COMPASS_TEMPEST_SGW_2025: Well Test

Stephanie J. Wilson

2025-02-11

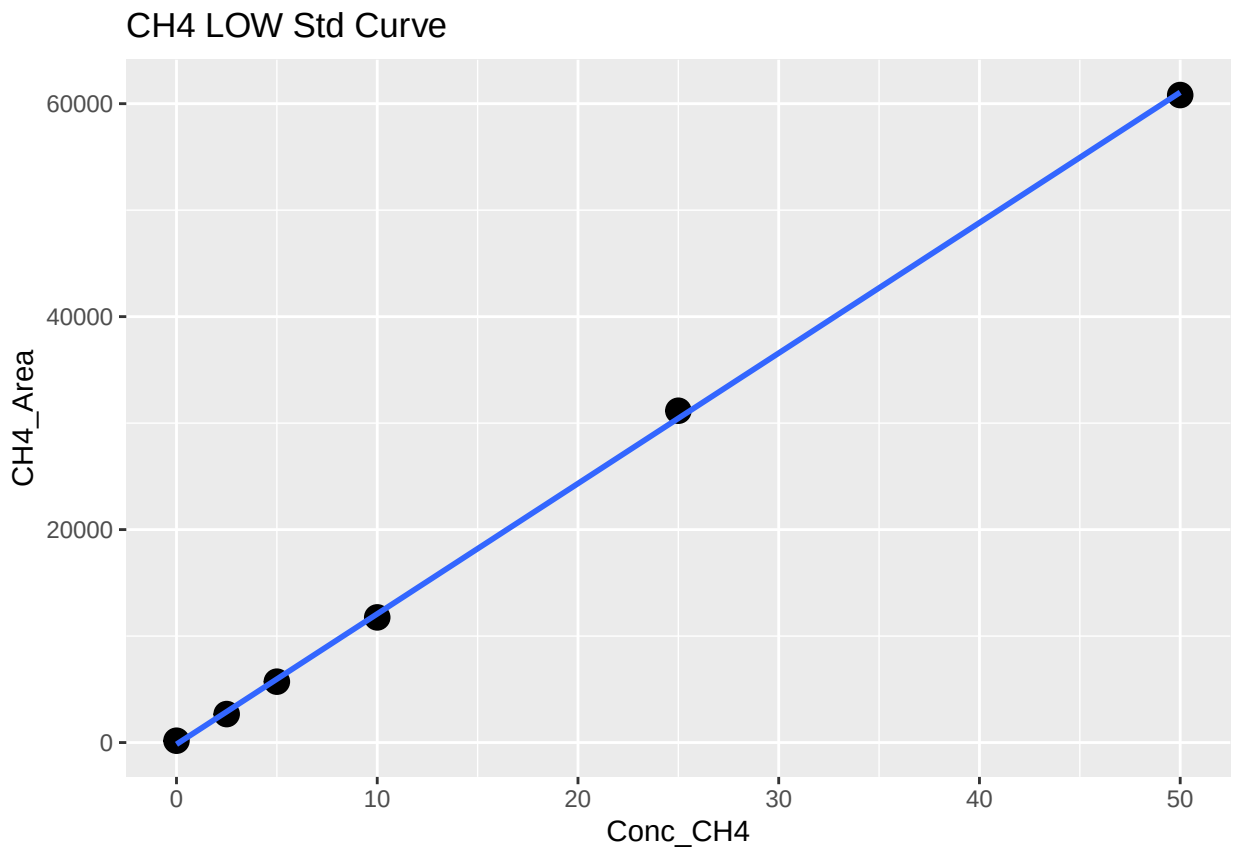
Contents

##Set Up

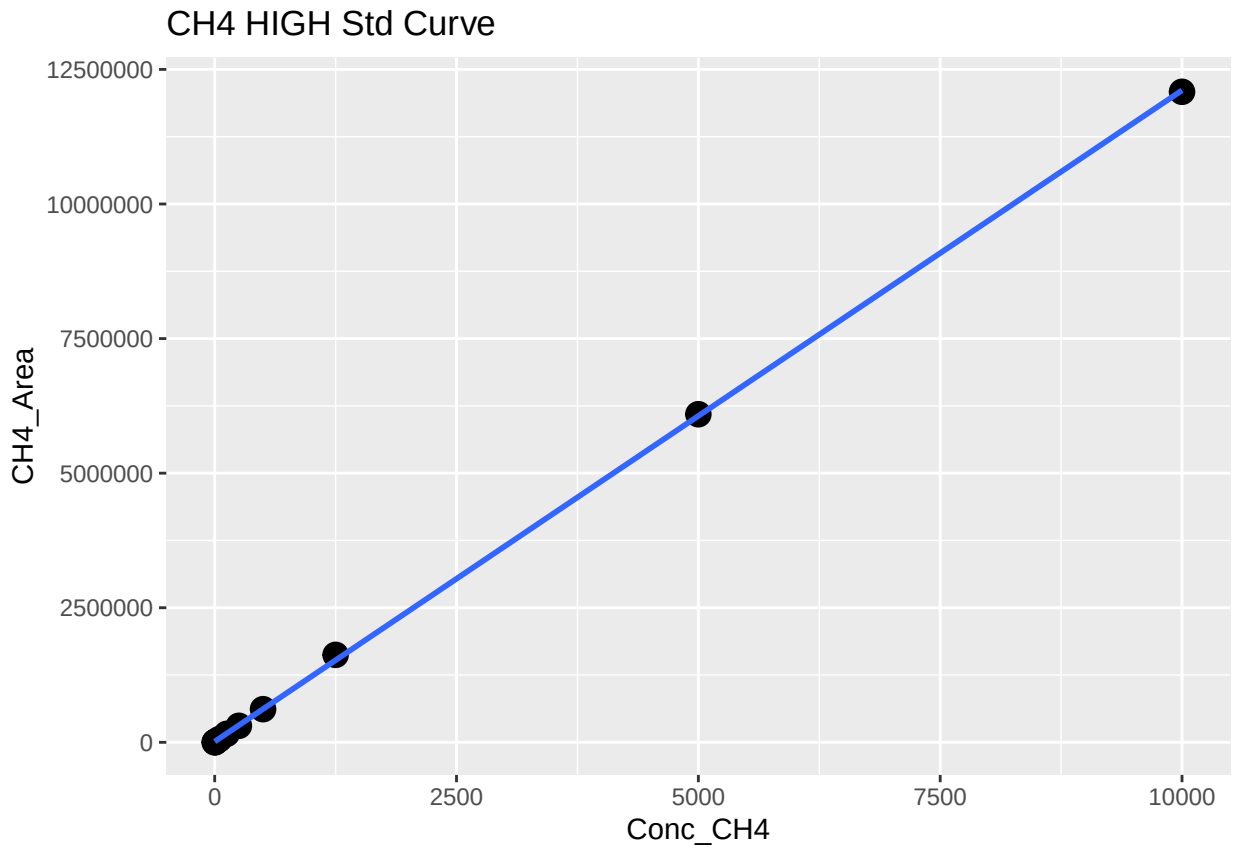
0.1 Read in first data file and assess standard curves

```
##       Machine       User Run_Date Sample_Year Sample_Month Sample_Day
## 1 Shimadzu GC B. Blakley 20250922          NA         <NA>         NA
## 2 Shimadzu GC B. Blakley 20250922          NA         <NA>         NA
## 3 Shimadzu GC B. Blakley 20250922          NA         <NA>         NA
## 4 Shimadzu GC B. Blakley 20250922          NA         <NA>         NA
## 5 Shimadzu GC B. Blakley 20250922          NA         <NA>         NA
## 6 Shimadzu GC B. Blakley 20250922          NA         <NA>         NA
##   Event_Stamp Sample_Time Time_Zone  G_W Sample_ID Sample_Type Conc_CO2
## 1        <NA>          NA      <NA> <NA>    LabAir     Lab air       NA
## 2        <NA>          NA      <NA> <NA>    Blank1       Blank       NA
## 3        <NA>          NA      <NA> <NA>    Blank2       Blank       NA
## 4        <NA>          NA      <NA> <NA>      0ppm    Standard       NA
## 5        <NA>          NA      <NA> <NA>    2.5ppm    Standard       NA
## 6        <NA>          NA      <NA> <NA>      5ppm    Standard       NA
##   Conc_CH4 Conc_N2O CO2_Area CH4_Area Dilution_Factor
## 1       NA       NA       NA     1277              NA
## 2       NA       NA       NA      164              NA
## 3       NA       NA       NA      166              NA
## 4      0.0       NA       NA      172              NA
## 5      2.5       NA       NA     2672              NA
## 6      5.0       NA       NA     5716              NA
```

