

COMPASS: TEMPEST Discrete DOC Data QAQC

June & July 2024

2025-06-23

Run Information

```
#identify which section you are in
cat("Run Information")

## Run Information

#a link to the Gitbook or whatever protocol you are using for this analysis
#steph will add this soon

#anything that needs to be changed do this in the first chunk
Date_Run = "09/09/24"
Run_by = "Stephanie J. Wilson"
Script_run_by = "Stephanie J. Wilson"
run_notes = "The TN curves are not good, samples were re-analyzed on the TOC
             with a good curve from another run and then the TN analyzer was serviced."

#file path and name for summary file
raw_file_name = "tmp_doc_raw_data_2024/TMP_202406_07.txt"
#file path and name for the all peaks file
raw_allpeaks_name = "tmp_doc_raw_data_2024/TMP_202406_07_allpeaks.txt"
#file path and name for processed data after QAQC
processed_file_name = "tmp_doc_processed_data_2024/TMP_PW_DOC_Processed_202406_07.csv"

#check standard concentrations - Update if running different checks:
chk_std_c = 1
chk_std_n = 1

#Log path
Log_path = "tmp_doc_raw_data_2024/COMPASS_TMP_TOCTN_QAQClog_2024.csv"
```

Setup

Pull in active porewater tracking inventory sheet

```
## File already exists. No download needed.
```

Import Data Functions

Import Sample Data

```
## Import Sample Data

## New names:
## * ' ' -> '...14'

## # A tibble: 3 x 4
##   sample_name      npoc_raw tdn_raw run_datetime
##   <chr>          <dbl>   <dbl> <chr>
## 1 TMP_SW_B4_20240624    4.97    6.70 7/19/2024 1:59:07 AM
## 2 TMP_FW_C6_20240624    7.71   16.2 7/19/2024 2:25:13 AM
## 3 TMP_SW_F6_20240715    5.32    0.121 7/19/2024 2:48:20 AM
```

