG Manuscript Results for more information on how to share results.

Reviewing a MARF

A MARF or a similar alternative is required for all manuscripts. Typically, study investigators will make requests for manuscript analysis, though faculty biostatisticians may also submit requests. The MARF is used to facilitate these requests. The lead biostatistician for the manuscript is responsible for reviewing the MARF.

A study investigator submits a request for manuscript analysis using a MARF whenever possible. Alternative forms of request (e.g., email, shell tables, presentations, etc.) are allowed, but need to be approved by the faculty biostatistician. If the faculty biostatistician allows an alternative form of request, save it in the manuscript list and label it as a MARF. The lead biostatistician or the faculty biostatistician acknowledges receipt of the MARF within one business day. The lead biostatistician follows up with the faculty biostatistician if it is not clear who will respond.

After acknowledgment, the lead biostatistician reviews the MARF. The first thing to consider is the degree of similarity with other manuscripts or MARFs. If it is difficult to distinguish the difference between the newly submitted MARF and other MARFs or manuscripts, discuss this issue with the faculty biostatistician. If this does not seem to be an issue, review all of the content in the MARF for feasibility.

To ensure the analysis can be completed and is appropriate within the context of the study, review the MARF in conjunction with the study materials (protocol, CRFs, SAP, etc.). If main outcomes designated in the MARF are not available in the study database, or if there are other major feasibility issues, the lead biostatistician notifies the faculty biostatistician. If there are no major issues, but the MARF lacks the necessary detail to begin drafting a MAP, follow up with the faculty biostatistician to determine how to move forward.

After the MARF has been reviewed and accepted, begin drafting the MAP (see WG Writing a MAP) and preparing questions for an initial meeting with the investigators. Do not update the MARF after the initial meeting. All further development is incorporated in the MAP.

References

- MARF Templates

Writing a MAP

A MAP is written for every manuscript. The lead biostatistician for a manuscript is responsible for writing and updating a MAP, though the faculty biostatistician may take on this responsibility.

The lead biostatistician uses the MARF, SAP, and information gathered from meetings, phone calls, and emails with the investigator(s) to draft a MAP. The MAP is written with sufficient detail to allow the analysis to be performed without revisiting questions with the investigator(s). The investigator(s) are the primary audience. The dual biostatistician for a manuscript will use the MAP in the QC process (see WG QC).

Components of the MAP

The MAP contains the following sections (see MAP Template). Each section is required. You may include additional sections if necessary.

Study Design

In a few sentences, summarize both the main study design and any additional design elements related to the manuscript.

Research Objective and Hypotheses

List the research objectives and related hypotheses for the manuscript.

Population Definition

Describe the population of the manuscript analysis. For secondary manuscripts, include a high-level summary of key inclusion/exclusion criteria.

Variables and Definitions

Define all variables that will be used in the analysis. Definitions include enough detail for investigator(s) to understand and evaluate the variables. Write definitions using familiar terms. Use form names and question wording when needed. Do not use write definitions using code or pseudo-code. Specify the timing of repeated measurements when writing definitions. Also explain the levels/categories of the defined variables,

including how new categories will be created, and any existing categories that will be combined.
Some example definitions:

- Age: Continuous age calculated as the number of years from birth date to randomization date. This is not rounded or floored.
- Age Categories: 0-<6 years, 6-<18 years, using the continuous Age variable described above.
- ED Disposition: Defined using the ED Disposition recorded on the subject information form. Categories are:
  - Admitted: Includes admitted, transferred, and observation unit
  - Discharged
  - Unknown: Includes unknown and missing
- Baseline Heart Rate: Earliest heart rate measured for the subject according to date/time recorded on the Vital Sign Log. Must be measured at, or prior to, randomization date/time. Missing if no such measurement is documented.

Begin programming variables after the definitions have been reviewed, and any necessary discussion has taken place (see WG Analysis Datasets for Manuscript Variables).

Data Summary and Analysis Plan

Describe the plan for performing the analysis in enough detail that another biostatistician could perform the analysis without making ad hoc determinations. For example, if a multivariable logistic regression is to be built using variable selection, the analysis plan would list the candidate predictors, how to select variables for the model, and any model diagnostics that will be used. Include statistical references and example code for non-standard analyses.

References

List references used. If no references are used, leave this section blank.

Summary of Revisions

After inferential results (e.g., p-values and/or estimates associated with the manuscript's main objectives) are shared with the investigator(s), use this section to summarize changes to the study design, objectives/hypotheses, population, variable definitions, and/or analysis plan sections. Include the date the decision was made to make the change, and the discussion or justification leading to the change. Do not summarize changes made prior to sharing inferential results. Do not summarize changes that clarify, but do not change, the intent of the analysis plan.

Appendix: Planned Tables and Figures (i.e., Mock Content)

Mock TLFs are included in the appendix. Use mock content for planning and communicating with the investigator(s). Mock content does not include study data, and does not need to be created with statistical software. Do not update mock content after the MAP is approved. Use TLFs generated with study data and statistical software to refine the presentation of results after the MAP is approved (see WG Manuscript Results).

A mock subject flow diagram is required. This diagram describes inclusion, exclusion, and other criteria used to identify the analysis population for the manuscript. When subject flow is uncomplicated, the diagram could be described in text instead of mocked up (e.g., all screened subjects in the database, with no exclusions).

Other TLFs that will be part of the manuscript analysis are mocked up or described. Include appropriate titles and footnotes with all TLFs.

Mock tables do not need to include all categories of all variables, or list all variables (all variables and levels are required in the Variables and Definitions section of the MAP). Instead, mock tables are used to show the general structure of the table. An example mock Table 1 may look like this:

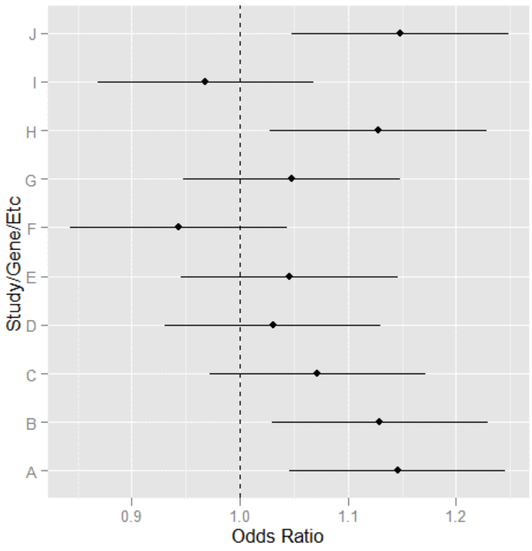
Table 1: Subject Characteristics

Characteristic	Active Treatment Arm N = XX	Placebo Treatment Arm N = XX
Mean Age (SD)	X.XX (X.XX)	X.XX (X.XX)
…other demographics: race, ethnicity, sex	X (X%)	X (X%)
…baseline laboratory variables listed above	X (X%)	X (X%)
…disease history variables listed above	X (X%)	X (X%)

Describe simple figures in words (e.g., "Scatterplot of Age vs. Weight"). Use examples or sketches to describe complex figures. A mock figure of logistic regression results might look like this:

Figure 2. Multiple logistic regression model of XXX

This figure will display the results of a multiple logistic regression model. The odds ratios and 95% Wald confidence intervals will be displayed in a figure similar to the following.



Workflow for producing a MAP

Begin writing the MAP using the MAP template, the MARF, SAP, and any resources provided by the investigator(s). Store the MAP in the Sharepoint manuscript list as an attachment (see WG Manuscript List).

