VERTEX

THE SCIENCE *of* POSSIBILITY

QUALITY ASSURANCE WITHIN STATISTICAL PROGRAMMING

TODD CASE, MARGARET HUANG

SEPTEMBER 7TH, 2023

AGENDA

- **Why** we perform Quality Assurance (QA) within Statistical Programming?
- **What** is Quality Assurance (QA)?
- **How** is Quality Assurance (QA) implemented within Statistical Programming?
- **When** to kick off Quality Assurance (QA)?
- Vertex Quality Assurance (QA) team Background and Overview
- Examples of Quality Assurance (QA) Findings

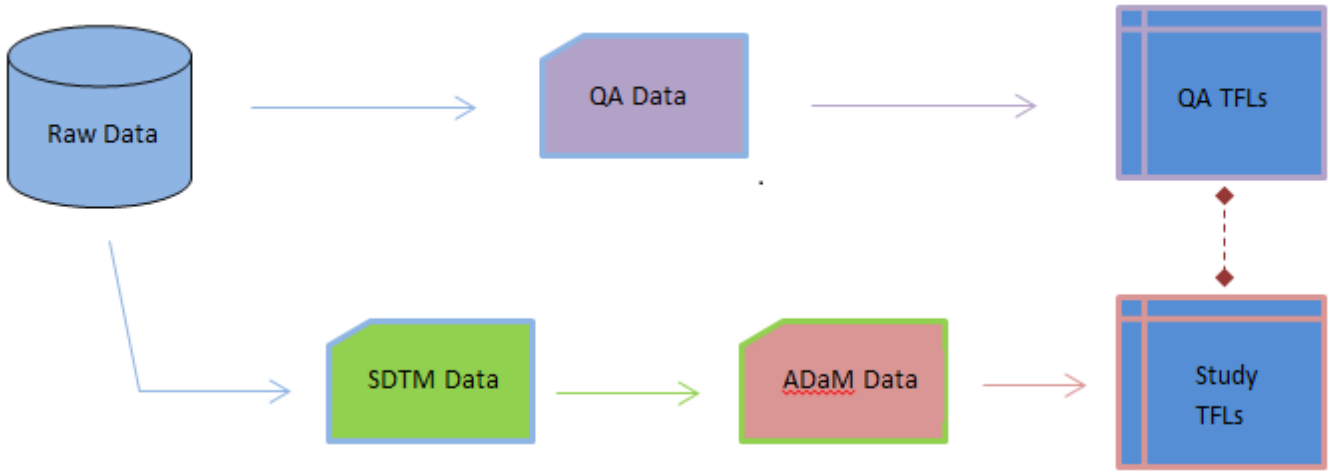
➢WHY WE PERFORM QUALITY ASSURANCE (QA) WITHIN STATISTICAL PROGRAMMING?

- **Quality** – 3rd level of independent review for key studies
  - identify and proactively address any potential issues and risk
  - ensure a higher quality outcomes
- Review documents independently to ensure definitions are clear, accurate, reliable, and align with the standards
- Ensure consistency within Vertex studies
- Opportunity to use non-SAS software in the QA team: R, R shiny, Python, etc.

What is Quality Assurance (QA)?

- Independent programming for TFLs from raw data in parallel with the study team's deliverables.
- Confirm that "high priority", "high risk", or "novel" deliverables received a "triple-check" of their accuracy.
- Confirm that the study team's TFL deliverables and study's SAP (Methods), Specifications, & tables shells are in alignment.
- To date, our QA work scope has included Key efficacy and Key safety deliverables for each study.

FLOWCHART OF THE QA PROCESS



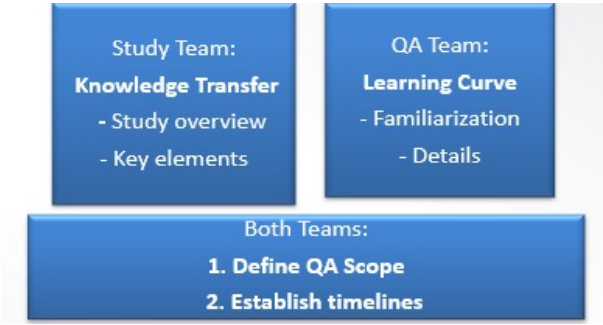
QA PROCESS VS QC PROCESS

	QA	QC
Who	QA team	Study team
Scope	Selected tables (Key efficacy + Key Safety)	ALL TFLs
Document	QA plan	QC plan
Reference Documents	Protocol, SAP, TFL shells	Protocol, SAP, TFL shells, Dataset Specifications
Source data	Raw data (* used same dummy data as study team's if Restricted Data)	SDTM/AdaM datasets
Study	Submission Studies, Pivotal Studies, Studies with unique efficacy endpoints, Outsourced Studies, etc.	ALL
Software	SAS, R, Shiny, Python, etc.	SAS