Assessing standard Curves

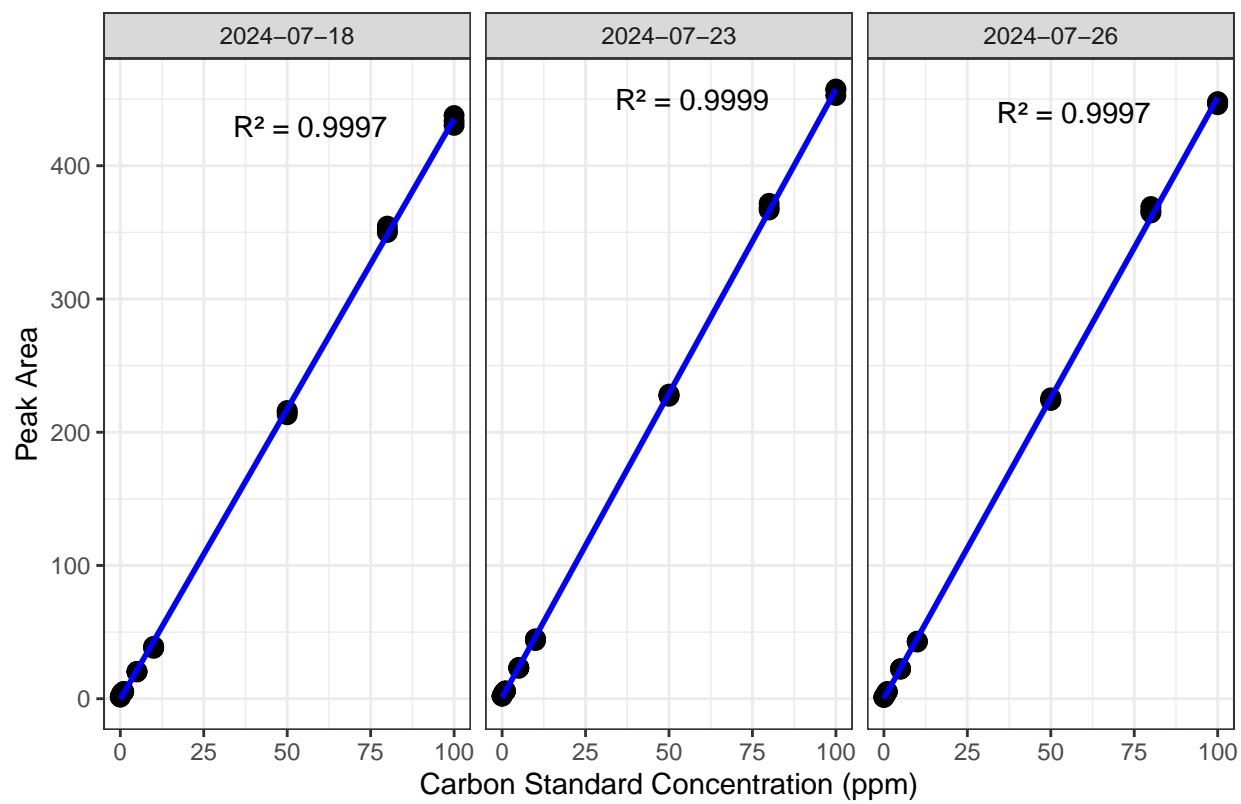
```
## Assess the Standard Curve
```

```
## New names:
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