First Draft

Input from the investigator(s) will be required to write each section of the MAP. It typically makes sense to initially draft only a few of the sections.

- If needed, work with the faculty biostatistician to determine which sections will be drafted first. For example, you may only want to draft the objectives, population, and primary outcomes.
- Begin drafting the MAP. Use comments, highlighting, etc. to leave questions for the investigator(s).

- Place the draft in Sharepoint, and notify the faculty biostatistician when it is ready for their review.

- The faculty biostatistician reviews the first draft and provides feedback.

- After approval from the faculty biostatistician, send the MAP and a link to the MAP in Sharepoint, to the investigator(s) for review.

- Work with the faculty biostatistician and the investigator(s) to set a schedule for reviewing, discussing, and revising the MAP.

Subsequent Drafts

Continue revising and writing the MAP. Send drafts to the faculty biostatistician for review, and then to the investigator(s). Discuss updates and revisions of the MAP in phone calls, meetings, emails, etc. In some cases, the faculty biostatistician may instruct you to send the MAP to the investigator(s) without his/her review. Continue the process of writing, reviewing, discussing, and updating the MAP until all sections are written and the MAP contains all the information you need to program variables and perform the analysis.

MAP Approval

Although the MAP is not final until the manuscript is accepted and published, the MAP is approved when the biostatisticians and the investigator(s) decide that it contains all the information needed for the biostatisticians to proceed with the analysis. The MAP approval date is the date when the investigator(s) confirm that they have no further comments.

Changes to the MAP before Results are Shared

As you program the variables and perform the statistical analysis, make any necessary changes to the MAP in Sharepoint. Ask the faculty biostatistician to review the changes after they have been added to the MAP. Share the updates to the MAP with the investigator(s).

Changes to the MAP after Results are Shared

After inferential results are shared, make changes and clarifications to the MAP as before. Additionally, when necessary, add a note to the Summary of Revisions section (see the “Summary of Revisions” section above).

Finalizing the MAP

Before QC is performed, the lead biostatistician reviews the MAP and verifies that it reflects the final manuscript analysis performed, making updates and notes in the "Summary of Revisions" section as necessary. The dual biostatistician reviews the MAP as part of the QC process.

References

- MAP Template

Analysis Datasets for Manuscript Variables

All variables needed for a manuscript's analysis are included in analysis datasets and dataset specifications. Variables and datasets created for manuscripts are dual programmed, as with study-level variables and datasets (see WG Analysis Datasets and WG Dual Programming Datasets).

Identify variables in the MAP that are needed for analysis and do not yet exist in the study-level analysis datasets. Add these variables to the study-level analysis datasets and specifications. If needed, add new datasets to the study-level datasets (e.g., a dataset that is applicable only to a specific subgroup, or a dataset that has a structure other than one observation per subject). Add these additional datasets before or after the main subject-level dataset depending on structure and use.

Derived datasets and dataset specifications specific to the manuscript may be created instead of adding them to the existing study-level analysis datasets if the results of the study primary analysis have been published or if the dataset structure for a manuscript differs significantly from that of the study datasets. The decision to create separate datasets and specifications is considered on a case-by-case basis. When datasets specific to the manuscript are created, the datasets, dataset specifications, and SAS programs for creating the datasets are stored in manuscript-specific folders under Statistical Analysis → Manuscripts (see WG Standard Folder Structure).

Descriptive Summaries

The lead biostatistician for a manuscript creates the descriptive summaries of all variables in the MAP for the subjects analyzed in the manuscript. These include numeric (e.g., frequency distributions, percentages, five number summaries) and graphical summaries (e.g., boxplots, histograms) as appropriate for the manuscript cohort. This output is not publication quality, but is organized in a way that can easily be understood by the investigator(s). A SAS program is written to generate all descriptive summaries and output them to a document or set of documents.

The lead biostatistician reviews the descriptive summaries to identify coding errors data issues, and questions about variable definitions/categorizations. Key review points for the descriptive summaries are below. Resolve coding errors and notify the study team of data issues you identified. Descriptive summaries may result in new data checks and/or queries (see WG Data Checks). Review the following:

- Number of observations with data
- Rare events
- Outliers
- Unexpected values
- Overall variable distributions

Summarize your findings, data issues and questions from your review of the descriptive summaries. Send this summary and the descriptive summaries to the faculty biostatistician. Summarize your findings and unresolved issues/questions for the investigator. Under the direction of the faculty biostatistician, email this summary and the descriptive summaries to the investigator. Save the descriptive summaries that were sent to investigators in the manuscript list for tracking purposes. The faculty biostatistician may instruct the lead biostatistician to distribute the descriptive summaries, findings and questions to the investigator(s) without his/her review.

Incorporate feedback from the investigator(s), and update variable definitions. If variables are modified or added, communicate with the faculty biostatistician and/or investigator(s) to determine if the descriptive summaries need to be updated and reviewed again.

Manuscript Results

Generating results

The lead biostatistician writes programs to produce all TLFs, and to perform all analyses outlined in the MAP. Write these programs as early as possible during enrollment, though results may not be shared with investigators until later. Update these programs to reflect changes made to the MAP by the writing team. Each program generates only one TLF, except in cases where content is extremely similar between multiple TLFs. When possible, the output of these programs requires little or no need to update results manually as the data change. Once the manuscript is drafted, one or more additional programs are written to generate numbers that are used in the text of the manuscript. It is not necessary to include numbers contained in TLFs that will be submitted with the manuscript.

Reviewing results

Review analysis results while you program. To ensure that the analyses were performed correctly,

- Check the SAS log for errors, warnings, and other important notes.
- Examine TLFs to confirm results are logical and correctly displayed.
- Assess model fit and other appropriate diagnostics.
- Compare unformatted output with formatted tables.

Summarize the results, provide interpretations, and describe issues you could not resolve. Send this summary and the results to the faculty biostatistician for review.

Sharing results

Summarize the results, interpretations and unresolved issues/questions for the investigator. Under the direction of the faculty biostatistician, email this summary and the results to the investigator. Copy both the faculty and supervising and/or managing biostatisticians on the email. Other team members may be included as needed. Save non-sensitive results that were sent to investigators in the manuscript list for tracking purposes. (Sensitive materials, such as primary manuscript results that are not to be shared at large until after publication, are not to be stored on Sharepoint. Work with your study team to determine where to store these for tracking.) The faculty biostatistician may indicate some results are not to be shared with investigators until a later point in time based on the study design and/or stage of the study (e.g., primary outcome results during enrollment). The faculty biostatistician may instruct the lead biostatistician to distribute the results, interpretations, findings and issues/questions to the investigator(s) without his/her review.

Generating final results

Final results are created using derived datasets generated from locked data snapshots. Save the SAS log showing that locked data were used to generate the raw data upon which the derived datasets and results are based.

Submission and Revision of Manuscripts

Preparing a manuscript for submission

The lead biostatistician for a manuscript is responsible for reviewing manuscript drafts and providing feedback to manuscript authors. The lead biostatistician is particularly responsible for ensuring that the manuscript describes statistical methods accurately and sufficiently, and that the statistical methods match the analyses performed. The lead biostatistician writes the statistical methods section of the manuscript unless the faculty biostatistician prefers to write it.

After reviewing a draft of the manuscript (this may entail writing a section, adding or updating results, editing, providing comments, etc.), send the draft to the faculty biostatistician for review. In some cases, the faculty biostatistician may instruct you to send the draft to the lead manuscript author without his/her review. Otherwise, incorporate any feedback from the faculty biostatistician into the draft. Then send the updated draft to the lead manuscript author.

