Volunteer Grab WQ R Code

Volunteer Monitoring\datamanagement\R\data\CRWC\2013\crwc13VolData.R

Required Packages:
XLConnect
plyr
dplyr
reshape2
psych
RODBC

DATA INPUT

Submission Details
- *Sub ID*
- *Activity Organization*
- Batch Type

Workbook Details
- Directory with data auxiliary files
- File path

Worksheet Details (Data and Project_Info)
-start and end row

subid
actorg
dbatch
dir
file, etc.

Load: QCcrit.csv, AnomCrit.csv, and CharNames.RData

Qccrit, AnomCrit, and CharNames provide data used by code to make decision and must be in the directory of your file

Load Excel workbook

gd- wq data

prj- project info

Rename headers

Generate:
gd$item (auto numbered)
gd$actroot (DateTime-LASAR)

Structural repairs for data types and NA's for blank values

Duplicate Data checks by row
1)All fields except item duplicates are DELETED
2) duplicates in results and actroot flagged in csv
3) Duplicate actroot values flagged in csv

RedundantData.csv

RedundantResults.csv

RedundantActRt.csv

RESHAPE MESSY gd DATAFRAME

prj

Get list of headers for each characteristic: Apply list of relevant suffixes to each unique characteristic ID in the prj df and append gdmeta for sample info

charsall- list with relevant column headers for each characteristic

Test all in list are gd headers

This is a one way test that does not to make sure that all the characteristics are covered

Subset gd by 0 rows and first list of headers from charsall, then rename to generic header names (gdmeta and sufx0

gdl

"LASAR", "SiteIDcontext","DateTime","DupBatchKey", "item", "actroot", for each characteristic: "_r", "_qual", "d_r", "d_qual", "_PREC", "_ACC", "_DQL", "_m", "_cmnt"

For each char in charsall subset gd by the vector of header names and rbind them to gdl. Then remove blanks

Generate gdl$ActType: DEQ Activity Type ("S" or "FP"

Generate Field Duplicate Data Before Stacking Duplicate Data in Result Column

prj
charAD
charLD

Assign gdl$QCcalc (NA, AbsDiff, LogDiff, or RPD") by For loop each row of data in gdl,
If ActType = "S" then Qccalc = NA
Else if charid on vector of charLD log diff char, the Qccalc = LogDiff
Else if charid on vector of charAD absolute diff only char, Qccalc=AbsDiff
Else if mean of duplicate pair > prj$LowLevelQCLimit Qccalc=RPD
Else Qccalc = AbsDiff

Generate gdl$prec_val with for loop through each row assigning NA or LogDiff, AbsDiff, or RPD value based on the gdl$QCcalc value .. Assign as numeric field