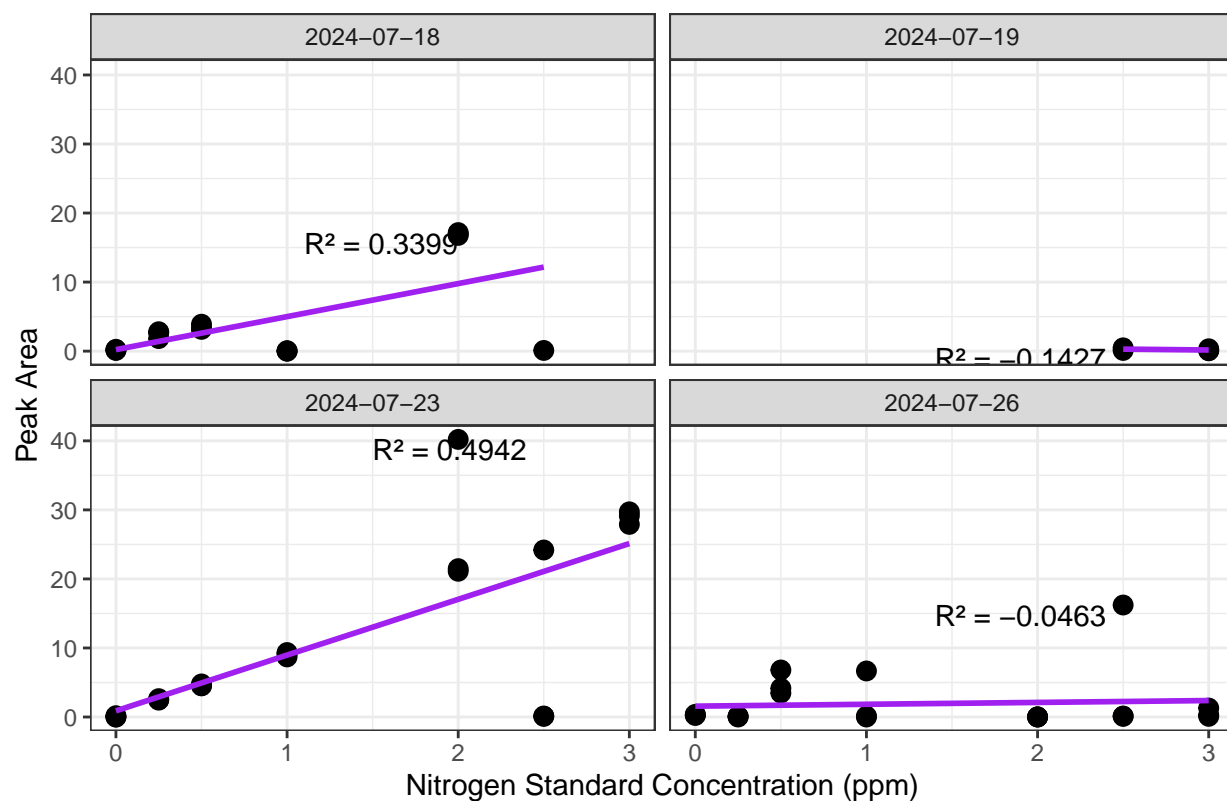
```
## * '' -> '...18'
```

NPOC Std Curve by Date



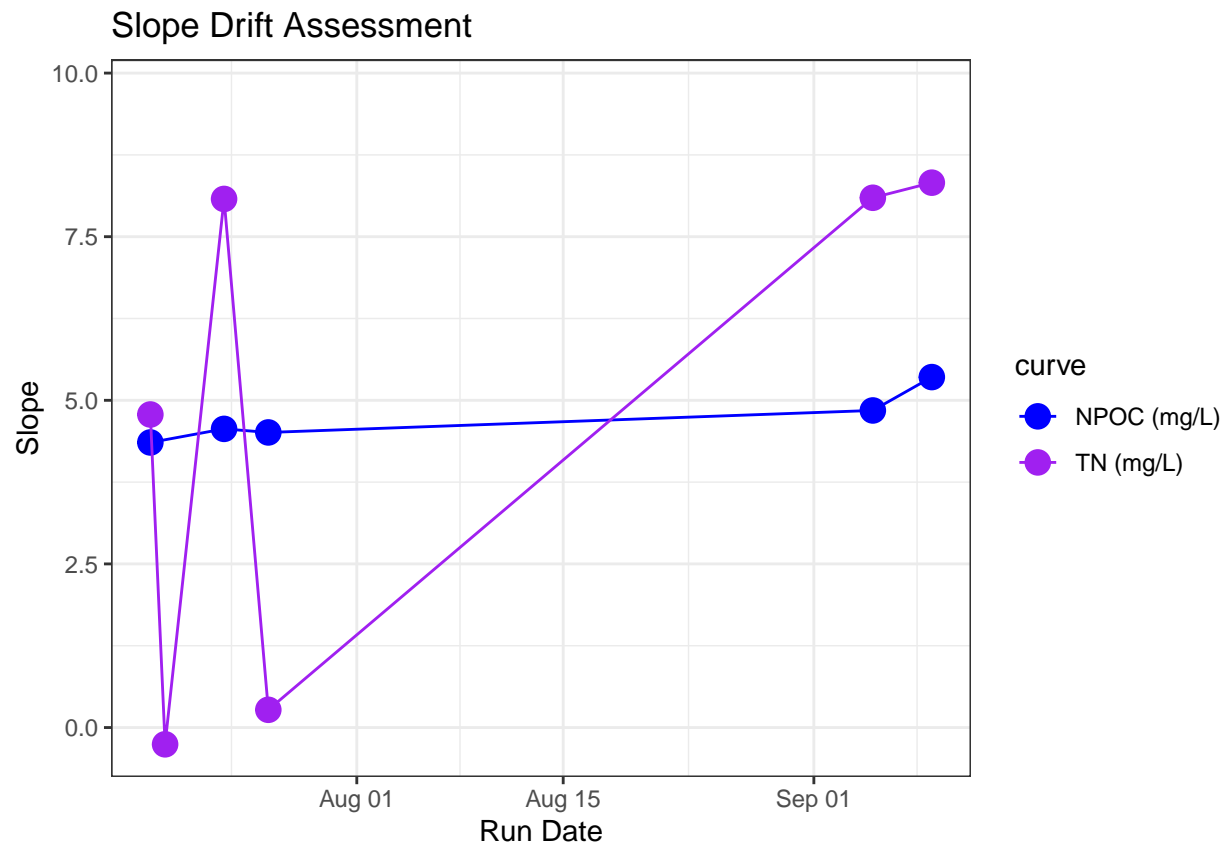
```
## 'geom_smooth()' using formula = 'y ~ x'
```

TN Std Curve by Date