Save manuscript drafts in the Sharepoint manuscript list.

Ensure that all results in the submitted version of the manuscript are accurate. Do this by coordinating QC for the manuscript, including dual programming, code review, and result verification (see WG Overview of QC Process).

Submission

The lead manuscript author will submit the manuscript to the journal. When you are an author, you may need to complete tasks requested by the journal (e.g., complete and submit a conflict of interest form). These tasks vary by journal.

Revisions

When revisions are requested by the journal, update any analyses needed and perform any additional analyses. The process of preparing the manuscript for resubmission, including providing feedback to the lead manuscript author and coordination of QC, is the same as the process of preparing the manuscript for submission explained above.

Some analyses may not be included in the resubmitted manuscript (e.g., responses to journal reviewers that do not affect the text or TLFs). These analyses are considered low risk and are not tracked in the QC tracking document (see WG Determining Risk Level and QC Requirements).

Once the manuscript is accepted for publication, coordinate the review of the manuscript archive folder (see WG Archiving Overview and Timeline).

QC

Overview of QC Process

Errors during the coding of datasets and analyses have the potential to affect study conduct in significant ways (for example, decisions of a DSMB to continue a study based on interim analysis) or interpretation of study results. This WG outlines the process for completing QC activities.

QC Process

Planning for the completion of QC activities is started as early as feasible in order to allow sufficient time for QC completion. The lead biostatistician for the deliverable does the following:

- Determine the risk level and associated QC requirements (see WG Determining Risk Level and QC Requirements). This is done as early as possible and definitely before any QC requests.
- Prepare the QC tracking spreadsheet and materials for the deliverable.
  - Add the names of all programs requiring QC and their associated output to the QC tracking spreadsheet (see WG QC Tracking Spreadsheet).
  - If any of the programs requiring QC will undergo code review, copy these programs to the QC folder found within the deliverable specific folder on the P: drive and use GIT to make an initial commit (see WG Standard Folder Structure).

  Note that these steps begin when no more major changes to the deliverable are expected in order to minimize the need for updating the QC request.
- Request that QC be completed by the dual biostatistician. This request is made by email and includes a brief description of the deliverable, as well as links to the location of the QC tracking spreadsheet and other materials needed for QC. The request is made so that sufficient time is allowed for the dual biostatistician to complete the QC, while still respecting the time needed for his/her other projects. One week is suggested, but specific QC requests may require more or less time. Note that dual programming of datasets and/or results may begin earlier and may also require more time to complete.
- Resolve any discrepancies found by the dual biostatistician during QC. This may require additional meetings/discussion and additional rounds of QC.

- Record the date when each QC activity is completed and all discrepancies are resolved, as well as the date when all activities are completed in the QC tracking spreadsheet.

- Notify your supervising or managing biostatistician that the QC tracking spreadsheet is ready for review. The supervising or managing biostatistician enters the date of the final review of the QC tracking spreadsheet. Note that deliverable materials can be sent prior to supervising or managing biostatistician review.

- Copy all materials used for the QC request to the archive along with the QC tracking spreadsheet (see WG Archive Content and Structure). Archiving of materials is done directly after QC is complete to ensure that the correct versions of all materials are stored in the archive.

Determining Risk Level and QC Requirements

The assessment of risk relates to the possibility of harm associated with coding, analysis, or interpretation errors. It also relates to the likelihood of error. The assessment of risk is divided into three categories: low, medium, and high.

The determination of risk is made in conjunction with the faculty and supervising and/or managing biostatisticians, and is done before starting QC activities. It is possible for different aspects of a project to have different risk levels. For instance, the publication of primary and secondary outcomes in a primary manuscript may be high risk, whereas the publication of exploratory analyses in secondary manuscripts may only be medium risk.

Each risk level has differing requirements for QC. The table below provides a summary of risk levels and QC requirements.

Risk Level	Result Type	Examples	QC Requirement
Low	Any output that is not medium or high risk	Internal reports, informal data requests	None
Medium	Exploratory and hypothesis generating results	Observational studies, exploratory analyses	Dual programming of derived variables. Code review of analyses. Verification of deliverable by one biostatistician.
High	Results that impact clinical care, study continuation, safety and efficacy of investigational interventions or significant financial decisions	Efficacy and safety analyses for randomized controlled trials	Dual programming of derived variables and analyses. Dual verification of deliverable, except for direct output* of statistical software; this is verified by one biostatistician.

*Direct output is not manually manipulated before being included in the final deliverable.

Note: The program(s) to create raw datasets require a minimum of code review, but may also be dual programmed.

The risk level for deliverables primarily depends on the impact of results included in the deliverable. The table below summarizes the likely risk level based on the deliverable type.

Deliverable	Risk Level
Abstract with low or medium risk results	Low
Abstract with high risk results	High
DSMB report	High
Manuscript	High or medium, depending on impact of results
Responses to journal reviewers with no associated changes to the manuscript	Low
Presentation/poster with medium risk results	Medium
Presentation/poster with high risk results	High
Report to funding agency	High, medium, or low, depending on impact of results

QC Tracking Spreadsheet

The QC tracking spreadsheet is required for all deliverables requiring QC. The spreadsheet is stored within the deliverable specific QC folder within the Statistical Analysis folder (see WG Standard Folder Structure). It consists of three sections: the header, QC of Statistical Programs, and Verification of Deliverable. Create the QC tracking spreadsheet, complete the header information, and list the programs requiring QC before QC is requested. The exceptions to this rule are the Date QC Completed and Date of Manager or Supervising Biostatistician Review, which are not added until after QC has been completed.

Spreadsheet header

The header contains general information about the deliverable and includes the following fields:

- **Study Name**
- **Deliverable Type**: DSMB report, Manuscript, Poster, Presentation, Sample Selection, Public Use Dataset (PUD), etc.
- **Risk Level**: High or Medium (for the deliverable)
- **Title or Date of DSMB meeting**: Abstract/presentation/manuscript title
- **Date QC Completed**: Date all components completed
- **Location and description of required items previously archived**
- **Validated Macros used**: Named and commit date of the version used
- **Manager or Supervising Biostatistician**: Name of the managing or supervising biostatistician who reviews the QC tracking spreadsheet
- **Date of Manager or Supervising Biostatistician Review**: Date the QC tracking spreadsheet is reviewed by the managing or supervising biostatistician

QC of statistical programs

This section lists all statistical programs that require QC as part of the request. There is one row per program. The following fields are recorded for each:

- **Programs**: Filenames for each program

- **Program Output**: The filename of any output files created and a brief description if needed.
- **QC Type**: This will be dual programming, code review, or validated macro for most programs. For macros written by reputable sources, write, "N/A, written by [reputable source name]." Common reputable sources include SAS and the CDC.
- **Lead**: The name of the lead biostatistician for the deliverable component.
- **Dual**: The name of the biostatistician completing the QC for the deliverable. Note that there may be multiple biostatisticians completing QC activities for a deliverable.
- **Verified by**: This field is only required for dual programming and is the name of the biostatistician who compares the final datasets or the final numbers in the output for dual programmed results.
- **Date Completed**: This is the date QC was completed for each particular program. Completion of QC means that the dual programming or code review has been done and all issues affecting results have been resolved. Note that this date is not recorded if dual programming is started before QC is requested. Only dates associated with formal QC for final materials are recorded.

As with the header, most of this information is entered in the spreadsheet before QC is requested. The exceptions are the "Verified by" and "Date Completed" columns. These are entered after QC is completed for each program.