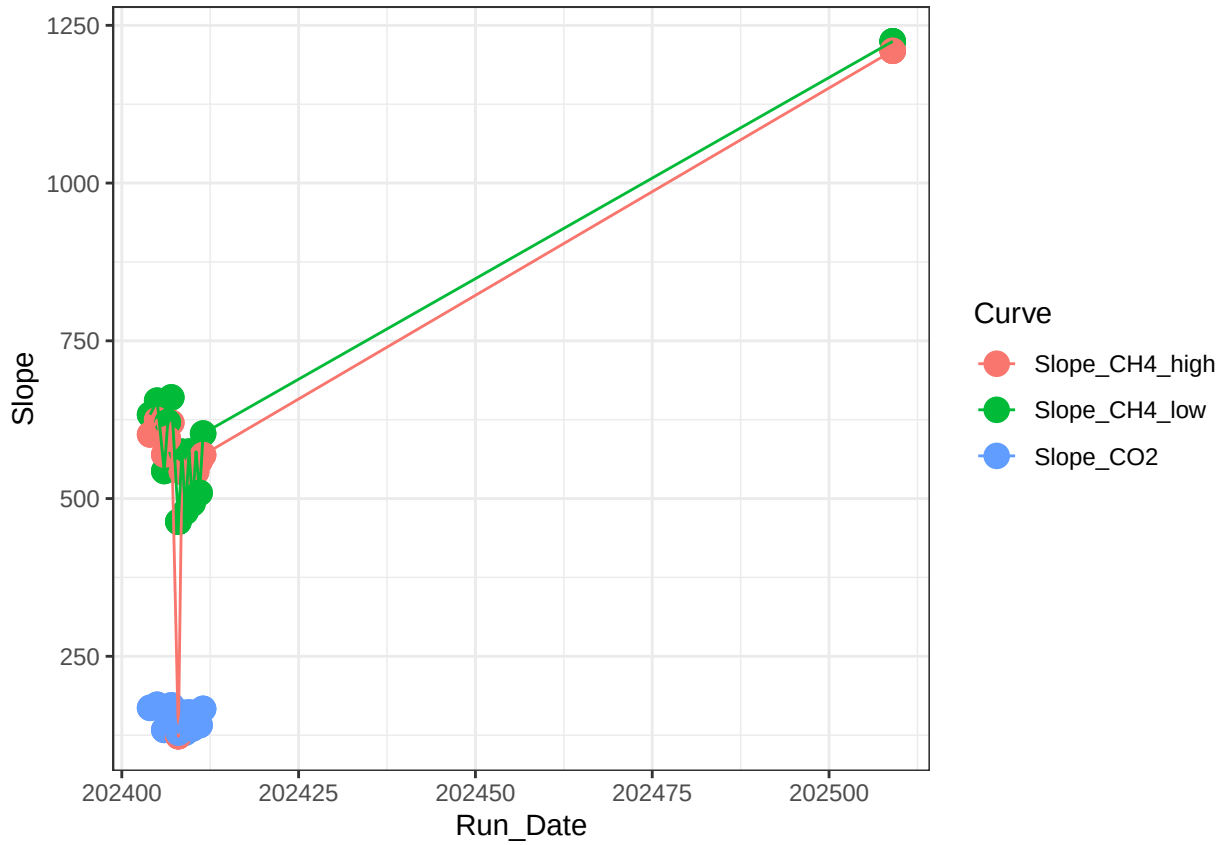
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
##
## Call:
## lm(formula = stds_ch4_low$CH4_Area ~ stds_ch4_low$Conc_CH4)
##
## Residuals:
##      1      2      3      4      5      6
##  337.1 -224.4 -241.9 -327.9  712.0 -255.0
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -165.1      258.6  -0.639    0.558
## stds_ch4_low$Conc_CH4   1224.6       11.1 110.319 4.05e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 474.9 on 4 degrees of freedom
## Multiple R-squared:  0.9997, Adjusted R-squared:  0.9996
## F-statistic: 1.217e+04 on 1 and 4 DF,  p-value: 4.049e-08