```
## Warning: Removed 15 rows containing missing values or values outside the scale range
## ('geom_point()').
```

```
## Warning: Removed 15 rows containing missing values or values outside the scale range
## ('geom_line()').
```



```
## [1] "NPOC Curve r2 GOOD"
```

```
## [1] "TN Curve r2 is below cutoff! - REASSESS"
```

Assess Check Standards

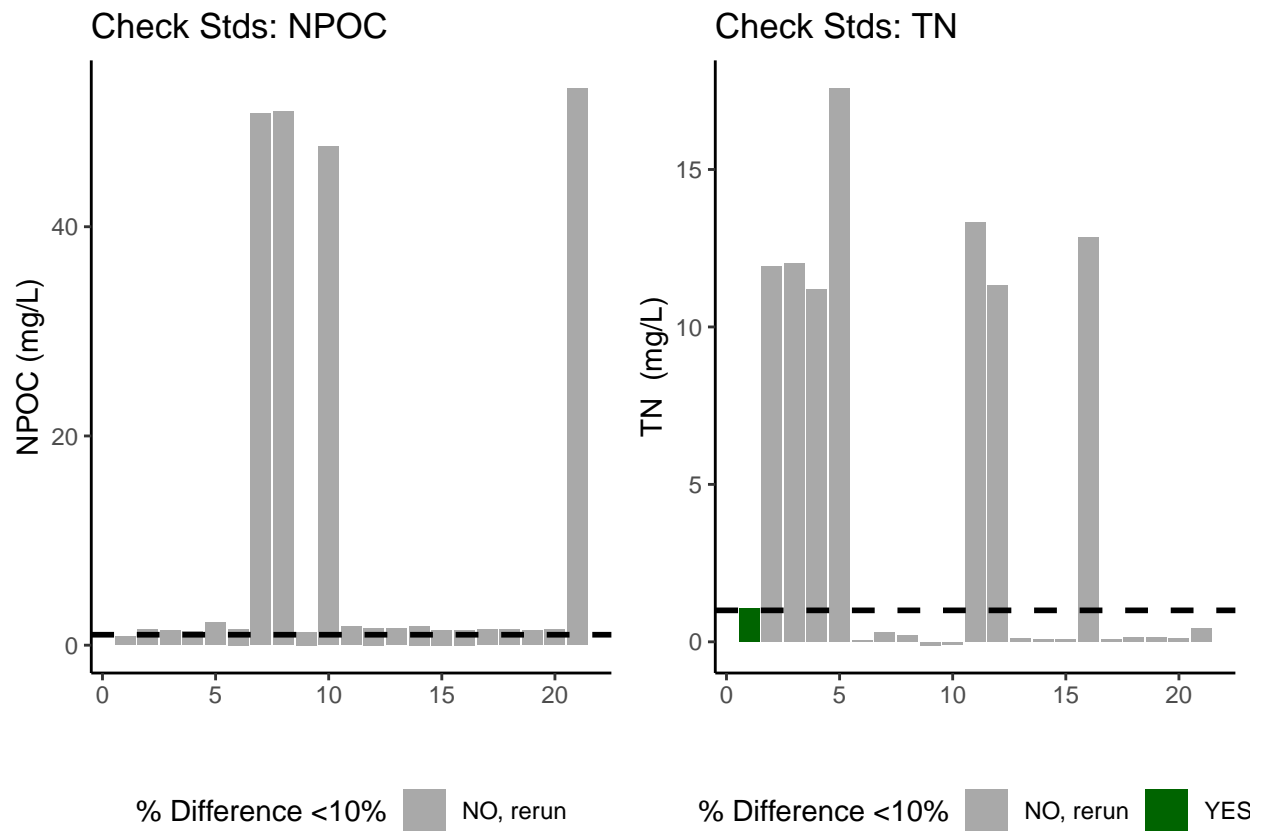
```
## Assess the Check Standards
```