QCcrit

Generate gdl$DEQ_prec by comparison of gdl$prec_val to QCcrit criteria

Complete Stacking/Tidying Data

prj

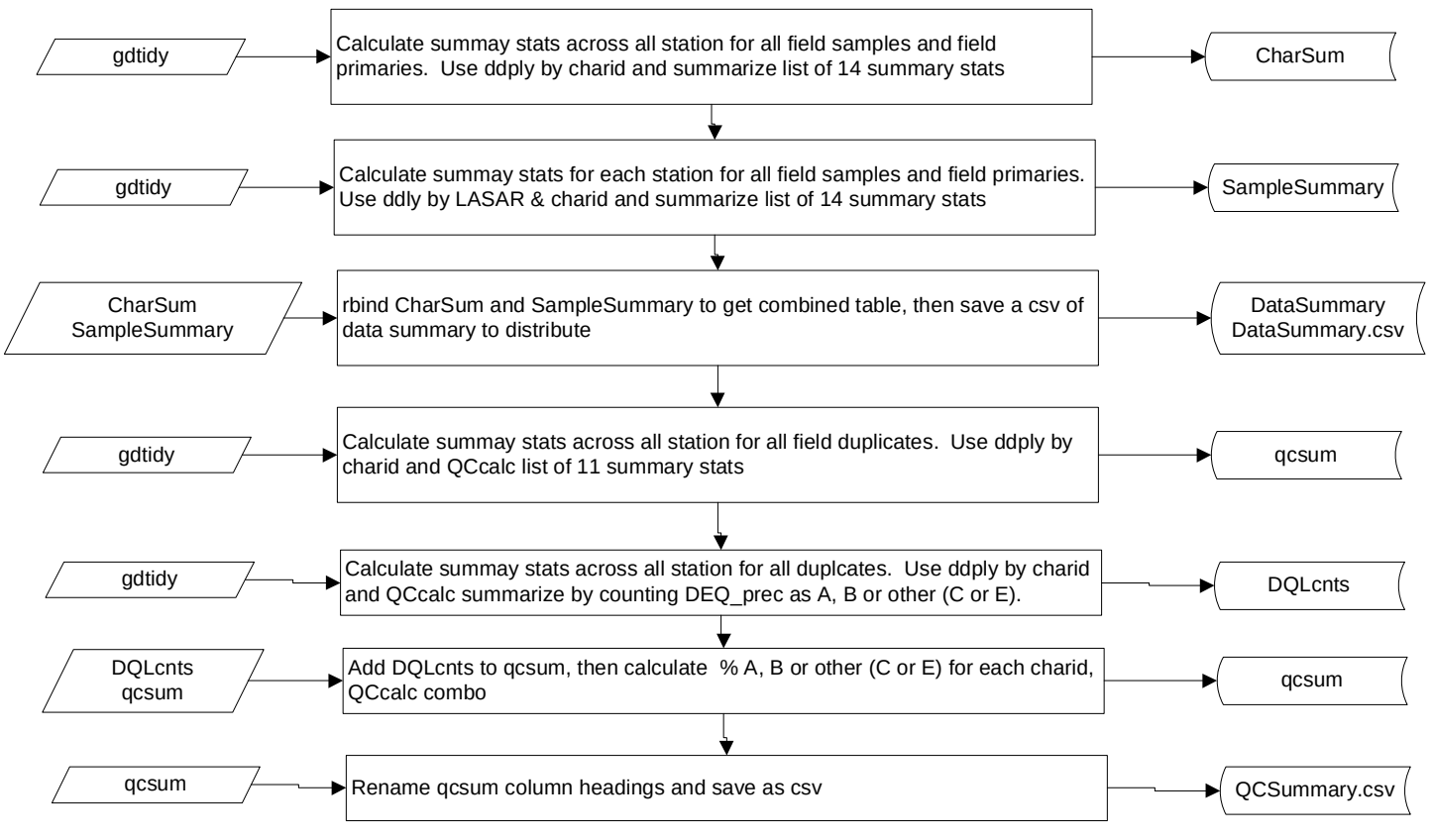
Assign gdl$mthd based on merge with charid/method combos from prj

Must be just one method per characteristic

1)Subset gdl columns only including meta data and duplicate result fields;
2) remove blanks
3) assign gdl$ActType as "FD"
4) rename headers then rbind to gdl

gdtidy

Create Data Summaries



Create Data Summary and QC Plots

Characteristic specific box plots of distributions at each station
Characteristic specific QC chart of duplicate differences vs. time
Characteristic specific plot of primary vs duplicate results

Activity and Result ID's

Generate **act** dataframe by subsetting gdtidy for activity and result relevant fields → act

Generate Activity Type codes consistent with Storet and assign to **act$AT**
1. identify characteristics which are "lab" or "field" from prj$FieldOrLab
2.for loop through act and assign AT values based on act$ActType and act$charid
- FM = field sample
- LSR = Lab sample
- FQMR,= Field duplicate primary
- LQD = Lab duplicate primary
- QFQMR= Field duplicate duplicate result
- QLQD = Lab duplicate duplicate result

→ lab, field

subid → Assign Activity ID's to **act$id** by pasting subid, act$actroot, and act$AT for each row

Assign Result ID to **act$rsltid** by pasting act$id and act$charid

Assign Activity and Result Table Fields to ID's

prj gdtidy → Merge gdtidy with subset of prj df by charid, and mthd fields to assign t.rslt$unit, MethodSpeication, AnalyticalOrganization and LOQ. → t.rlst