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




```
##
## Call:
## lm(formula = stds_ch4$CH4_Area ~ stds_ch4$Conc_CH4)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -27445 -14156 -13201  -6134  98453
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       13820.463  11476.617   1.204    0.256
## stds_ch4$Conc_CH4  1209.769      3.529 342.782   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35680 on 10 degrees of freedom
## Multiple R-squared:  0.9999, Adjusted R-squared:  0.9999
## F-statistic: 1.175e+05 on 1 and 10 DF,  p-value: < 2.2e-16

##   X          Curve        R2    Slope  Intercept Run_Date
## 1 1  Slope_CH4_low 0.9996407 633.0314  -320.6965   202404
## 2 2 Slope_CH4_high 0.9993443 601.5512 14817.1191   202404
## 3 3      Slope_CO2 0.9999907 168.3200 10075.2183   202404
## 4 4  Slope_CH4_low 0.9996407 633.0314  -320.6965   202404
## 5 5 Slope_CH4_high 0.9993443 601.5512 14817.1191   202404
## 6 6      Slope_CO2 0.9999907 168.3200 10075.2183   202404
```

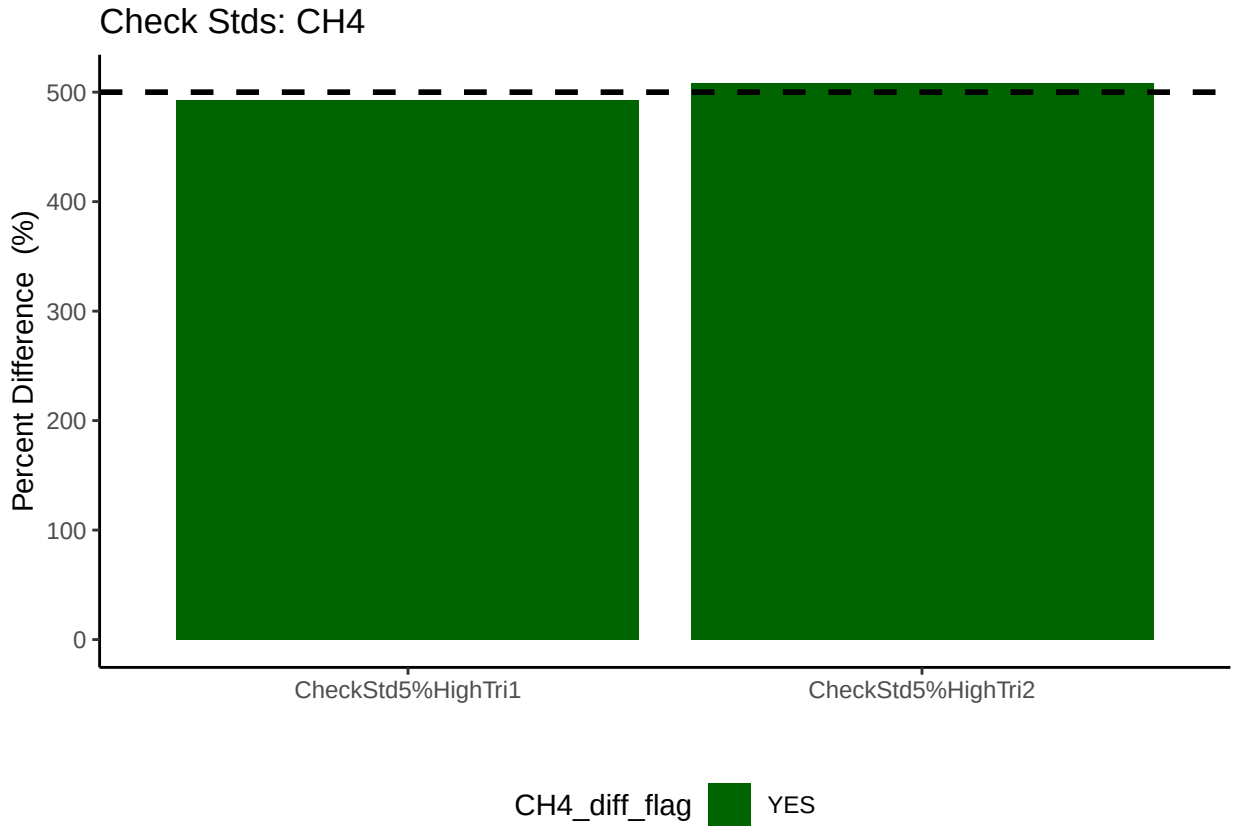


0.2 Now calculate the CH4 & CO2 concentrations in ppm