Verification of deliverable

This section lists any components of the deliverable containing results that are not direct output from programs, including numbers in the text of manuscripts or DSMB section write-ups. This section also lists tables and figures included with the deliverable, unless the exact tables/figures from program output are used. There is one row per component/file. The following fields are recorded for each:

- **Deliverable Component**: Description of the deliverable component.
- **Filename of Deliverable Component**: Specific filename for each component.
- **Reviewer #1**: The name of the first reviewer that verifies all results.
- **Reviewer #1 Date Completed**: The date the deliverable component has been reviewed and any issues resolved.

- **Reviewer #2**: The name of the second reviewer that verifies all results (only if required).
- **Reviewer #2 Date Completed**

This information, except for the "Date Completed" column, is filled in before the QC request is made.

Updating the QC tracking spreadsheet

If changes to the deliverable result in changes to the required QC, new items are added to the QC tracking spreadsheet as needed. Any items having undergone previous QC are replaced if they have changed. The QC tracking spreadsheet contains the versions of all items used in the final deliverable.

References

- QC Tracking Spreadsheet Template
- QC Tracking Spreadsheet Examples

Dual Programming Datasets

Dataset derivation is dual programmed for all medium and high risk projects and deliverables. This process involves the lead and dual biostatistician independently programming all datasets based on the analysis dataset specifications and other study documentation. The dual biostatistician is involved as soon as possible to break up programming work and for early identification of problems. When completing QC for a deliverable, the lead and/or dual biostatistician(s) will perform a final comparison of the datasets. The dual biostatistician reviews the analysis dataset specifications for accuracy and clarity as well as ensuring that the format values and algorithm for each variable is logical. It is the responsibility of both the lead and dual biostatistician to ensure variable algorithms are appropriate based on the protocol, SAP, MAP, MOO, and raw data documentation. During the dual programming process, the dual biostatistician can ask for clarification of definitions and data sources, but neither the lead nor dual biostatistician looks at the other biostatistician's code. When clarification is needed, notes are added to the specifications or incorporated into the algorithms. Additionally, the dual biostatistician does not use any non-validated macros in programming the datasets.

The lead and dual biostatistician datasets are compared to ensure all characteristics described in the analysis dataset specifications are equal, and that the same records are included in each dataset. Example code for this comparison is shown here. At times, values in dual programmed datasets are not exactly equal, but are the same to many decimal places. In these circumstances, the “criterion = ” option can be used to compare values to a certain number of decimal places. Note that the threshold presented in “criterion = 0.000001” is a general threshold to use unless the biostatistics team decides that a different level is needed.

```
* Note: the var statement is useful for preliminary comparisons
of specific variables, but the final dataset comparison will
include all variables;

proc sort data = data1 out = _lead; by StudySubjectID; run;
proc sort data = data2 out = _dual; by StudySubjectID; run;
proc compare data = _lead compare = _dual listall criterion = 0.000001;
id StudySubjectID;
var var1 var2 var3 var4;
run;
```

When comparing datasets, it is often easiest to start the comparison with just the most basic variables, then incrementally compare more complex variables. There are three main reasons for discrepancies when comparing datasets: (1) actual errors in one or both programs; (2) different interpretations of variable or population definitions; (3) errors or conflicting information in raw data source. When discrepancies are found, it is useful to first determine which value is correct by selecting a discrepant value and manually calculating the correct value using the input data and the variable's algorithm in the analysis dataset specifications. Once both the lead and dual biostatistician agree on the correct value, the analysis datasets and/or specifications are updated accordingly. If the discrepancy is caused by conflicting information in the raw data, notify the CDM to query the site(s). If the study team decides not to query, decide with the biostatistics team how to derive the variable given the conflicting raw data.

Formats created for the analysis dataset must also be dual programmed. To facilitate comparison, a formats dataset is created:

```
proc format lib=der cntlout=der.formats;
value NumericFormatName
val = "formatted value"
       ;
quit ;
```

This is only done for new formats; formats that already exist in the raw data are not dual programmed. Once all new formats have been created, compare the derived formats libraries:

```
proc sort data = der.Formats out = _lead; by FmtName Start; run;
proc sort data = comp.Formats out = _dual; by FmtName Start; run;
proc compare base = _lead comp = _dual listall;
id FmtName Start;
run;
```

Dual Programming Results

Prior to dual programming, the lead biostatistician is responsible for ensuring the dual biostatistician has access to study materials required to replicate the results, including the protocol, SAP, MAP, analysis dataset specifications, mock content, and any other related documentation. In addition, the lead biostatistician verifies that the final analyses performed reflect the documentation.

For efficiency, and to reduce the likelihood of error, it is important that the dual biostatistician use the most straightforward and simple statistical procedures with very minimal code to format the dual programmed results. The dual programmed results are not publication quality. Additionally, the dual programmed results are organized to allow easy comparison with the primary results generated by the lead biostatistician. Specifically, the dual biostatistician uses the same order of tables, variables within tables, and formatted values for each variable. The same statistics are reported with minimal extraneous statistics, and the same precision is used for quantitative measures. Figures are dual programmed to look as similar as possible without spending excessive time matching the exact formatting. In situations where the figure is too complex to easily dual program, an alternative method may be used with approval from the supervising or managing biostatistician. The dual biostatistician may use simple macro programs they write to repeat the similar tasks multiple times (e.g., repeat analysis for multiple subgroups). However, the dual biostatistician cannot use non-validated macros from the Statistics project in BitBucket.

While dual programming, it is the dual biostatistician's responsibility to verify that the appropriate methods and variables are utilized based on the study materials. The dual biostatistician will compare output early as programming the results. It is the lead biostatistician's responsibility to follow up on any discrepant results that arise from the dual programming process until agreement is achieved. While completing the QC, the lead and/or dual biostatisticians will do a final compare of all dual programmed output. If there are discrepancies due to incorrect study materials, the lead biostatistician updates the original source material (e.g., SAP, MAP).

There may be situations where a faculty, supervising, or managing biostatistician will fill the role of the lead biostatistician for a deliverable; only one additional biostatistician needs to dual program the results in this case.

Preparing for and Requesting Code Review

Code review requests occur when there are no anticipated changes to the code. The lead biostatistician notifies the dual biostatistician of upcoming code review requests, and informs the dual biostatistician of timeline changes. The initial notification occurs 1-3 weeks before code review is requested. The dual biostatistician plans their schedule to accommodate upcoming code review.

Preparing for code review

The lead biostatistician ensures the dual biostatistician can find the following materials associated with the code requiring review:

- Study protocol
- SAP, if applicable
- MAP, if applicable
- Raw datasets referenced in the code and associated documentation
- Analysis datasets referenced in the code and associated specifications
- Output created in the code
- SAS log
- Current version of the deliverable containing results from the code output
- QC tracking spreadsheet

Additionally, the lead biostatistician copies the code requiring review to the deliverable specific QC folder (see WG Standard Folder Structure) as code review is performed only on copies of the programs, not the originals. Code used to create raw datasets, verify results in the deliverable, and non-validated macro programs referenced in the code are included. The lead biostatistician avoids including programs or code that are not required for the deliverable. Additionally, GIT version tracking is turned on for code requiring review and an initial commit is made (see WG Versioning Project Code).

Simple macro programs used to repeat a simple task multiple times may be code reviewed. However, if complex, non-validated macro programs are used, the macro programs are not code reviewed. There are two options for QC of these results: the dual biostatistician dual programs the results or the lead biostatistician codes the results without macros and arranges for the dual biostatistician to review the non-macro code. The decision of which option to use is made by the biostatisticians working on the deliverable.