```
## New names:
```

```
## * ' ' -> '...14'
```

```
## [1] "Carbon CHECK STANDARD RSD TOO HIGH - REASSESS"
```

```
## [1] "Nitrogen CHECK STANDARD RSD TOO HIGH - REASSESS"
```



```
## [1] "<60% of Carbon Check Standards are within range of the expected concentration - REASSESS"
```

```
## [1] "<60% of Nitrogen Check Standards are within range of the expected concentration - REASSESS"
```

Assess Blanks

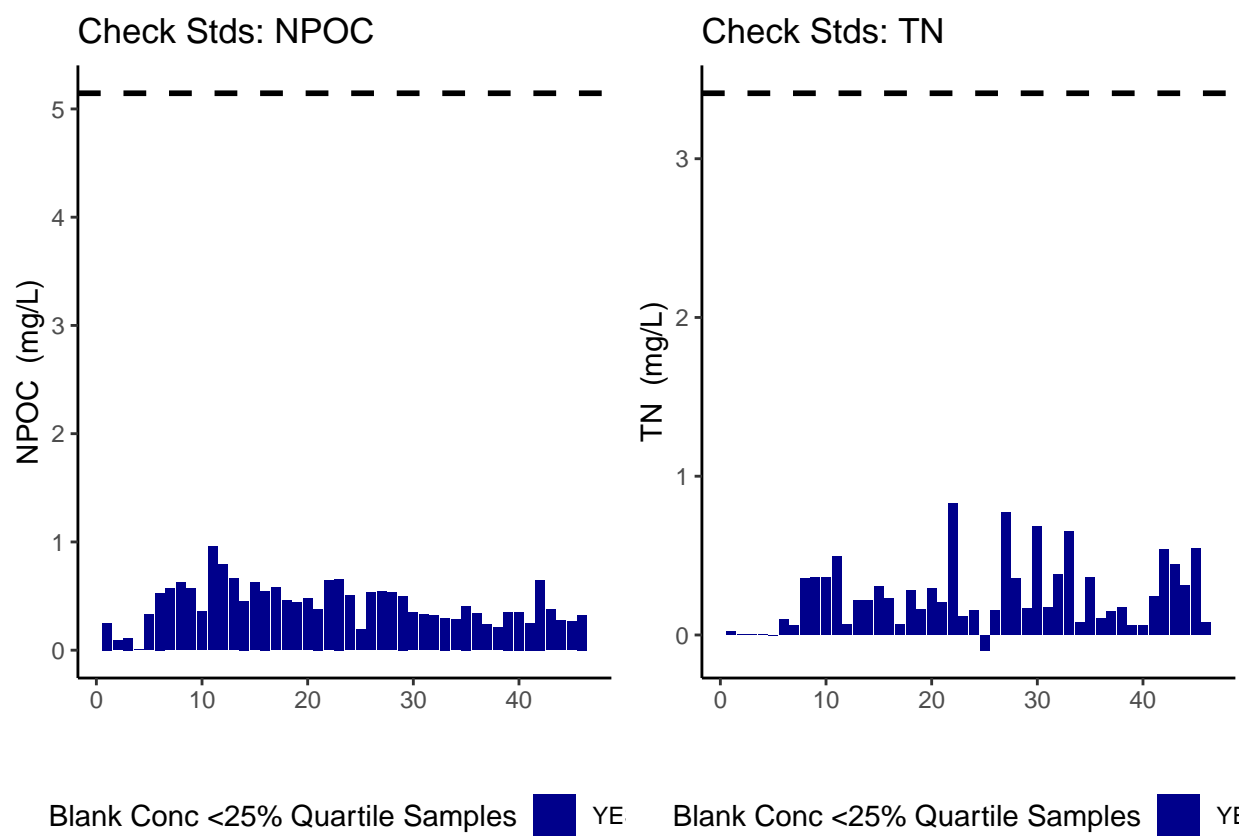
```
## Assess Blanks
```