➢**How** is QA implemented?

- ➢Collaborate closely with the study team
  - ➢Define the QA Scope
  - ➢Set up the QA Timeline
- ➢Utilize the QA Checklist
- ➢Establish clear QA Communication channels
  - ➢Ad hoc meetings/emails/Teams/Phone calls
- ➢QA Work Path – a separate work folder under each study's reporting folder
- ➢Maintain a QA Calendar yearly for QA assignments.

  (Flexible to revisit if there are any urgent requests / high priority requests)



➢ **When** to Kick off QA process?

- Study Team should have these done prior to QA

Ensure that the Protocol, SAP (Methods), CRF, and TFL shells are stable.

Resolve any Known raw data issues or document them for the QA team's awareness.

Generate Study Datasets and TFLs for at least one dry-run and pass internal QC.

➢QA Checklist – Documents

Review the study SAP (Methods), protocol, CRF, and TFL shells to ensure they are accurate, clear, consistent, and conform to Standards.

Ensure all changes made along the way (e.g., those are made through emails, Teams or Ad hoc/urgent meetings) are clearly documented as the study progresses towards DBL.

➢ QA Checklist - Others

- ➢ Has the timeline been discussed with the study team?
- ➢Has the QA plan been agreed upon with the study team?
- ➢Do QA results match with study reports?
- ➢Have QA results been communicated with the study team?
- ➢Are the study populations across TFLs consistent?
- ➢Do the data summaries make sense?
- ➢Do the listings match with corresponding data summaries?
- ➢Do precision on decimals follow the Table shells?
- ➢Any QA findings need to be escalated?
- ➢Any suggestions or recommendations to the study team?

QA Communications with the Study team

TIMELY

- Respond to questions as soon as possible

DOCUMENT

- Track all communication in QA plan

SUMMARY

- Provide final comments in email

➢Summary

Who: A separate/independent team within statistical programming.

What: An independent review of key efficacy/safety tables for key studies.

When: Initiate kick-off discussion.

- QA work is performed after TFLs have passed QC.
- QA team provides QA comments prior to database lock.

Where: In a QA sub-directory at the reporting level managed by the QA team.

How: Raw data is used to generate QA data and outputs.

- Documents used : *study protocol, SAP Methods, CRF, and TFL shells.*
- Don’t use project team’s SDTM/ADaM dataset Specifications.
- Differences in table outputs are presented to the study team by QA team.

- Quality Assurance: Background/Overview in Vertex
  - **2015: Independent QA begins.**
  - **2018: More standard practice begins:**
    - *Reusable programs generated for QA datasets & TFLs.*
  - **2019 Milestones achieved:**
    - *QA of many studies*
    - *Creation of QA directory and QA folder.*
    - *Present QA findings at various meetings.*
  - **2021 Expanding the use of non-SAS software**
    - R & R shiny)

EXAMPLES FROM QA FINDINGS

- *Inconsistencies within the same study*
- *Inconsistency of displayed parameters across domains for similar tables or different tables in the same domain*
- *Inconsistency with specs / Shells vs SAP (Methods)*
- *Footnotes or title and table not aligned*
- *Multiple Units were used for the same parameter in the same deliverable (e.g. protocol defined unit is not SI unit)*
- *Inconsistency with Standardized TFL shells*

EXAMPLES FROM QA FINDINGS

- ***Inconsistency through different studies:***
  - *Age at Visit (Years) calculation – 15 days vs 15.22 days vs 0.5 months.*
  - *Solution: Ongoing study will follow the rule for using 15.22 days instead of 15 days (*ensure relevant study team members are aware of this topic before making a decision)*