Requesting code review

When all the materials are prepared, the lead biostatistician requests code review from the dual biostatistician. The lead biostatistician also provides timelines for code review completion. Typically, when the dual biostatistician has been given advance notification of the code review request, one week is given to code review all relevant materials for a given deliverable.

If a program is modified after code review has already been completed, the code is reviewed again. The lead biostatistician prepares any needed materials as outlined above. The old copy of the program is replaced with the updated program. A GIT commit is made for the updated versions of the code to be reviewed. The lead biostatistician again requests code review from the dual biostatistician.

Performing Code Review

The dual biostatistician performing the code review must become familiar with the study protocol, analysis plan, raw and analysis datasets, raw data documentation, and analysis dataset specifications. It is not necessary for the dual biostatistician to execute the code being reviewed unless this is needed to allow adequate review. The dual biostatistician verifies the following:

- The SAP/MAP provides sufficient detail for the statistical analyses used in the deliverable and the code is consistent with the planned analyses.
- Analyses are executed as intended.
- Code is accurate, with correct syntax.
- Any ERROR, uninitialized, or WARNING notes in the SAS log(s) are brought up with the lead biostatistician. These do not all have to be eliminated if there is a reasonable explanation and results are not affected. If these are not eliminated, add a note to the QC tracking spreadsheet.
- Variables are utilized correctly.
- Programming Standard WGs are followed (see WG Programming Standards).

The dual biostatistician does not need to verify that the information reported in the deliverable reflects the results generated in the statistical program.

After the dual biostatistician has completed the code review, questions and suggestions are provided back to the lead biostatistician. The lead biostatistician responds to each question and/or suggestion made by the dual biostatistician and requests additional review if needed.

After all the feedback from the dual biostatistician is resolved, the lead biostatistician makes a message only commit in the original location of the programs indicating that the code review is complete (see WG Versioning Project Code).

Code review of previously reviewed code

If a program is modified after code review has already been completed, the code is reviewed again. GIT may be used to compare the two versions so that only the changes are code reviewed. However, the dual biostatistician investigates whether the changes may affect the accuracy of the rest of the code and reviews the unchanged code as needed.

Verification of Deliverable

Verification of the deliverable includes verifying all numbers, statements and interpretations in the deliverable text and TLFs. It also includes verifying the statistical methods section of the paper to ensure it accurately reflects the analyses performed. Since transcription errors can occur when results are manually typed or copied and pasted, verify results exactly as they appear in the deliverable, after all editing and formatting are complete.

To verify results in the deliverable, compare results in the deliverable against results generated after QC of the statistical programs is complete. Specifically, complete both the dual programming of datasets and the code review/dual programming of results, with all issues resolved, before verifying results in the deliverable. For medium risk deliverables, the lead biostatistician will typically verify the results in the deliverable. For high-risk deliverables, the result verification is done by both biostatisticians. To verify results, visually compare output from QC'd programs to all of the results in the final deliverable (e.g., sample size, descriptive statistics, p-values, table header N's, footnotes, result interpretations). If listings are more than a page long, spot-check a few random rows of the listing within the deliverable. If listings are short, compare all values of the listing within the deliverable.

Validation of Macros

Validation of macros is the process of testing and confirming that a macro program works as expected. Validated macros generally do not require additional QC. However, the biostatistics team may opt to dual program or code review a macro depending on the risk level of the deliverable.

General process of validating a macro repository

1. A repository containing one or more macros is created and uploaded to BitBucket for the biostatistics team to use (see WG WG Shared Code).
2. A biostatistics manager authorizes the validation of the macro repository.
3. The macro author, or "Lead", clones the macro repository into P:\Statistics\Validated Macros\[Repository Name].
4. Lead creates a validation protocol/report document and drafts the validation protocol. These are stored as part of the repository in a subfolder named P:\Statistics\Validated Macros\[Repository Name]\Validation Materials. Materials that do not need to be committed (e.g., datasets, output) may be saved separately in ...\Validation Materials\Uncommitted Materials.
5. A second biostatistician, or "Dual", reviews the validation protocol and provides feedback until both the Lead and Dual approve the validation protocol.
6. A biostatistics manager reviews the validation protocol and any necessary updates are made until the manager approves it.
7. Dual performs validation and completes the validation report. Lead makes necessary updates until the validation report has no remaining discrepancies.
8. A biostatistics manager reviews and finalizes the validation report by entering his/her name and the date in the report header.
9. Lead updates the macro ReadMe by adding " - Validated" to the repository title.
10. Lead pushes the validated macro repository back to BitBucket and
    - Provides information in the commit message to indicate that the repository is validated.
    - Adds a commit tag of "Validated" to this commit as an extra indication.

- Updates the description of the macro in “Repository Settings” to begin with “Validated -”.

11. Lead notifies all biostatisticians that the macro repository is validated.

12. Biostatisticians pull the validated version of the macro repository to use in current projects.

Details of the process

- **Writing the validation protocol**: Lead writes a validation protocol using the first three columns of the Validation Spreadsheet Template.

  - Program Functions: A function describes a task the macro performs. Functions describe what a user can observe when executing the macro. A function does not describe the inner workings of the macro or how a function is implemented. For example, a function could be, “Rename the variables of a dataset by appending a given prefix to each variable name.” A function would NOT be, “Use proc datasets to modify a dataset and rename each variable using a do-loop.”
    A macro may consist of several functions. In order to create a complete list of functions, Lead considers inputs, outputs, and options/arguments of the macro.
    Lead lists each function in a separate row within the first column of the validation protocol.
    In addition to program functions, Lead lists general requirements in the program functions column. General requirements may not meet the definition of program function as stated above, but they are included in the validation. Include the following general requirements:

    ∗ Delete datasets created by the macro that are not part of the output.
    ∗ Clear global macros created by the macro that are not part of the output.
    ∗ Clear local macro variables before reusing them.
    ∗ Restore system options that were modified by the macro.
    ∗ Use macro comments (%* or /**/).
    ∗ Include a ReadMe document with executable example(s).
    ∗ Follow programming practices explained in WG General Programming Standards and WG SAS Specific Programming Standards.

  - Validation Steps: A validation step describes what the Dual will do to test the program function. For example, one validation step for the rename function example given above may be, “Verify that all variable names in the new dataset consist of the prefix appended to the beginning of the original variable name." The validation step does not include specifics about how the step will be completed.

A single program function may have multiple validation steps. For example, a second step for the rename function example could be, "Verify that each variable in the original dataset results in one variable (no more, no less) in the resultant dataset."

  - Details: The details column of the validation protocol explains how Dual will perform the steps. This column does not include SAS code. This column may optionally reference SAS procedures, datasets, or use pseudo code. Details for the first validation step of the rename validation step example may be, "Run PROC CONTENTS on a dataset before and after the macro, and compare the variable names. They will not match due to the prefix."

    Lead lists all program functions/general requirements, validation steps, and details necessary to validate the macro. If a general requirement does not apply, a reason is listed in the "Validation Steps" column.

    No program function(s) are omitted from the validation plan. If the macro performs a function(s) that Lead does not want validated, that functionality is removed from the macro.

- **Reviewing the validation protocol**: After Lead drafts the validation protocol, Dual reviews the protocol. Dual ensures that the functions, steps, and details include enough information to perform the validation in a thorough and convenient way.

  Dual reviews the macro program ReadMe file and the validation protocol, and ensures that all of the macro program functions are included in the protocol. He or she also verifies that the steps and details will sufficiently validate the program functions.

  Lead incorporates feedback from Dual into the validation protocol before validation begins. Lead or Dual may make changes to the protocol in consultation after validation has begun.

