Semi-quantification: Homologous series vs ML model predictions

1) Finding homologue series compounds
2) Semi-quantification with homologue
3) Predicting response factors for homologue series compounds + quantification
4) Comparison in table + plots

Finding homologue series compounds

From Thomas' data, a dataset was generated, which contained all compounds that had at least one homologue within the dataset.

Assumption: Two compounds are considered homologues when their difference in molecular formula is CF_2.

This summary is made on the example of CF_2 homologues only (data for CF_2CF_2 homologues available if needed).

```
## # A tibble: 18 x 3
##    Compound Homologue pattern_CF2
##    <chr>    <chr>     <chr>
##  1 PFDA     PFNA      smaller
##  2 PFDA     PFUnDA    bigger
##  3 PFDoDA   PFTriDA   bigger
##  4 PFDoDA   PFUnDA    smaller
##  5 PFHpA    PFHxA     smaller
##  6 PFHpA    PFOA      bigger
##  7 PFHxA    PFHpA     bigger
##  8 PFHxA    PFPeA     smaller
##  9 PFNA     PFDA      bigger
## 10 PFNA     PFOA      smaller
## 11 PFOA     PFHpA     smaller
## 12 PFOA     PFNA      bigger
## 13 PFPeA    PFHxA     bigger
## 14 PFTeDA   PFTriDA   smaller
## 15 PFTriDA  PFDoDA    smaller
## 16 PFTriDA  PFTeDA    bigger
## 17 PFUnDA   PFDA      smaller
## 18 PFUnDA   PFDoDA    bigger
```

Semi-quantification with homologue

For each compound, calibration curve of a homologue was used for semi-quantification. If two homologues existed (bigger and smaller), quantification was done with both.

First approach: Only slope (RF) was used to calculate concentrations (regression line was not forced to go though zero).

$(conc = area/slope_{homologue})$

Second approach: Both slope and intercept were used to calculate concentrations.

$(conc = (area - intercept_{homologue})/slope_{homologue})$

Predicting response factors for homologue series compounds + quantification

For each homologue series compound, the compound was removed from the training data and prediction model was trained (10 prediction models were trained in total). Then, the model was used to predict IE of the compound. IE was predicted to all training data to predict RF from IE and concentration of compound was calculated.