```
## New names:
```

```
## * ' ' -> '...14'
```

```
## [1] ">60% of Carbon Blank concentrations are below the lower 25% quartile of samples"
```

```
## [1] ">60% of Nitrogen Blank concentrations are below the lower 25% quartile of samples"
```



```
## carbon blanks:
```

```
## [1] 0.4249826
```

```
## nitrogen blanks:
```

```
## [1] 0.245943
```

Assess Duplicates - if there are any

```

“{#r Check Duplicates, echo=FALSE}
cat("Assess Duplicates")
#Take a look at the raw data #head(dat_raw)

#pull out any rows that have "dup" in the sample_name column dups <- dat_raw %>%
select(!c(npoc_flag, tdn_flag)) %>% filter(str_detect(sample_name, "dup")) #have to change this to match
data

#create a new dataframe and remove dups from sample dataframe dat_raw2 <- dat_raw %>%
filter(!str_detect(sample_name, "dup"))

#remove the dup from these IDs so we will have duplicate sample names dupssample_name <-
gsub("dup", "", as.character(dupssample_name)) dups <- dups[ ,c(4)] #remove the run date time
for colnames(dups) <- c('sample_name', 'npoc_raw_dup', 'tdn_raw_dup') head(dups)

QAdups <- merge(dat_raw2, dups) head(QAdups)

df2 <- as.data.frame(QAdupsnpoc_raw) df2dups <- QAdups$npoc_raw_dup
df2sds <- apply(df2, 1, sd) df2mean <- apply(df2, 1, mean)

QAdupsnpoc_dups_cv <- (df2sds/df2mean)*100 QAdupsnpoc_dups_cv_flag <- ifelse(QAdups$npoc_dups_cv
<10, 'YES', 'NO, rerun')

df3 <- as.data.frame(QAdupstdn_raw) df3dups <- QAdups$tdn_raw_dup
df3sds <- apply(df3, 1, sd) df3mean <- apply(df3, 1, mean)

QAdupstdn_dups_cv <- (df3sds/df3mean) * 100 QAdupstdn_dups_cv_flag <- ifelse(QAdups$tdn_dups_cv
<10, 'YES', 'NO, rerun')

head(QAdups)

#plot dups output as a bar graph to easily check - want any over 10% to be red need to work on this
C_dups <- ggplot(data = QAdups, aes(x = sample_name, y = npoc_dups_cv, fill = npoc_dups_cv_flag))
+ geom_bar(stat = 'identity') + theme_classic() + labs(x = "Sample ID", y = "CV of NPOC
Dups (%)") + scale_fill_manual(values = c("YES" = "darkgreen", "NO, rerun" = "red")) +
theme(legend.position = "none") + geom_hline(yintercept = 10, linetype = "dashed", color = "black", size = 1)
+ guides(fill = guide_legend(title = "CV Between Dups <10%")) + theme(axis.text.x = element_text(angle
= 90, hjust = 0.5))

N_dups <- ggplot(data = QAdups, aes(x = sample_name, y = tdn_dups_cv, fill = tdn_dups_cv_flag)) +
geom_bar(stat = 'identity') + theme_classic() + labs(x = "Sample ID", y = "CV of TN Dups (%)") +
scale_fill_manual(values = c("YES" = "darkgreen", "NO, rerun" = "red")) + theme(legend.position = "none")
+ geom_hline(yintercept = 10, linetype = "dashed", color = "black", size = 1) + guides(fill = guide_legend(title = "CV
Between Dups <10%")) + theme(axis.text.x = element_text(angle = 90, hjust = 0.5))

ggarrange(C_dups, N_dups, ncol = 2, nrow = 1)