- **Validating the program and completing the validation report**: Dual writes statistical programs and uses test dataset(s) to perform the steps outlined in the validation protocol. In many cases, validation steps require the review of output. In some cases, validation steps require that Dual perform some process(es) outside of statistical programs. Dual organizes programs, output, data used for testing, and any other items necessary to perform the validation in the repository folders.

  Data and other files used in validation may be stored in the repository.

  As Dual performs the validation, he or she suggests changes to the validation protocol so that the final protocol matches the final validation performed. Dual also completes a full code review and verifies that general requirements are met.

Dual uses the "Validated by", "Discrepancies found and resolution(s)", and "Date Completed" columns to track progress. Issues listed in "Discrepancies found and resolution(s)" are addressed by Lead. A resolution is required for each listed discrepancy. Dual enters a completion date when the validation step is complete. Changes made by Lead to the macro program are saved and committed to the local repository.

It may be useful to keep notes, questions or comments while working on the protocol and report. After the validation is complete, remove any notes that are not a part of the header or the protocol or report columns.

Repeating the validation

In the event that a macro repository requires re-validation, the process is repeated. Reasons for re-validation include updates to the programs, new versions of SAS software, etc. Existing functionality is re-validated along with any updated functionality. Changes are tracked using Git commits and described in comments. When re-validating, the steps described above are followed as in the original validation, with the following exceptions:

1. If there were no changes or additions to the functionality of the program(s), the protocol is not revised or re-approved. If there were changes or additions, the protocol is revised and changes approved by the biostatistics manager. In either case, the validation is repeated and documented in the report, and the biostatistics manager reviews and finalized the report.

2. After committing and pushing changes to BitBucket, a "Validated2" commit tag is added to the commit for the second round of validation, "Validated3" to the third, and so on.

Folder structure

Materials used for macro validation are stored on the P: drive as follows:

- *P:\Statistics\Validated Macros\[Repository Name]*
    - Programs that define the macro(s)
    - ReadMe.md file
    - .git (this is a GIT repository)

- .gitignore file (optional: can be used to ignore files in the Uncommitted Materials subfolder)
- *Validation Materials* (subfolder)
  - Validation Protocol and Report Excel file
  - Programs and other materials used to validate the repository. These may be organized in subfolders.
  - *Uncommitted Materials* (subfolder)
    - Datasets, output, and other materials that do not need to be stored in the macro repository in BitBucket, but are required to perform the validation.

References

- Validated Macros
- Validation Spreadsheet Template
- Validation Example (RENAME macro)

Archiving

Archiving Overview and Timeline

All of the materials necessary for the production and documentation of a deliverable classified as medium or high risk (see WG Determining Risk Level and QC Requirements) are archived within a compressed subfolder of the Statistical Analysis\Final archive folder.

After QC has been requested, the lead biostatistician creates the archive folder for the deliverable following the standard structure. Directly after completion of necessary QC, the lead and dual biostatisticians copy all relevant materials to the archive folder (see WG Archive Content and Structure).

If the deliverable is changed after the archive folder has been created and files have been copied over, the lead and dual biostatisticians replace the associated materials in the archive folder with those that reflect the changes.

Once the deliverable has been accepted for publication or presentation, and no other changes are expected to be made, the lead and dual biostatisticians complete a review to ensure all items are present. Following this review, the lead biostatistician notifies his or her superivising or managing biostatistician that the archive subfolder is ready for a final review. Once complete, the archive folder for the deliverable is compressed and made read-only by the lead biostatistician. The following steps compress and add the read only property to the deliverable folder:

1. Right click the deliverable folder.
2. Select "Send to" from the drop down menu.
3. Select "Compressed (zipped)" folder from the submenu.
4. After the compression completes, right click the newly created zipped folder.
5. Select "Properties" from the drop down menu.
6. At the bottom of the "General" tab of the window that appears, select the "Read-only" checkbox.
7. Click "Apply", then "OK".

Archive Content and Structure

Use the checklist below to make sure all necessary items are included in the archive folder after completion of QC.

Data	
Raw datasets	□
Lead derived datasets	□
Dual derived datasets	□
Documents	
Analysis dataset specifications	□
MAP	□
SAP (if applicable)	□
Deliverable QC Tracking Spreadsheet	□
Final deliverable	
Deliverable version after QC is complete, includes final edits if applicable	□
Output (exact output generated by programs)	
Lead output files	□
Dual output files (if applicable)	□
Programs	
Raw data	□
Metadata for raw data (ODM file, REDCap export files, etc.)	□
Lead macro programs (if applicable)	□
Lead derived data programs	□
Lead output programs	□
Lead SAS log(s) for archived programs	□
Dual macro programs (if applicable)	□
Dual derived data programs	□
Dual output programs (if applicable)	□
Dual SAS log(s) for archived programs	□

Exclude materials that are not relevant to the final version of the deliverable from the archive. Additionally, when items exist in multiple versions, only the version used in (or applicable to) the final deliverable is archived. Moreover, if precisely the same materials are used for multiple deliverables, then identical copies of the materials are not archived separately for each deliverable. For example, raw datasets extracted after database hard-lock are archived with the primary manuscript, but are not archived again for secondary manuscripts. In such cases, the QC Tracking spreadsheet specifies the location of previously archived materials (see WG QC Tracking Spreadsheet).

The structure of the archive folder parallels the folder structure outlined in the Final archive checklist and is consistent with the applicable components of the Statistical Analysis Folder.

Below is the archive folder structure; use this structure for all archived deliverables. Folder names are bolded and brackets indicate the folder name describes the specific deliverable. Indentation indicates a nested folder. If a folder only contains a single file, it may be replaced by the file itself. Delete empty folders.

- **Final archive**
  - [**Name**] ([Name] is a descriptive name for the deliverable; add the date to the folder name [Name YYYY MM DD] when multiple deliverables have the same name (e.g., DSMB reports))
    - **Data**
      - **Raw**
      - **Derived**
        - **Lead**
        - **Dual**
    - **Documents** (Analysis dataset specifications, SAP, MAP, QC Tracking spreadsheet)
    - [**Final Deliverable Name**] (Deliverable version containing final edits, after QC is complete. This includes TLFs as well as supplementary TLFs.)
    - **Output** (all TLFs and supplementary information; exact output generated by programs)
      - **Lead**
      - **Dual**
    - **Programs**
      - **Raw data** (all programs/files needed for production raw data)
        - **Metadata**
      - **Lead** (all programs/files needed for [Name] developed by lead)
        - **Macros**
        - **Derived data**
        - **Output**
        - **SAS logs** (SAS logs for archived programs)
      - **Dual** (all programs/files needed for [Name] developed by dual)
        - **Macros**
        - **Derived data**
        - **Output**
        - **SAS logs** (SAS logs for archived programs)

Programming Standards

General Programming Standards

Regardless of the software being used, all final statistical programs meet the following criteria:

- A program header is included for documentation (see Program Header Template).
- Comments are included throughout in order to organize and document code, facilitate code review, and make code more easily transferable between biostatisticians. Comments are included during programming rather than adding them once the program is complete.
- Code is formatted for readability.
- Code is organized to ensure logical flow and ease of maintenance.

