# US EPA TOXCAST DATA RELEASE ASSAY QUALITY SUMMARY OCTOBER 2014

This file describes the contents of the October 2014 ToxCast Assay Quality Summary release. The zip file contains the following assay quality statistic and summary file, not including this README file:

[1] "toxcast_assay_summary_quality_statistics_20141021.csv"
[2] "toxcast_assay_detailed_quality_statistics_20141021.csv"

In addition to the above listed files, the ToxCast program also released a MySQL dump file containing all data and a beta version of the R package (tcpl) that interacts with the MySQL database used to process all of the data for this release. For information/data not included in the listed summary files, users will need to download and interact with the MySQL database. We also encourage the database users to utilize the 'tcpl' R package containing numerous queries and functionality for easily loading and visualizing the data. At the bottom of this file is an R script to produce all of the listed files, utilizing the MySQL database and 'tcpl' R package.

All information in the summary file is reported at the assay endpoint level. The assay endpoint detailed statistics are derived from the raw concentration response data and provide assay-plate-wise statistics common to the high throughput screening community, including z-prime and ssmd (strictly standardized mean difference). The detailed file provides the median and median absolute deviation across all plates, where applicable.

```
aeid   = assay endpoint id (unique id)
assay_component_endpoint_name = name of assay endpoint
analysis_direction = the analyzed positive (upward) or negative (downward) direction
signal_direction = the direction observed of the detected signal
normalized_data_type = fold induction or percent positive control
key_positive_control = positive control used to normalize data
zprm.mdn = z-prime median across all plates (where applicable)
zprm.mad = z prime median absolute deviation (mad)
ssmd.mdn = strictly standardized mean difference median across all plates
ssmd.mad = strictly standardized mean difference mad across all plates
cv.mdn = coefficient of variation median across all plates
cv.mad = coefficient of variation mad across all plates
sn.mdn = signal-to-noise median across all plates
sn.mad = signal-to-noise mad across all plates
sb.mdn = signal-to-background median across all plates
sb.mad = signal-to-background mad across all plates
```

Many of these calculations result in NA values because there may not be plate-level details provided to us or because the analysis process precludes us from making the calculation. This initial release of the quality statistics are for general and relative reference only. Due to

the diverse assay technologies and study designs deployed, a highly generalized and robust (median and mad vs mean and sd) set of calculations were performed.

aeid = assay endpoint id (unique id)
ocnc = overall concordance among chemical replicates
calculated as the percentage of time all samples for a chemical were either negative or positive (e.g., 0 out of 3 or 3 out of 3) over the total number of chemicals with replicates.
hcnc = hit concordance among chemical replicates
calculated as the percentage of time all samples for a chemical were positive (e.g., 3 out of 3) over the total number of chemicals with any replicate being positive (e.g., 1 out of 3 or 2 out of 3).
**It should be noted that most of these chemical replicates were separately procured and that these concordance values are highly influenced by the number of replicates.*
aenm = assay endpoint name (i.e., assay_component_endpoint_name)
resp_unit = response unit (fold induction or percent activity)
bmad = baseline median absolute deviation for the assay (based on the response values at the 2 lowest tested concentrations)
nconc = nominal number of tested concentrations
coff = the response cutoff used to derive the hit calls (e.g., 5*bmad, 10*bmad)
test = total number of samples tested
acnt = number of active samples
apct = percent active samples
icnt = number of inactive samples
ipct = percent of inactive samples
ncnt = number of samples that could not be modeled (e.g., having less than 4 concs)
npct = percent not modeled
mmed = maximum observed response across the assay
cmax = target (nominal) maximal tested concentration
cmin = target (nominal) minimal tested concentration
mtop = maximum modeled response across the assay (max top of curve)
nrep = target (nominal) number of replicates
npts = target (nominal) number of points (nconc * nrep)
cnst = percent constant model winner (based on having lowest AIC value)
hill = percent hill model winner (based on having lowest AIC value)
gnls = percent gain-loss model winner (based on having lowest AIC value)
rmse = median root mean squared error across all winning models

The summary quality statistics file provides a nice overview of the target study design for each assay endpoint as well as summary statistics around active prevalence and hit-calling criteria.

