

%kqtlfcomp

Overview

The `%kqtlfcomp` macro performs comprehensive comparisons between production and QC versions of TLF (Tables, Listings, and Figures) datasets. It provides extensive customization options for handling formatting differences, variable attributes, and comparison criteria, making it particularly useful for clinical trial reporting quality control processes.

Version Information

- Version: 1.0
- Last Updated: 09SEP2021
- Author: Atorus Research

Dependencies

- SAS Version: SAS 9.4 V9
- No macro dependencies

Parameters

Parameter	Required	Default	Description
output	Yes	-	Output dataset name to compare
prod_lib	No	tfl	Production library name
qc_lib	No	vtfl	QC library name
qc_output	No	-	QC dataset name (if different from production)
where	No	1	WHERE clause for filtering comparison
prod_drop	No	-	Variables to drop from production dataset
dropchecked	No	N	Flag indicating QC review of dropped variables
ignspl	No	Y	Ignore split characters in comparison
chrspl	No		Split character to ignore
ignln	No	Y	Ignore variable lengths
ignspc	No	N	Ignore spacing differences
ignlbl	No	Y	Ignore variable labels
ignfor	No	Y	Ignore variable formats
ignifor	No	Y	Ignore variable informats
compress	No	N	Remove all spaces before comparing

Parameter	Required	Default	Description
crit	No	-	PROC COMPARE criteria specification
debug	No	N	Retain temporary datasets for debugging

Return Values/Output

- Creates comparison datasets:
 - `prod_[output]`: Production dataset with applied modifications
 - `qc_[output]`: QC dataset with applied modifications
 - `dif_[output]`: Differences between datasets
- Log messages indicating:
 - Missing or invalid parameters
 - Dataset existence checks
 - Variables dropped from comparison
 - Comparison results and differences

Processing Details

1. Parameter validation:
 - Checks required parameters
 - Validates Y/N parameters
 - Verifies dataset existence
 - Confirms library accessibility
2. Dataset preparation:
 - Applies specified filters
 - Drops requested variables
 - Handles variable attributes
 - Processes formatting differences
3. Text handling:
 - Manages RTF instructions
 - Processes split characters
 - Handles spacing differences
 - Applies compression if requested
4. Comparison execution:
 - Performs dataset comparison
 - Generates difference reports
 - Manages temporary datasets
 - Handles debug output

Examples

```
/* Basic comparison */
%kqtlfcomp(output=t_14_3_1_1);

/* Detailed comparison with custom settings */
%kqtlfcomp(output=t_14_3_1_1,
    prod_drop=row1b1,
    dropchecked=Y,
    prod_lib=tf1,
    qc_lib=work,
    where=%str(lbcat="HEMATOLOGY"),
    ignspl=Y,
    chrsp1=|,
    ignln=N,
    compress=Y);

/* Debug mode with custom criteria */
%kqtlfcomp(output=t_14_3_1_1,
    crit=0.00001,
    debug=Y);
```

Common Issues and Solutions

1. Dataset Not Found

- Error: "production/qc file did not exist"
- Solution: Verify library and dataset names

2. Dropped Variables

- Warning: "vars dropped from the production dataset"
- Solution: Review and set dropchecked=Y if correct

3. Multiple QC Datasets

- Error: "qc library did not contain a clear dataset"
- Solution: Specify exact QC dataset name

Notes and Limitations

1. Dataset handling:

- Maximum output name length: 27 characters
- Automatic dataset cleanup
- Case-insensitive dataset names
- Library-specific processing

2. Comparison options:

- Multiple ignore flags available
- RTF formatting handling
- Split character processing

- Space compression options

3. Performance considerations:

- Memory usage for large datasets
- Temporary dataset creation
- Debug mode impact

See Also

- [%kqucomp](#): General dataset comparison
- [%xuct](#): Controlled terminology validation
- [%xumrgcs](#): Dataset merging utility

Change Log

Version 1.0 (09SEP2021)

- Initial release
- Comprehensive comparison functionality
- Multiple formatting options
- Debug mode support