The following guidelines help ensure that the above criteria are met and keep code consistent within the biostatistics team:

- Keep line size limited.
- Indent statements from their parent statements.
- Align logically connected sections of code vertically.
- Separate sections of programs with white space.
- Separate logical sections of code into procedures, macros, functions, subroutines, or stored procedures.
- Include a comment for each section of code; include other comments as necessary.
- Name variables and datasets to enhance clarity and organization of the program (insert reference). Appropriate variable and dataset names are:
  - Meaningful/informative of their use/purpose
  - Concise
  - Easy to remember
  - Unambiguous

  Use consistent prefixes or suffixes for similar types of variables (e.g., date/time variable share the same suffix such as “DTM”, variables from the same lab event might all have the prefix “Lab”).

- Keep code as simple as possible. Code will be code reviewed or possibly transferred or modified by other biostatisticians. Use the clearest and most direct route to accomplish the coding task.

References

- Program Header Template

SAS Specific Programming Standards

In addition to all of the basic programming standards (see WG General Programming Standards), all SAS programs conform to the following guidelines:

- DATA and PROC steps are concluded with a *run;* or *quit;* statement as appropriate.
- Specify the dataset with the $data =$ option for every PROC that supports the $data =$ option.
- File paths are defined in setup or run programs (see WG Creating Setup and Run Programs). Other programs reference these paths when needed using macro variables or relative paths.
- SAS comments
  - Include comment blocks to introduce/explain data steps or procedures. If several processes are logically connected, they may be defined within the same comment block. The purpose and data flow of joins or merges are also described in comment blocks.
  - SAS allows for multiple types of comments within SAS programs. The type of comment allowed depends on where the comment is placed within the code. The following table outlines the SAS comment types and where they apply.

SAS Comment	Allowed in
/* comment; */	Anywhere (data/proc steps, macro definition, open code)
*comment;	Data/proc steps, open code
%*comment;	Macro definition, open code

Creating Setup and Run Programs

Setup and run programs are created to streamline the process by which programs for a project/deliverable are executed. Generally, these programs are used at the project level (for creating raw datasets, derived datasets, etc.), for DSMB reports, and for manuscripts. Using these programs also ensures consistency across projects to allow increased familiarity with program structure when a biostatistician transitions to a project. Templates of these programs are available in the Study Startup repository in BitBucket.

File locations

The setup and run programs for a project are stored at the following level on the shared drive: P:\...\Statistical Analysis\Programs\[Lead or Dual]. These files for a deliverable are stored at the deliverable level. For example, the setup and run programs for a manuscript are stored in P:\...\Statistical Analysis\Programs\[Lead or Dual]\Manuscripts\[Name].

Setup

The purpose of the setup program (typically named setup.sas) is to set up the SAS environment specifically for a given project or deliverable. It includes

- All needed libname statements.
- Working directory or macro file path definitions.
- *%include* statements for additional resources (e.g., macro programs).
- Macro variables.
- SAS options setup.
- Other elements of the SAS environment that will be used (e.g., ods escape character).

Run

A run program is a script used to execute several related SAS programs in the appropriate order to accomplish a task. This program is named according to what it accomplishes (e.g. “run all derived data.sas”, “run asthma manuscript.sas”). It is not named “run everything.sas” because this isn’t descriptive. This program typically contains the following elements:

- A PRINTTO procedure for saving the SAS log.
- Changing the working directory to the project Statistical Analysis folder.
- *%include* statements for all of the programs to be run in the appropriate order. In nearly all cases, the setup program is the first program that is executed by the run program.

Versioning Project Code

Git is used to track versions of statistical program files. Programs are created and stored under P:\...\Statistical Analysis\Programs (see WG Standard Folder Structure). The location where the program is stored must be a GIT repository or a subfolder of a GIT repository in order to track changes in programs using GIT commits (see the DCC GIT User Guide for definitions and instructions on creating a repository and other basic GIT procedures). The following folders in P:\...\Statistical Analysis\Programs are GIT repositories: Lead, Dual, and Raw Data.

Versioning programs

- A message is required for every commit. The message is typically "Initial Commit" the first time a file is committed. Messages for later commits will describe the purpose of the commit (e.g., modifications that were made, QC complete, updates for a deliverable, a commit prior to a major modification).
- Commits are required to initiate code review and upon completion of QCfor a deliverable (see below).
- Frequent commits unrelated to QC are encouraged (e.g., whenever the program is in a working state). The exact frequency of commits will depend on the programmer's preference.

Version control and code review

GIT tracking is used to facilitate iterative code review. Follow these steps:

1. Create a QC folder within the deliverable specific subfolder of P:\...\Statistical Analysis\[Manuscripts or Reports and data requests] and make the QC folder a local GIT repository.
2. Copy all programs requiring code review to this location and make a commit with a message indicating "Initial code review". Only files that will undergo code review need to be tracked with GIT.
3. When additional code review is needed, copy the additional or changed programs to the same QC local GIT repository and make a commit with a message explaining the changes.
4. Repeat this process until the deliverable is finalized.

Version control and QC completion

When QC for a deliverable is completed (see WG Overview of QC Process), make a commit in your programs folder at the P:\...\Statistical Analysis\[Lead or Dual] level with a message specifying the deliverable, e.g., "QC is complete for the Dec 2018 DSMB report". All programs associated with the deliverable are committed at this time. If there are no uncommitted changes for programs associated with the deliverable, this is a message-only commit. If additional QC is needed after this commit, another commit is needed once QC is completed.

References

- DCC GIT User Guide

Shared Code

Code intended to be shared with or used by other biostatisticians is stored and managed differently than other statistical programs. Shared code is stored in a GIT repository using BitBucket rather than solely on a local computer or a shared drive. Additionally, a ReadMe markdown file is required (see ReadMe Template). For specific instructions for completing GIT tasks mentioned below (e.g., clone, commit, push) see the DCC GIT User Guide. Some best practices for writing SAS macro programs are summarized in Best Practices for Shared SAS Macro Programming.

Sharing code

The following steps outline the process of sharing code:

- Log into BitBucket and create a new repository.
- Clone this empty repository to a local location (C: or P: drive).
- Copy all the files (or create them) to this repository and commit both the program and ReadMe files to begin version tracking.
- Once the code is ready to share, commit the programs and push the local repository to BitBucket.
- Communicate to the biostatistics team (via email or team meeting) that the code is available in BitBucket.

Updating shared code

Identify the biostatistician responsible for maintaining the BitBucket repository in the repository's ReadMe. If it is unclear, consult with your manager. Discuss your proposed changes with the current owner. In this discussion, decide which, if any, changes will be made as well as who will make and test the changes.

- Clone the BitBucket repository to a local location, or pull the repository to ensure you are working with the most up to date version.
- Make the desired changes to the code in the local repository.
- Be sure to TEST all changes.
- Once testing is complete, push the changes to BitBucket.

Using shared code

- Log into BitBucket and select the desired repository.
- Clone the repository to a location of your choosing (typically under .\Programs\[Lead or Dual]\Macros. If this location is already within an existing repository, the repository must be added as a submodule.

References

- ReadMe Template
- DCC GIT User Guide
- Best Practices for Shared SAS Macro Programming

Project Management

Standard Folder Structure

Files directly related to the statistical support of a study are stored on the shared network drive for the project in a folder named Statistical Analysis. For network studies, the full path will typically be P:\[Network]\[Study]\Statistical Analysis. This folder is organized according to the standard defined in the reference, Statistical Analysis Folder Structure.docx.

When a study involves reporting confidential results, the lead biostatistician requests IT to restrict access to the Statistical Analysis folder to only the study biostatisticians (e.g., lead, dual, supervising, faculty, managing). In such cases, it may be necessary to request explicitly that anyone with global privileges to the shared drive be denied access to the folder. Once the results of the study are no longer confidential (e.g., after publication of the primary manuscript), the Statistical Analysis folder is made accessible to everyone with access to the study folder.