```
#head(raw)

#pull out methane standards
Samples <-  raw %>%
  filter(!str_detect(Sample_Type, "STD_CH4")) %>%
  filter(!str_detect(Sample_Type, "STD_CO2")) %>%
  filter(!str_detect(Sample_Type, "Blank")) %>%
  filter(!str_detect(Sample_Type, "Chk_STD")) %>%
   filter(!str_detect(Sample_Type, "CHKSTD")) %>%
  filter(!str_detect(Sample_Type, "NA"))
#head(Samples)

#Now flag any areas that are above the 100ppm area for CH4
Samples$CH4_Curve <- ifelse(Samples$CH4_Area >71000, "High", "Low")
#head(Samples)

#Calculate CH4 concentrations in ppm
Samples$CH4_Conc_ppm <- ifelse(Samples$CH4_Area >71000, (Samples$CH4_Area-Slope_CH4_high$Intercept)/Slo

#Calculate CO2 concentrations in ppm
#Samples$CO2_Conc_ppm <- ((Samples$CO2_Area-Slope_CO2$Intercept)/Slope_CO2$Slope)
```

```r
#head(Samples)

##########make flags for any dilutions needed
#highest CH4 standard = 10000
#highest CO2 standard = 50000

Samples$CH4_Flag <- ifelse(Samples$CH4_Conc_ppm >10000, "Needs Dilution", "Within Range")
#Samples$CO2_Flag <- ifelse(Samples$CO2_Conc_ppm >50000, "Needs Dilution", "Within Range")
#head(Samples)
```

