1. PURPOSE

This document outlines the verification process for datasets, tables, figures, and listings produced by Global Statistics and Data Science (GSDS) personnel to support BeiGene's regulatory and commercialization activities.

2. SCOPE

2.1 In Scope

2.1.1 This WI applies to all GSDS personnel at all BeiGene locations who work directly on BeiGene systems on the development and verification of datasets and TFLs that support regulatory and commercialization efforts, such as but not limited to:

2.1.1.1 Clinical study reports and regulatory submissions

2.1.1.2 Investigator's brochures, periodic regulatory safety updates, and trial results disclosures in public databases such as EudraCT or FDAAA

2.1.1.3 Data monitoring or steering committee meetings (DMC)

2.1.1.4 Publications, abstracts and presentations associated with seminars and congresses

2.1.2 Programs originally generated on BeiGene systems that are to be transferred to a third party should be developed and verified in accordance with this manual up to the point of transfer.

2.2 Out of Scope

2.2.1 Output associated with internal exploratory analyses and data cleaning or monitoring, such as data review listings, patient profiles, tracking reports, and data reconciliation reports are out of scope of this procedure.

2.2.2 Utilities such as macros and scripts produced and maintained by the GSDS Infrastructure Team are validated via the procedures described in WI *Biometrics Macros and Utilities Validation* (VV-QDOC-00170) and Form *SAS Macro/Utility Change Control Form* (VV-QDOC-00037) and therefore out of scope of this procedure.

3. ABBREVIATIONS AND DEFINITIONS

3.1 **Analysis:** The set of programming deliverables that support a specific regulatory or commercial effort. For example, a primary analysis includes all data sets and TFLs plus all programs, logs, specifications, and documentation that support a primary CSR. A study can have multiple analyses.

3.2 **CSR:** Clinical Study Report

3.3 **DMC:** Data Monitoring Committee

3.4 **EDC:** Electronic Data Capture

3.5 **Output:** Datasets, Tables, Listings and Figures

3.6 **SAP:** Statistical Analysis Plan

3.7 **SLP:** Study Lead Programmer

3.8 **Programming QC Project Tracker:** A file with metadata for all programs, output, and verification documentation in an analysis.

3.9 **TFL:** Table, Figure and Listing

3.10 **VDF:** Verification Documentation Form

4. RESPONSIBILITIES

4.1 **Developer:** Programmer or Statistician responsible for developing the source code to generate a deliverable such as table, figure, listing, or dataset based on relevant specifications.

4.2 **Statistician:** Statistician assigned to the project or study and primary author of the SAP. The Statistician works with the SLP to determine if any dataset or output can benefit from additional statistical review beyond the minimum level of verification identified.

4.2.1 Statisticians are responsible for TFL specifications with input from the SLP; and for reviewing the specifications for all analysis datasets before verification begins.

4.2.2 At a minimum, Statistician will conduct output review and cross-check review to ensure overall quality control.

4.2.3 Statistician may also function as the Developer or Validator. This should be considered for key efficacy datasets/tables that employs hypothesis testing or complicated models. Statistician provides the SAS code for these models which requires independent statistician to validate the statistics. Statistician can conduct spot checks based on either SDTM or ADaM, whichever is more appropriate for validation purpose.

4.2.4 Statistician provides input into Program Tracker.

4.2.5 Signs off the VDF

4.3 **Study Lead Programmer (SLP):** Programmer primarily responsible for:

4.3.1 Assigning staff to project or study programming deliverables

4.3.2 Developing and overseeing delivery timelines

4.3.3 Reviewing SDTM-related specifications

4.3.4 Producing, reviewing, and maintaining ADaM-related specifications

4.3.5 Overall quality of all project deliverables relative to specifications

4.3.6 Compliance with BeiGene procedures for the production and verification of all deliverables on the project or study.