For questions or concerns, please contact Monica Linnenbrink at: linnenbrink.monica@epa.gov.

```
#########################################################################
## R Script to produce October 2014 ToxCast Tox21 Data Release
#########################################################################

rm(list = ls())
library(tcpl)
library(data.table)

#DETAILED QUALITY STATS FOR OUTPUT

#LARGE QUERY: can wrap and run by aeid (assay endpoint)
query <-
"
SELECT level3.aeid, level0.l0id, level0.acid, level0.spid, level0.cpid,
level0.apid, level0.rowi, level0.coli, level0.wllt, level0.wllq, level0.conc,
level0.rval, level0.srcf, level1.cndx, level1.repi, level2.cval, level3.bval,
level3.pval, level3.logc, level3.resp, assay_component_endpoint_name,
assay_component_endpoint_desc, assay_function_type, normalized_data_type,
analysis_direction, burst_assay, key_positive_control, signal_direction,
intended_target_type, intended_target_type_sub, intended_target_family,
intended_target_family_sub
FROM (((level0 INNER JOIN level1 ON level0.l0id = level1.l0id) INNER JOIN
level2 ON (level1.l1id = level2.l1id) AND (level0.l0id = level2.l0id)) INNER
JOIN level3 ON (level2.l2id = level3.l2id) AND (level1.l1id = level3.l1id)
AND (level0.l0id = level3.l0id)) INNER JOIN assay_component_endpoint ON
level3.aeid = assay_component_endpoint.aeid
"

dat <- tcplQuery(query = query, db = options()$TCPL_DATA)

dato <- dat

dat[ , bval := median(cval[(cndx %in% 1:2 & wllt == "t") | wllt == "n"],
na.rm = TRUE),
       by = list(aeid, apid)]
dat[ , bval.mad := mad(cval[(cndx %in% 1:2 & wllt == "t") | wllt == "n"],
na.rm = TRUE),
       by = list(aeid, apid)]
dat[wllt %in% c('p','v','m') , tval := median(rval, na.rm = TRUE),
                                 by = list(aeid, apid, wllt, cndx)]
dat[wllt %in% c('p','v','m') , tval.mad := mad(tval, na.rm = TRUE),
                                 by = list(aeid, apid, wllt, cndx)]
dat[ , tval.min := min(tval, na.rm = TRUE),
                                 by = list(aeid, apid)]
dat[ , tval.max := max(tval, na.rm = TRUE),
                                 by = list(aeid, apid)]
  # finds corresponding mad value (only take min to assure single value)
dat[ , tval.mad.min := min(tval.mad[tval.min == tval], na.rm = TRUE),
                       by = list(aeid, apid)]
  # finds corresponding mad value (only take min to assure single value)
dat[ , tval.mad.max := min(tval.mad[tval.max == tval], na.rm = TRUE),
                       by = list(aeid, apid)]
dat[signal_direction == 'gain', pval := tval.max]
```

```
dat[signal_direction == 'loss', pval := tval.min]
dat[signal_direction == 'gain', pval.mad := tval.mad.max]
dat[signal_direction == 'loss', pval.mad := tval.mad.min]

agg <- unique(dat[ , list(assay_component_endpoint_name, export_ready,
                          analysis_direction, signal_direction,
                          normalized_data_type, key_positive_control,
                          aeid, apid,
                          bval, bval.mad, pval, pval.mad)])


agg[ , zprm := 1 - ((3 * (pval.mad + bval.mad)) / abs(pval - bval))] #Robust
z-prime calculation
agg[ , ssmd := (pval.mad - bval.mad) / sqrt( pval^2 + bval^2 )] # Robust SSMD
calculation
agg[ , cv := bval.mad/bval]
agg[ , sn := (pval - bval)/bval.mad]
agg[ , sb := pval/bval]

agg[ , zprm.mdn := median(zprm, na.rm = TRUE), by = aeid]
agg[ , zprm.mad := mad(zprm, na.rm = TRUE), by = aeid]
agg[ , ssmd.mdn := median(ssmd, na.rm = TRUE), by = aeid]
agg[ , ssmd.mad := mad(ssmd, na.rm = TRUE), by = aeid]
agg[ , cv.mdn := median(cv, na.rm = TRUE), by = aeid]