AGEVIS	Age at Visit (Years)	Num(8)		Derived: Derive age at every visit in years using age collected at informed consent i.e. VS.VSSTRESC when VS.VSTESTCD="VSAGE" and VS.VISIT="SCREENING". Derived as 1. age at informed consent in days i.e. [(years*365.25)+(months*30.4375)+15] 2. age at visit in days i.e. [age at informed consent in days+(assessment date-ADSL.RFICDT)] 3. age at visit in years i.e. [age at visit in days/365.25].
AGEVIS	Age at Visit (Years)	Num(8)		Derived: Derived age at each visit in years. AGEVIS is derived using the following steps. 1.) Age at informed consent is calculated in days as (years*365.25)+(months*30.4375)+15.22. Age in years and months is taken from VS.VSSTRESC where VSTESTCD="VSAGE". 2.) Age at the visit of measurement in days is calculated as age at informed consent in days (from step 1)+ measurement date (ADT at each visit)- informed consent date (RFICDT). 3.) AGEVIS in years is calculated as age at visit in days/365.25.
AGEVIS	Age at Visit (Months)	Num(8)		Derived: Age in months used to derive z-scores based on Centers for Disease Control and Prevention (CDC) growth chart. Populated only for ZBMI, ZWEIGHT, and ZHEIGHT. Derived as: 1. floor((Analysis Date - Informed Consent Date)/30.4375 + Age at Informed Consent in Months)+0.5) 2. For ZWEIGHT and ZHEIGHT, add 0.5 if the age derived in step 1 is less than 240 months at a visit. For ZBMI, add 0.5 if the age derived in step 1 is less than or equal to 240 months at a visit.

EXAMPLES FROM QA FINDINGS

- ***Incorrect analysis flag being used.***
  - Safety tables (LB, VS, ECG) containing records outside the Treatment-Emergent period, linked to the usage of *ONTRTFL vs. ANL01FL.*
  - *Incorrect derivation of Analysis visit window*
  - *Nominal* visits used to assign AVISIT instead of *analysis visit* windowing by dates (not following SAP (Methods)). This occurred in one vendor's outsourced study.
  - In MMRM table shells, encourage the study team to include dynamic footnotes for the covariance structure rather than pre-specifying (in cases of non-convergence).

Replace pre-specifying "Covariance structure = UN " or "Covariance structure = CS " with "Covariance structure = &cov." in the model specification footnote.

ONE EXAMPLE OF THE QA SUGGESTIONS

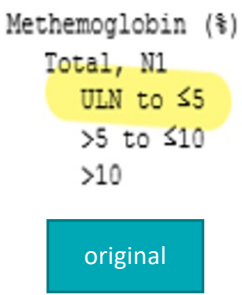
Example: QA Suggestions for Lab Threshold Analysis (TA) Criterions

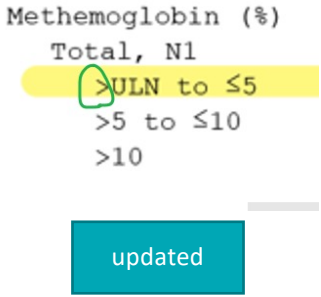
Team agreed.

QA suggested to use

‘>ULN to <=5’ instead of

‘ULN to <=5’ to remove the cases which are equal to ULN.





USE OF NON-SAS SOFTWARE IN THE QA TEAM

QA is a good place to explore and utilize non-SAS software.

- Implemented a data **Exploration & Visualization** platform for interactive data visualization.
- Quality Checking: Developed a reusable tool for Graph Validation.
- Linked key safety and efficacy datasets on an R platform to provide shortcuts for accessing patients' information (2021 Summer Intern Project).
- Evaluated the admiral package and created ADSL by R & Identified other useful packages for dataset creation (2023 Summer Intern Project).
- Cross-checked cSDRG against SDTM datasets and metadata
  - Extracted relevant data/tables from cSDRG
  - Compared the data to relevant data in SDTM datasets and metadata
- On-going: Generating CSR TFLs using R (R Working Group)

THANK YOU FOR YOUR TIME!!

Questions??

Todd Case

Senior Director, Statistical Programming

Vertex Pharmaceuticals, Inc.

50 Northern Ave, Boston, MA 02210

todd_case@vrtx.com

www.vrtx.com