4.3.7 Creating the Program Tracker with input from the Statistician

4.3.8 Creating the VDF and collect signatures before the completion of the Clinical Study Report.

4.3.9 Uploading the signed VDF into the study electronic Trial Master File (eTMF).

Note that the SLP may also function as a Developer or Validator on deliverables at his or her discretion.

4.4 **Validator:** Programmer or Statistician are responsible for verification of the deliverable generated by the Developer based on relevant specifications. Providing documented evidence of the verification process is the responsibility of the Validator.

5. INSTRUCTION

5.1 General Principles

5.1.1 No programming or verification can begin until a written specification exists on the programming server, consisting at minimum of a data set specification for SDTM or ADaM data sets and a shell for TFLs. Specifications must be aligned with the SAP. See Section 4 for individuals responsible for preparing and reviewing specifications.

5.1.2 In CSR analyses, output development follows a linear process such that TFLs are created mostly based on ADaM data sets, which are created solely based on SDTM data sets, which in turn are based on raw data sets. Early studies which do not support filings can be exceptional depending on management approval.

5.1.3 Verification efforts are unnecessary on output files that ultimately are not included in CSRs or publications. Thus, unplanned output for CSRs not described in the SAP and any output for publications need not undergo verification until the Statistician confirms that it will be included in the ultimate report or publication. Until that point, the unplanned or publication output file is considered exploratory.

5.1.4 Verification at BeiGene is risk-based. The SLP and Statistician determine the risk level and verification method according to the impact of the analyses per the guidance in Section 5.2.

5.1.5 Verification involves at least two individuals: a Developer and a Validator. Either or both roles can be fulfilled by a Statistical Programmer or a Statistician. To maintain independence, the Developer cannot also be the Validator for the same output file.

5.1.6 Developers and Validators must critically review relevant specifications before and during their work, and ask for clarification from the SLP or Statistician if anything is unclear, incorrect, or incomplete. This also includes a review of titles and footnotes in the Program Tracker for typing errors and congruency with TFL content.

5.1.7 Developers are responsible to perform due diligence towards producing correct deliverables. Documented verification on a deliverable is the responsibility of the Validator.

5.1.8 Validators must document the first test cycle on a set of deliverables produced by a program in the Template *Verification Documentation Form* (VV-QDOC-02099). Key verification comments should be recorded succinctly in the comments field along with how they were addressed. If this first verification occurred before database snapshot or lock, record a second (final) verification cycle in the VDF after snapshot or lock and before output delivery. Interim rounds of verification that occurred between the first and the final round need not be recorded.

5.2 Analysis Risk Assessment

5.2.1 High Risk: All ADaM and TFLs that directly support clinical study reports, integrations for regulatory submissions, product inserts, periodic safety reports for regulators, responses to regulatory questions, payer and reimbursement documents, safety or independent review groups such as DMCs, and external publications are usually considered as high risk with a few exceptions. The primary and secondary endpoints, and key safety endpoints going to health authorities and publication are critical with high risk. Some TFLs that do not directly support the above analyses, but support company's major decisions can also be considered as high risk.

5.2.2 Low Risk: All other analyses, such as but not limited to those supporting draft or internal data or output reviews and exploratory analyses for new study designs, are usually considered as low risk. Subset table with the same program but different filters, sensitivity analysis, exploratory endpoint can be considered low risk.

5.2.3 The risk level and verification methods should be communicated between statistician and SLP and documented in Template *Programming QC Project Tracker* (VV-QDOC-11820) before the verification starts. Below table is recommended verification methods (see section 5.3 for detailed description) per impact of analyses.

Risk Level	Verification Methods
High	Basic Check + Independent Programming (IP)
Low	Basic Check + Code Review (CR)

5.2.4 Statistician's verification is also needed for some selected high risk TFLs such as but not limited to statistical results for key efficacy datasets/tables that employs hypothesis testing or complicated models, which should also be documented in Program Tracker.