agg[ , cv.mad := mad(cv, na.rm = TRUE), by = aeid]
agg[ , sn.mdn := median(sn, na.rm = TRUE), by = aeid]
agg[ , sn.mad := mad(sn, na.rm = TRUE), by = aeid]
agg[ , sb.mdn := median(sb, na.rm = TRUE), by = aeid]
agg[ , sb.mad := mad(sb, na.rm = TRUE), by = aeid]

out <- unique(agg[ , list(aeid, assay_component_endpoint_name, export_ready,
                          analysis_direction, signal_direction,
                          normalized_data_type, key_positive_control,
                          zprm.mdn, zprm.mad, ssmd.mdn, ssmd.mad,
                          cv.mdn, cv.mad, sn.mdn, sn.mad, sb.mdn, sb.mad)])

setkeyv(out,'assay_component_endpoint_name')
write.csv(out,"toxcast_assay_detailed_quality_statistics_20141021.csv")

##### SUMMARY STATS FOR OUTPUT

dat <- tcplLoadData(5L)
dat <- tcplPrepOtpt(dat)

agg <- dat[ , list(
  bmad  = max(bmad, na.rm = TRUE),    #baseline median absolute deviation
(mad around the first 2 tested concentrations
  nconc = as.double(median(nconc, na.rm = TRUE)), #nominal number of
concentrations tested for the assay endpoint
  coff  = max(coff, na.rm = TRUE),  #global response cutoff established for
the assay (methods available within pipeline)
  test  = .N, #total number of samples tested in concentration response
  acnt  = as.double(lw(hitc==1)),  # active count
```

```
  apct  = lw(hitc==1)/.N,  # active percentage
  icnt  = as.double(lw(hitc==0)),  #inactive count
  ipct  = lw(hitc==0)/.N,  #inactive percentage
  ncnt  = as.double(lw(hitc==-1)), # could not model count (<=3
concentrations with viable data)
  npct  = lw(hitc==-1)/.N, # could not model percentage
  mmed  = max(max_med, na.rm = TRUE), # maximum response (median at any given
concentration) across entire assay endpoint
  cmax  = 10^median(logc_max, na.rm = TRUE), # nominal maximum tested
concentration (target concentration)
  cmin  = 10^median(logc_min, na.rm = TRUE), # nominal minimum tested
concentration (target concentration)
  mtop  = max(modl_tp, na.rm = TRUE), # maximum modeled response (top of
curve) across entire assay endpoint
  nrep  = as.double(median(nrep, na.rm = TRUE)), # nominal number of
replicates per sample (target number of replicates)
  npts  = as.double(median(npts, na.rm = TRUE)), # nominal number of data
points per sample
  cnst  = lw(modl=='cnst')/.N, # percentage of sample-assayendpoints where
the constant model won (may not all be 'actives')
  hill  = lw(modl=='hill')/.N, # percentage of sample-assayendpoints where
the hill model won (may not all be 'actives')
  gnls  = lw(modl=='gnls')/.N, # percentage of sample-assayendpoints where
the gain-loss model won (may not all be 'actives')
  rmse  = median(modl_rmse, na.rm = TRUE) # median root mean squared error
across all model winners for an assay endpoint
  ), by = list(aeid, aenm, resp_unit)]

setkeyv(agg,"aenm")


agg2 <- dat[hitc >= 0 ,
                list(n = .N,
                     acnt = sum(hitc)
                     )
                , by = list(aeid, chid)]
agg3 <- agg2[n > 1, list(
                          ocnc = lw(acnt==n | acnt==0)/.N,  # overall
concordance among chemical-replicates
                          hcnc = lw(acnt==n)/lw(acnt>0)  # hit concordance
among chemical-replicates
                          # (may be samples from different sources)
                          ), by = aeid]
setkey(agg3,"aeid")
setkey(agg, "aeid")
agg <- agg3[agg]


write.csv(agg,"toxcast_assay_summary_quality_statistics_20141021.csv")
######################################################################
## End R script
######################################################################
```