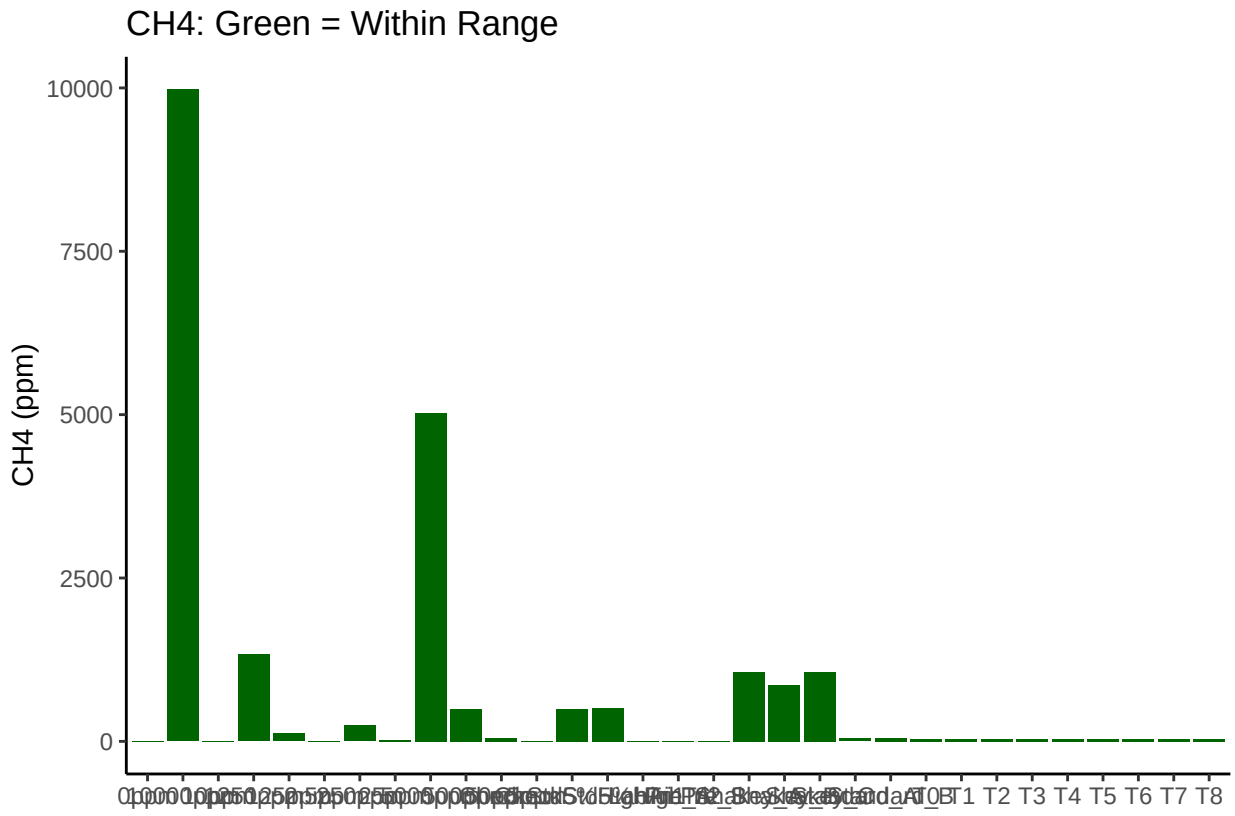
0.3 Check the Check Standards



0.4 Dilution correct samples

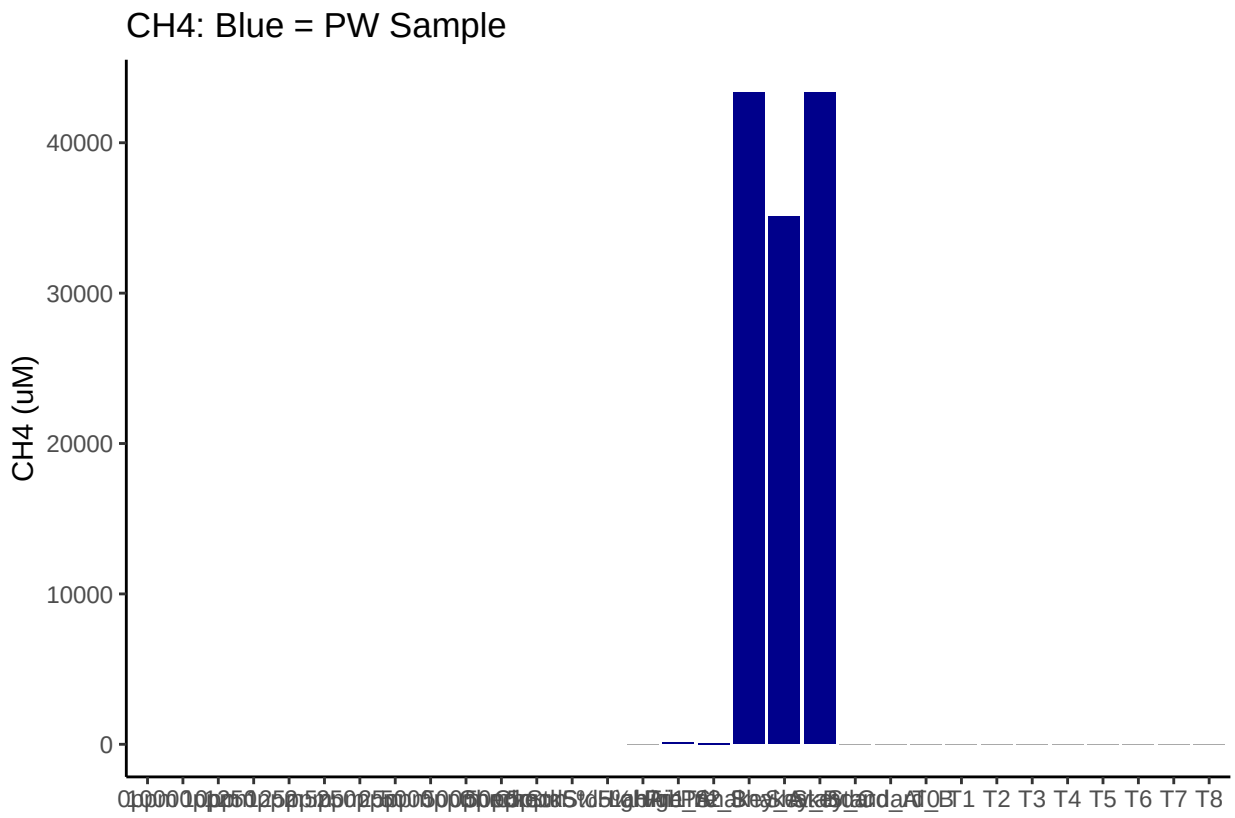
```r
#multiply the concentration by the dilution factor
Samples$CH4_Conc_ppm_dilcorr <- (Samples$CH4_Conc_ppm * Samples$Dilution_Factor)

#Samples$CO2_Conc_ppm_dilcorr <- (Samples$CO2_Conc_ppm * Samples$Dilution_Factor)

#check results
#head(Samples)
```

```
#quick first look at the samples
ch4_samples <-  ggplot(data = Samples, aes(x = Sample_ID, y = CH4_Conc_ppm, fill=CH4_Flag)) +
       geom_bar(stat = 'identity') +
        scale_fill_manual(values=c("darkgreen","red"))+
        #scale_fill_gradient2(low='red', mid='white', high='blue', space='Lab') +
        theme_classic() + labs(x= " ", y="CH4 (ppm)", title="CH4: Green = Within Range") +
        theme(legend.position="none")


#co2_samples <-  ggplot(data = Samples, aes(x = Sample_ID, y = CO2_Conc_ppm, fill=CO2_Flag)) +
#       geom_bar(stat = 'identity') +
 #       scale_fill_manual(values=c("red", "darkgreen"))+
        #scale_fill_gradient2(low='red', mid='white', high='blue', space='Lab') +
 #       theme_classic() + labs(x= " ", y="CO2 (ppm)", title="CO2: Green = Within Range") +
 #       theme(legend.position="none")

#ggarrange(ch4_samples, co2_samples, nrow=1, ncol=2)


print(ch4_samples)
```