5.3 Verification Method Descriptions

5.3.1 **Basic Check:** both developer and validator should follow the process below:

5.3.1.1 **Log review (LR) (self-inspection):** Check the log and list file (if any). The log file should not have any ERROR messages. WARNINGs and Note messages such as uninitialized values, implicit conversion, "repeats of by variables" should be addressed by code correction and rerun wherever feasible, or else described in the VDF as irresolvable but harmless. Any other messages in the log file that may signal deficiencies in code execution must either be addressed via code adjustment and rerun or described by the Developer in the VDF.

5.3.1.2 **Output review (OR):** In data sets, sense-check missing values and variables with only missing values, outliers within variables, and context between variables.

In TFLs, confirm titles and footnotes align with content of the display and match the TFL shell; check for imbalances between treatment groups; check that summary statistics and point estimates are consistent with the shells, SAP, and other specifications; sense-check output content (e.g., in a vital signs table the title should match content and systolic blood pressure range cannot present a maximum value below diastolic).

It also applies to statistician's regular review of outputs.

5.3.1.3 **Spot-check (SC):** Check subset of final output with original data, check selected results such as p-value, confidence interval, Ns, frequency, means, etc. It also applies to statistician's additional verification on datasets/outputs.

5.3.1.4 **Cross-check (CC):** Compare values in the output to similar values in other deliverables (e.g., a table to another or a listing to a table). This method primarily applies to TFLs. It also applies to statistician's regular review of outputs.

5.3.2 **Code review (CR):** Based on the specifications, review the code to confirm specifications are followed and programming logic is robust. Global macros do not need to be code reviewed, but the macro call should be checked for correct parameter usage. Code review (CR) is always conducted in addition to basic check defined in section 5.3.1.

5.3.3 **Independent programming (IP):** Validator creates a SAS program or updates existing QC program to reproduce all data values in the source output file programmatically based on the same specifications. The Validator will not view or copy code from any source program (even from a different study) into a verification program or vice versa. Where available, use validated global macros in both source and verification programs unless it is inefficient to do so; this approach focuses the IP efforts on all custom pre-macro data manipulation. Independent programming (IP) is always conducted in addition to basic check defined in section 5.3.1.

To validate output, proceed with PROC COMPARE with the following options: listobs listvar criterion=0.00001 method =absolute.

5.4 Verification Process

5.4.1 During production programming, the Developer conducts log review, output review, spot-check and cross-check on his or her output files. This need not be recorded in the Program Tracker.

5.4.2 When ready to start verification, the Developer informs the Validator that the output produced by the program is ready for verification. The Program Tracker provides the locations of all output to be verified. The location of the specifications is provided by the SLP.

5.4.3 Regardless of the type of analysis, the Validator always conducts output review and, for TFLs, cross-check on the production output.

5.4.4 For deliverables determined as high risk, the Validator conducts independent programming. For deliverables determined as low risk, the Validator can decide to use either independent programming or code review, depending on which s/he deems more efficient. See step 5.5 for a diagram of the process using independent programming, and step 5.6 for a depiction of the process using code review.

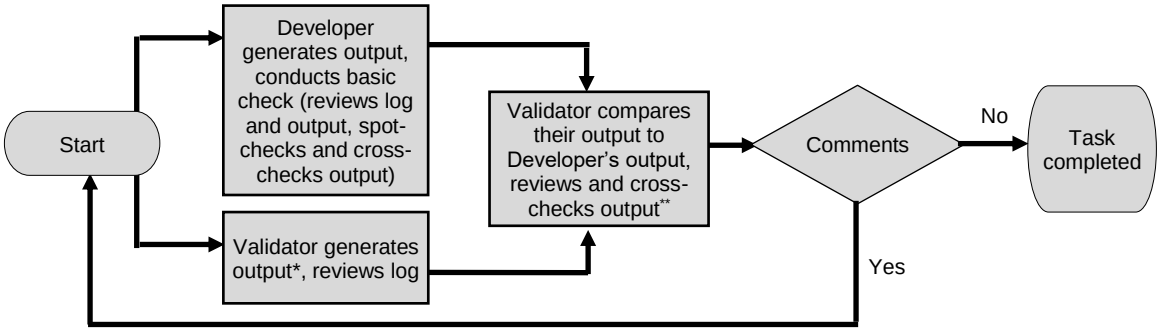
5.4.5 If verification reveals discrepancies, the Validator will communicate the findings to the Developer, who if necessary, will make changes and inform the Validator when the program is ready for re-verification. Disagreements regarding resolution of findings between the Developer and Validator will be escalated to the SLP and, if appropriate, the Statistician to adjudicate.