References

- Statistical Analysis Folder Structure
- Statistical Analysis Folder Template

Leading Biostatistics Team Meetings

Biostatistics team meetings are held to coordinate biostatisticians' effort on a project. Among other things, meetings are used to make decisions that require the whole biostatistics team, to set deadlines and to review complex issues. Meetings also allow biostatisticians to track the history of a study through meeting notes, and to regularly follow-up on action items. Do not delay progress on action items when an issue can be easily resolved between team meetings using in-person conversations, phone calls, and/or emails.

Scheduling

You are responsible for coordinating and leading biostatistics team meetings when you are the lead biostatistician for the study. Decide when to begin meeting; biostatistics team meetings typically begin when there is statistical work to discuss or coordinate outside of the project study team meetings. The decision of when to begin holding meetings is made in consultation with the faculty biostatistician. Work with the administrative (Admin) group to schedule meetings.

The lead, dual, supervising, and faculty study biostatisticians attend biostatistics team meetings.

Schedule meetings regularly, but note that the frequency and duration of meetings may vary over the course of the study based on study needs. Typically, meetings are 30-60 minutes and are held twice a month. During busy stages of a study, meetings may be held weekly; at other times, they may be held monthly. If appropriate, regular study updates and ad hoc meetings may be used in place of regular stats team meetings.

Pre-meeting preparation

Prepare an agenda in OneNote 1-2 days prior to the meeting. The agenda includes items for discussion and known action items. The agenda indicates which action items are complete, and lists due dates for items that are not complete. Provide all members of the study biostatistics team with access to the agenda so that they may add items to it.

Arrange the agenda so that topics requiring faculty biostatistician input (e.g. prioritization of activities, complex analysis questions) are discussed at a time that works for the faculty biostatistician (usually the beginning or end of the meeting). Faculty biostatisticians are welcome to stay and participate in all discussions, but he or she may be excused for topics that do not require their input (e.g. discussion of discrepancies in QC, organization of programs and results).

When study results are confidential, write biostatistics team meeting notes carefully to exclude confidential data or save the notes in the access-restricted Statistical Analysis folder (see WG Standard Folder Structure).

Prior to the meeting, send the materials that require review to meeting participants. Send complex or lengthy materials 1-2 days or more prior to the meeting. Materials that are simple enough to review during the meeting may not be sent out ahead of time.

Canceling a meeting

Meetings are canceled for a number of reasons, including (1) the agenda does not contain sufficient discussion items to warrant a meeting and no other participant opposes, or (2) you will not attend the meeting and no other participant opposes to it being canceled.

When you cancel a meeting, notify all meeting participants and send status updates in an email. Additionally, request that the meeting calendar item be canceled by Admin.

When you will not attend the meeting, but it is not canceled, designate another member of the biostatistics team to lead the meeting.

Leading the meeting

Everyone arrives on time to the meeting prepared to discuss the items on the agenda. You, or another designated member of the biostatistics team, take notes of key decisions, action items, and due dates. When possible, assign due dates to action items. Meeting notes may be included in the OneNote meeting agenda. Other meeting participants are expected to take notes pertaining to their work, and to keep notes organized in some way.

Allow all participants to express their thoughts, opinions, and ideas. At the end of the meeting, briefly review decisions, action items and due dates.

Meeting follow-up

Within one business day of the meeting, finalize meeting notes, including decisions made, action items, and due dates. Incorporate any decisions made during the meeting into the applicable statistical documentation (e.g. SAP, MAP, and dataset specifications).

Communication and Email

Biostatisticians are responsible for communicating regularly with members of their study teams and investigators. Respond to all email correspondence that requires follow-up within one working day. This may only be an acknowledgement of receipt of the email, with an approximate timeline for when to expect a complete response. If you will be out of the office, turn on the automatic reply feature in your email. Include the date that you will be back in the office in your automated response.

Include the faculty and managing and/or superivising biostatisticians in all correspondence with investigators. Depending on study needs, the PM or Network Director may also need to be copied on correspondence with investigators. If an investigator sends you a request via email, include an approximate timeline in your response. After a call and/or webinar with an investigator, send an email to him or her summarizing the decisions made, as well as any agreed upon action items. Send this email as soon as possible after the meeting, but at most one working day post-meeting.

For more details on sharing analysis results with investigators, please refer to the WG Manuscript Results.

Project Hand Off

Project hand off occurs when a biostatistician (old biostatistician) is transferring a project to another biostatistician (new biostatistician). The old biostatistician prepares for project hand off by completing tasks in the list below. The study documentation is updated to reflect the current status of the project so it can be a guide to the new biostatistician. The new biostatistician prepares to take over the project by reading all of the study documentation and verifying that access has been granted to study documents.

When possible, the old biostatistician meets with the new biostatistician to pass on information. He or she provides the new biostatistician with the current status of the project, including the next steps for the new biostatistician. If the old biostatistician is leaving the DCC, her or she provides documentation on the status of the project to the his/her manager, supervising biostatistician (if applicable), and the new biostatistician. Meetings with other project team members (e.g., PM and/or CDM) may also be useful.

Once the transfer of information is complete, the new biostatistician begins to act in his/her new role. The old biostatistician, if still at the DCC, attends 1-2 study team meetings and 1-2 biostatistics team meetings (or more if necessary) to facilitate a smooth transition. The new biostatistician directs questions to the old biostatistician as needed after the project has been handed off.

The following checklists are used for project hand off:

- For the old biostatistician:
    - Verify that all programs are up to date.
        - Review comments in code and add any additional comments.
        - Provide notes to the new biostatistician on the status of coding.
        - All programs are committed in GIT (no uncommitted changes).
        - Obsolete programs are moved to subfolders or deleted.
    - Verify that the analysis dataset specifications are up to date.
        - Clean up notes, highlighting, etc., for all changes that have already been incorporated into the datasets. Only leave notes and highlighting that will help the new biostatistician incorporate changes that still need to made.
    - Update manuscript items.
        - Update the manuscript list.
        - Update MAPs.

  - Remove any obsolete output files. Move older file versions to subfolders or delete them.
  - If possible, meet with the new biostatistician to go over the status of the project with them. If the old biostatistician is the lead on the project and is leaving the DCC, provide a document that contains details on the project to his/her manager, the project supervising biostatistician (if applicable), and the new biostatistician. Provide the following information in this document:
    - Current overall/general status of the project, background about the investigators, project history, issues that have been solved or tabled, acronyms and other domain vocabulary, important email chains, etc.
    - Current status of deliverables (reports, manuscripts, etc.) including what has been completed and delivered to the investigator to date.
    - Next steps, including deadlines and timelines for deliverables.
  - The old biostatistician provides his/her manager, the project supervising biostatistician (if applicable), and new biostatistician with a list of items to which the new biostatistician needs access (e.g., OpenClinica, SQL warehouse(s), Query Manager, P: drive folders). If possible, ensure that the new biostatistician has access to all of these items. Provide the new biostatistician with a list of people to whom they can ask questions.

- For the new biostatistician:
  - Complete any study related trainings (OpenClinica, REDCap, Query Manager, etc.).
  - Review study documentation (protocol, SAP, etc.).
  - Review the database structure (e.g., CRFs or OpenClinica).
  - Verify access to everything needed for the study (e.g., OpenClinica, SQL warehouse(s), Query Manager, P: drive folders) based off the list of items provided by the old biostatistician.
  - If dual, meet with the lead biostatistician.
  - Review programs and analysis dataset specifications.
  - Verify that all invitations to biostatistics and team meetings have been received.