#calculate the percent of check standards that are within the range based on the flag c_dups_percent <-
(sum(QAdupsnpoc_dups_cv_flag == "YES")/nrow(QAdups))*100 n_dups_percent <- (sum(QAdupstdn_dups_cv_flag
== "YES")/nrow(QAdups))*100

#report out if flags indicate need for rerun ifelse(c_dups_percent >= chks_flag, ">60% of Car-
bon Duplicates have a CV <10%", "<60% of Carbon Duplicates have a CB <10% - REASSESS")
ifelse(n_dups_percent >= chks_flag, ">60% of Nitrogen Duplicates have a CV <10%", "<60% of Nitrogen
Duplicates have a CB <10% - REASSESS")

#write out a flag to the sample dataframe if more than 60% of the dups have CVs out of range if
(c_dups_percent <= chks_flag) { dat_rawnpoc_flag <- ifelse(dat_rawnpoc_flag != "", paste0(dat_raw$npoc_flag, ";
NPOC dups out of range"), "NPOC dups out of range" ) }

```



```
if (n_dups_percent <= chks_flag) { # assuming you have tn_chks_percent similarly dat_rawtdn_flag <
  -ifelse(dat_rawtdn_flag != "", paste0(dat_raw$tdn_flag, "TN dups out of range"), "TN dups out of range"
) }
```

```
\newpage
```

```
## Sample Flagging
```

Sample Flagging

```
<!-- --> 
```

```
\newpage
```

```
## Visualize Data by Plot
```

Visualize Data

Site_Code Plot Grid_Square Date

1 TMP SW B4 20240624

2 TMP FW C6 20240624

3 TMP SW F6 20240715

Site_Code Plot Grid_Square Date sample_name npoc_raw tdn_raw

1 TMP SW B4 20240624 TMP_SW_B4_20240624 4.969 6.704

2 TMP FW C6 20240624 TMP_FW_C6_20240624 7.708 16.180

3 TMP SW F6 20240715 TMP_SW_F6_20240715 5.321 0.121

run_datetime npoc_flag

1 7/19/2024 1:59:07 AM NPOC checks out of range

2 7/19/2024 2:25:13 AM NPOC checks out of range

3 7/19/2024 2:48:20 AM NPOC checks out of range

tdn_flag

1 TN r2 low; TN checks out of range; value above cal curve

2 TN r2 low; TN checks out of range; value above cal curve

3 TN r2 low; TN checks out of range; blank is > 25% of sample value

<!-- --> ![](tmp

\newpage

Convert data from mg/L to uMoles/L

Add in/check metadata

Check Sample IDs with Metadata

A tibble: 3 x 2

sample_name metadata_recorded

1 TMP_SW_B4_20240624 TRUE

2 TMP_FW_C6_20240624 TRUE

3 TMP_SW_F6_20240715 TRUE

```
## Export Processed Data
```

Export Processed Data

```
# A tibble: 3 x 21
```

```
Project plot grid Depth_cm sample_type Vial_ID date npoc_mgL npoc_uM
```

```
1 COMPASS: TEMP~ SW B4 15 DOC SW_B4_~ 2024~ 4.97 414.
```

```
2 COMPASS: TEMP~ FW C6 15 DOC FW_C6_~ 2024~ 7.71 642.
```

```
3 COMPASS: TEMP~ SW F6 15 DOC SW_F6_~ 2024~ 5.32 443.
```

```
# i 12 more variables: npoc_flag , tdn_mgL , tdn_uM ,
```

```
# tdn_flag , Analysis_runtime , Run_notes ,
```

```
# Evacuation_date_YYMMDD , Collection_Date_YYYYMMDD ,
```

```
# Collection_Start_Time_24hrs , Collection_End_Time_24hrs ,
```

```
# EST_EDT , Volume_mL
```

```
““
```

```
#end
```