Run test that number of results does not change

t.rslt act → Merge t.rslt with subset of act df by actroot, LASAR, DateTime, ActType to assign t.rslt2$id (activity ID) and rsltid → t.rlst2

Run test that number of results does not change

t.rslt2 → Subset t.rslt2 for fields in t_result table in Volunteer Database and assign matching field names. Save a csv copy of the t.result in working directory. → t.result, t.result.csv

act → Group act df by activity id (act$id) → by_id

Prior to this run dcast to get the count of character id's vs activity ID's and make sure there are no duplicates.

by_id subid → Generate the **actiinfo** df summarizing by_id df to get count of id occurances, plus LASAR, DateTime, DupBatchKey, AT, actroot and item. Then add subid → actinfo

gd acinfo actorg → For loop throu each actinfo$id to get actinfo fields needed for database: site.desc, siteIDcontext, OrgComnt, DEQComnt, endDateTime, media, smplColMthd, smplColEquip, smplColEquipName, smplColComnt, smplDpth, smplDpthunit, samplers. If these are not provided in gd then assigned NA → actinfo

actinfo → Subset actinfo for fields in t_Activity in Volunteer database and rename fields to match the field names in the database. → t.Activity, t.Activity.csv

Anomaly Checks for Outliers or Suspect Data

AnomCrit prj → Subset AnomCrit and prj dataframes by different anomaly criteria: Real Range, ambient distributions, water quality standards, LOQ's. → AnomCRR, AnomCRRU, AnomAmb, AnomCwqs, AnomLOQ

Generate an empty data frame for adding result ID and anomaly type records to → t.Anomaly

Charsall, t.rslt2, CharSum, SampleSummary, AnomCRR, AnomCRRU, AnomAmb, AnomCwqs, AnomLOQ →

For each characteristic in charsall:
1. create tmpr df of results for only that characteristic from t.rslt2
2. for each result in tmpr compare to anomaly criteria if the characteristic is in the corresponding anomaly specific dataframes
   a. AnomCRR: real range both low and high (pH)
   b. AnomCRRU: real range with only an upper limit based on expected limits
   c. AnomAmb: 1,5,95, and 99th percentile ranges for ambient data, all stations, at least 3 years of data
   d. AnomCwqs: Water quality standards including two criteria for examples like temp and bacteria. Not determinations of WQ violations!
   e. AnomLOQ: results reported as less than 90% of LOQ
3. If the result does not meet the criteria then the corresponding result ID and a short description of the anomaly type are added to t.Anomaly
4. t.Anomaly is exported

→ t.Anomaly.csv

Assign Activity Group ID's and Activity to Activity Group Junction Table and add to Volunteer Database