0.5 If samples are water calculate gas in water

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



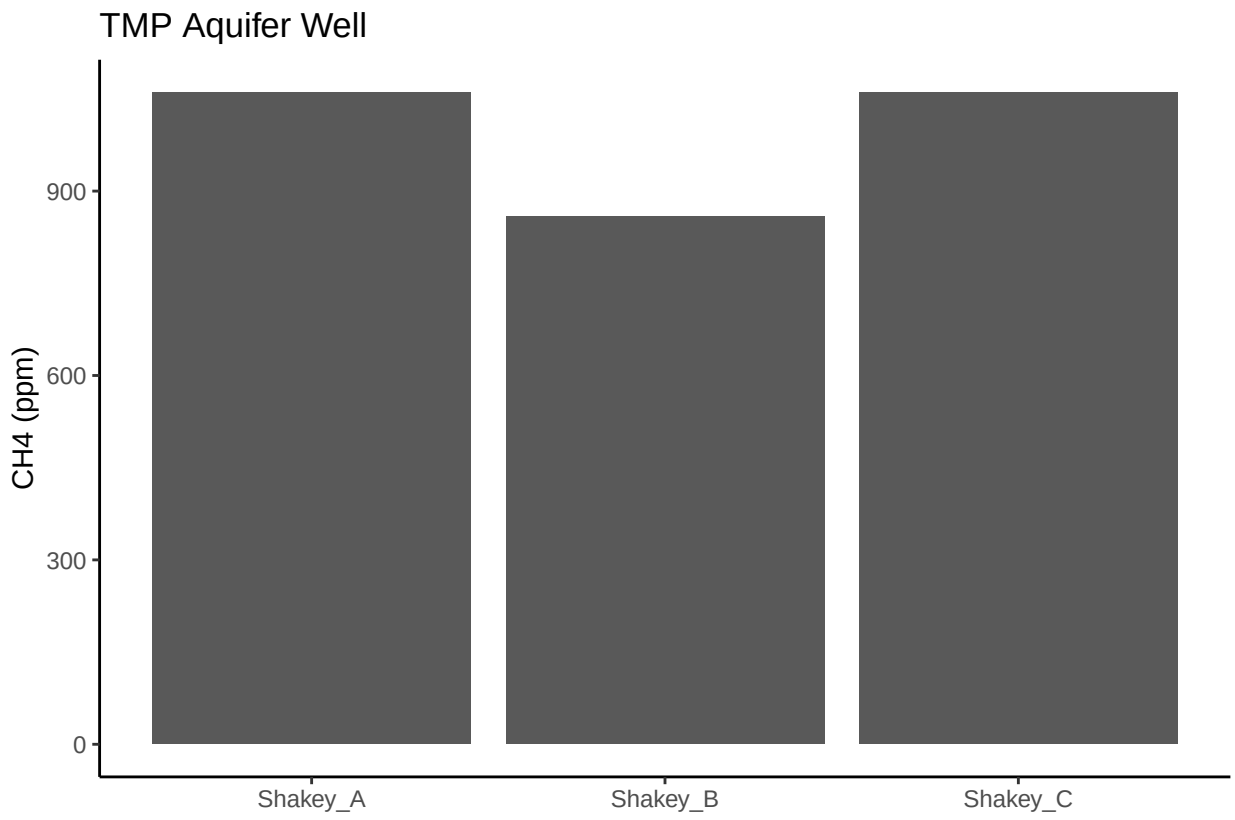
0.6 Write out processed data & slopes

```
#check results
#head(Samples)

#pull out what we need
#Samples1 <- Samples[ ,c(1:3,6:17,21:24)]
#head(Samples1)

Samples1 <-  Samples %>%
  filter(str_detect(Sample_ID, "Shakey"))

final_data <- Samples1 %>%
  #select(Project, Plot, grid, sample_name, Vial_ID, date, ) %>%
  mutate(
    Project = "COMPASS",   # new column with same value on every row
    Experiment = "TEMPEST: Well Test",
    Sample_Date = "2025-09-04",
    Sample_Time = c("14:00", "14:00", "14:00"),
    Replicate = c("A", "B", "C")#,
    #Run_notes = run_notes     # new column with notes about the run
  )

#this needs altered to match the tempest metadata and clean up
```

```r
final_data <- final_data %>%
  rename(
    sample_name = Sample_ID,
    CH4_ppm = CH4_Conc_ppm_dilcorr ,
    CH4_uM = CH4_Conc_umol,
    # add more rename pairs as needed
  ) %>%
  select(
    Project, Experiment, Sample_Date, Sample_Time, Replicate, sample_name,
    CH4_ppm, CH4_uM, CH4_Flag #, tdn_mgL, tdn_uM, tdn_flag, Analysis_runtime,
    #Run_notes
    # list columns in the order you want them
  )

head(final_data)
```

```
##   Project         Experiment Sample_Date Sample_Time Replicate sample_name
## 1 COMPASS TEMPEST: Well Test  2025-09-04       14:00         A    Shakey_A
## 2 COMPASS TEMPEST: Well Test  2025-09-04       14:00         B    Shakey_B
## 3 COMPASS TEMPEST: Well Test  2025-09-04       14:00         C    Shakey_C
##     CH4_ppm   CH4_uM     CH4_Flag
## 1 1060.6322 43.35094 Within Range
## 2  858.7977 35.10141 Within Range
## 3 1060.6586 43.35202 Within Range
```

```r
ch4_shakeys <-  ggplot(data = final_data, aes(x = sample_name, y = CH4_ppm)) +
       geom_bar(stat = 'identity') +
        #scale_fill_manual(values=c("darkgrey","darkblue"))+
        #scale_fill_gradient2(low='red', mid='white', high='blue', space='Lab') +
        theme_classic() + labs(x= " ", y="CH4 (ppm)", title="TMP Aquifer Well") +
        theme(legend.position="none")
ch4_shakeys
```




```
write.csv(final_data, "Processed Data/TMP_20250904_FW_WellTest_GHG_Processed.csv")
```

#end