5.4.6 Steps 5.4.1 through 5.4.5 will be repeated until the output is deemed accurate on all verification methods.

5.4.7 After completion of verification, the Validator records all necessary details in the Program Tracker/VDF and sign off the VDF.

5.4.8 SLP files the signed form into the study electronic Trial Master File (eTMF) at the conclusion of analysis when the CSR is finalized.

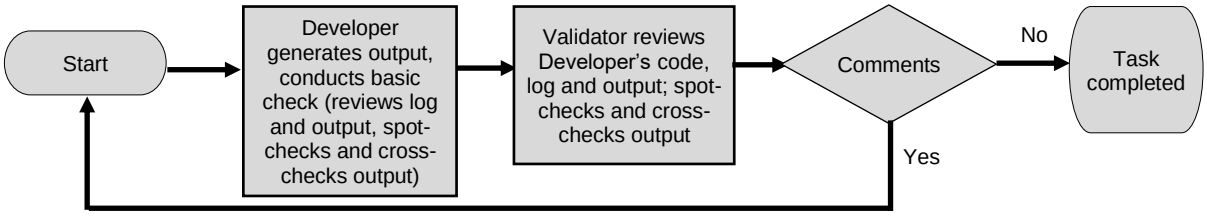
5.5 Flow Diagram of Verification Using Independent Programming



*For TFLs verification, validator can generate and compare the corresponding datasets creating the TFLs.

**For Statistician, output reviews and cross-checks will be conducted regularly. And additional review using spot-checks will also be decided by Statistician based on analysis contents.

5.6 Flow Diagram of Verification Using Code Review



5.7 Verification of Deliverables from Outsourced Studies

5.7.1 When receiving datasets and TFLs from third parties, follow the verification steps below. These are in addition to verification done by the third party per their operating procedures.

5.7.2 The SLP stores all programming deliverables, including but not limited to SAS programs and logs, TFLs, and SDTM and ADaM datasets on the AWS as per the departmental directory structure.

5.7.3 The SLP and Statistician identify which TFLs are considered key to the analysis. These TFLs and all variables in SDTM and ADaM data sets supporting them are registered in a Program Tracker and will undergo output review, cross-check (TFLs only), and independent programming verification per Section 5.4 and step 5.5 in this WI.

5.7.4 If the study is included in a submission, the SLP or designee ensures SDTM and ADaM CDISC conformance checks are reviewed and addressed; checks define.xml for accuracy; reviews all production logs for errors and warnings; and performs code and output review on all other deliverables.

5.8 Changes in SDTM or Analysis Datasets After TFL Delivery

5.8.1 Occasions may arise where there are changes in SDTM or analysis datasets referenced in TFLs that were already delivered outside of GSDS. Example reasons include a database unlock, non-EDC data has been updated or come in late after database lock, programming correction, or specification change. Datasets and TFLs impacted by such changes must be recreated, re-verified, and re-delivered while those not impacted may be left unchanged per the steps below. For raw data changes, Data Management must inform SLP and Statistician which raw datasets and variables were changed.

5.8.2 Developer creates a permanent back-up copy of the dataset in a subfolder on the file server, modifies the production program as needed, and reruns the production data set, overwriting the current output file.