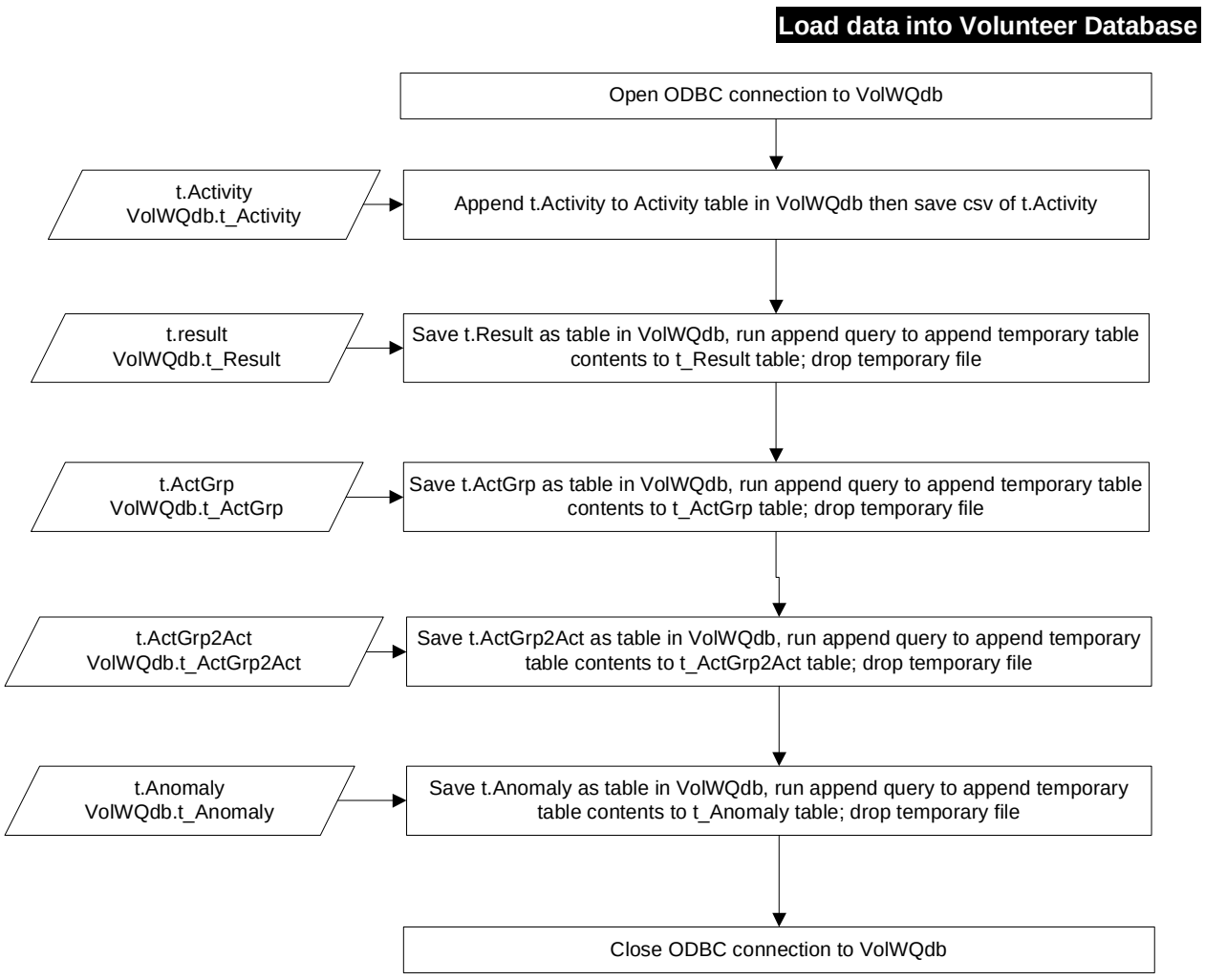
dbatch → dbatch == "Day"

- FALSE → dbatch == "Day+Crew"
- TRUE → Assign t.ActGrp$ActGrpID by subsetting the actinfo$id text string by the subid and date then pasted with the actinfo$Dupassign ActGrpType of "QC Sample" and further cut out the subid from the ActGrpID to get the date as a comment (actinfo)

dbatch == "Day+Crew"

- TRUE → Assign t.ActGrp$ActGrpID by subsetting the actinfo$id text string by the subid and date then pasted with DupBatchKey, ActID and DupBatchKey added (actinfo)
- FALSE → dbatch == "Day+Crew"
  - TRUE → Unique samplers names from actinfo$DupBatchKey are identified, placed in the AG2AComment and concatenated with submission ID for ActGrpID's (actinfo)

→ t.ActGrp, t.ActGrp.csv, t.ActGrp2Act, t.ActGrp2Act.csv


Load data into Volunteer Database
Open ODBC connection to VolWQdb
t.Activity
VolWQdb.t_Activity
Append t.Activity to Activity table in VolWQdb then save csv of t.Activity
t.result
VolWQdb.t_Result
Save t.Result as table in VolWQdb, run append query to append temporary table contents to t_Result table; drop temporary file
t.ActGrp
VolWQdb.t_ActGrp
Save t.ActGrp as table in VolWQdb, run append query to append temporary table contents to t_ActGrp table; drop temporary file
t.ActGrp2Act
VolWQdb.t_ActGrp2Act
Save t.ActGrp2Act as table in VolWQdb, run append query to append temporary table contents to t_ActGrp2Act table; drop temporary file
t.Anomaly
VolWQdb.t_Anomaly
Save t.Anomaly as table in VolWQdb, run append query to append temporary table contents to t_Anomaly table; drop temporary file
Close ODBC connection to VolWQdb