5.8.3 Validator performs verification on the new output file per Section 5.4.

5.8.4 Validator also confirms via a PROC COMPARE to the back-up copy that only the expected changes occurred. If this comparison shows unexpected changes, consults with the SLP and Statistician on appropriate action. Adds this comparison to the verification program, and records in the comments field of the VDF why the re-run occurred and the complete location of the comparison output.

5.8.5 SLP determines for each dataset update which other datasets have dependencies on the updated dataset. Follows the steps 5.8.2 through 5.8.4 for those datasets. If the comparison output on a dependent dataset confirms all its content is unchanged, indicates in the VDF that no further datasets or TFLs dependent upon this dataset need to be re-run.

5.9 Changes in TFLs Following Dataset Updates After Delivery

5.9.1 After a dataset update, previously delivered TFLs only need to be re-run if they reference variables with values shown to have changed in that dataset per step 5.8.4.

5.9.2 SLP, with advice from the Statistician, determines which TFLs depend on updated variables and works with the Developer and Validator to re-run and re-verify those TFLs. Alternatively, if many datasets were updated, the SLP with advice from the Statistician can decide to re-run all TFLs.

5.9.3 Validator records a brief note in the VDF comments field describing which dataset updates triggered the re-run for TFLs that were re-run and re-verified.

6. REFERENCES

6.1 Controlled Documents

6.1.1 VV-QDOC-00037 - Form - *SAS Macro/Utility Change Control Form*

6.1.2 VV-QDOC-00170 - WI - *Biometrics Macros and Utilities Validation*

6.1.3 VV-QDOC-02099 - Template - *Verification Documentation Form*

6.1.4 VV-QDOC-11820 - Template - *Programming QC Project Tracker*

6.2 Regulatory References

6.2.1 N/A

6.3 Other References

6.3.1 N/A

7. APPENDICES

7.1 N/A

8. DOCUMENT HISTORY PAGE

Version	Effective Date	Brief Description of Change
1.0	13 Sep 2018	Original Work Instruction.
2.0	10 Sep 2020	Updates to document for Statistics Participation in QC, risk assessment, validation and verification methods, and details around use of the tracker. Removed template in this document.
3.0	30 Oct 2020	Admin change to correct the section 4.4 and 4.3 as Study Lead Programmer responsibilities were inadvertently moved to Validator responsibility section.
4.0	01 Jun 2021	Added the requirement of uploading signed VDF into eTMF in section 4.3.9 and 5.4.8. Removed controlled document VV-QDOC-00697 from section 6 due to no reference in WI text.

Document Approvals

Approved Date: 20 May 2021

Task: SME Approval Verdict: Approve changes & release	Wazir Woods, (wazir.woods@beigene.com) Functional Area Representative 19-May-2021 20:15:47 GMT+0000

Task: SME Approval Verdict: Approve changes & release	Kelly Shart, (kelly.shart@beigene.com) Functional Area Representative 19-May-2021 21:10:47 GMT+0000

Task: SME Approval Verdict: Approve changes & release	Shibao Feng, (shibao.feng@beigene.com) Functional Area Representative 19-May-2021 22:24:01 GMT+0000

Task: SME Approval Verdict: Approve changes & release	Yasha Li, (yasha.li@beigene.com) Document Owner 20-May-2021 01:08:46 GMT+0000

Task: SME Approval Verdict: Approve changes & release	Cindy Song, (cindy.song@beigene.com) Functional Area Representative 20-May-2021 01:32:04 GMT+0000

Task: SME Approval Verdict: Approve changes & release	Carrie Shi, (carrie.shi@beigene.com) Functional Area Representative 20-May-2021 01:47:19 GMT+0000

Document Approvals

Approved Date: 20 May 2021

Task: QA Approval Verdict: Approve changes & release	Namisha Bansal, (namisha.bansal@beigene.com) QA Approval 20-May-2021 22:57:22 GMT+0000