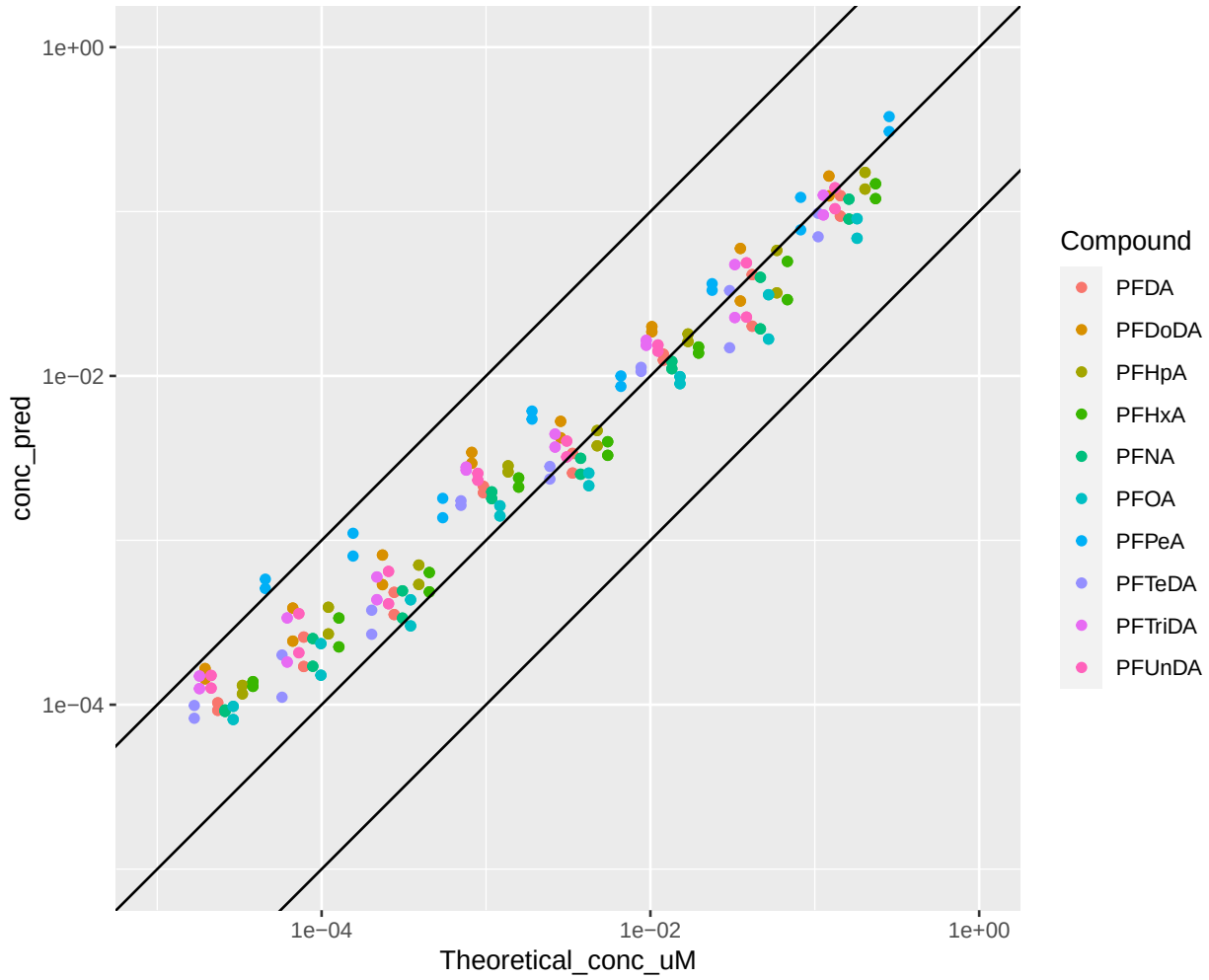
$(conc = area/slope_{predicted})$

Comparison in table + plots

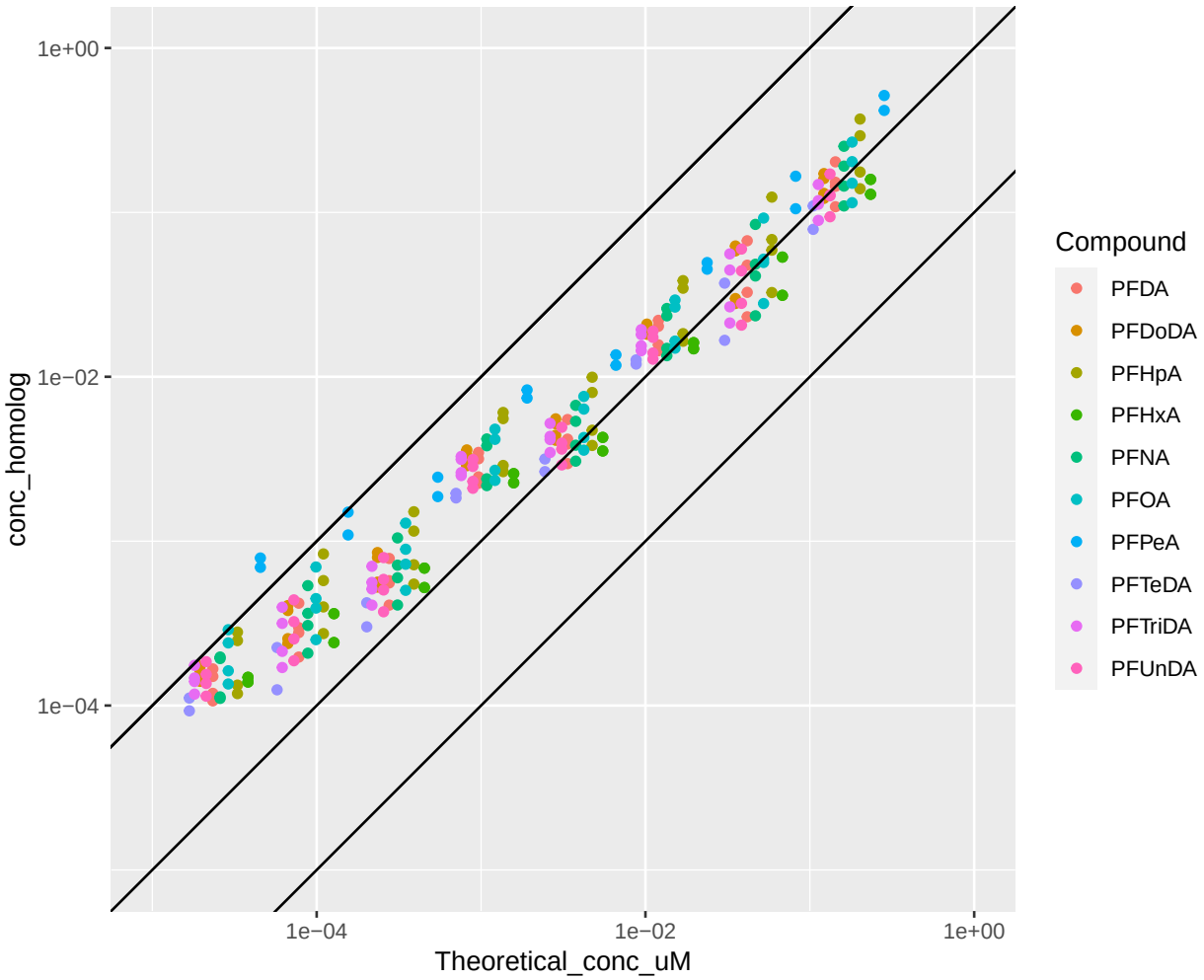
Comparing semi-quantification results from predicted slopes and homologue series compounds slopes with theoretical concentration. Ideal regression and ten-times error lines were added.

```
# Plot of concentrations calculated with predicted IEs vs experimental
IE_c_plot = ggplot(data = summary_table_CF2_filtered)+
  geom_point(mapping = aes(x = Theoretical_conc_uM,
                           y = conc_pred,
                           color = Compound)) +
  scale_y_log10(limits = c(10^-5, 10^0)) +
  scale_x_log10(limits = c(10^-5, 10^0)) +
  geom_abline(slope = 1, intercept = 0) +
  geom_abline(slope = 1, intercept = 1) +
  geom_abline(slope = 1, intercept = -1) +
  theme(aspect.ratio = 1#,
        #legend.position = "none"
        )

# Plot of concentrations calculated with homologue series compound vs experimental
homolog_c_plot = ggplot(data = summary_table_CF2_filtered)+
  geom_point(mapping = aes(x = Theoretical_conc_uM,
                           y = conc_homolog,
                           color = Compound,
                           text = Compound_homolog)) +
  scale_y_log10(limits = c(10^-5, 10^0)) +
  scale_x_log10(limits = c(10^-5, 10^0)) +
  geom_abline(slope = 1, intercept = 0) +
  geom_abline(slope = 1, intercept = 1) +
  geom_abline(slope = 1, intercept = -1) +
  geom_abline(slope = 1, intercept = 1) +
   theme(aspect.ratio = 1#,
         #legend.position = "none"
         )


# Plot of concentrations calculated with homologue series compound vs experimental
homolog_c_plot_intercept = ggplot(data = summary_table_CF2_filtered)+
  geom_point(mapping = aes(x = Theoretical_conc_uM,
                           y = conc_homolog_withIntercept,
                           color = Compound,
                           text = Compound_homolog)) +
  scale_y_log10(limits = c(10^-5, 10^0)) +
  scale_x_log10(limits = c(10^-5, 10^0)) +
  geom_abline(slope = 1, intercept = 0) +
```

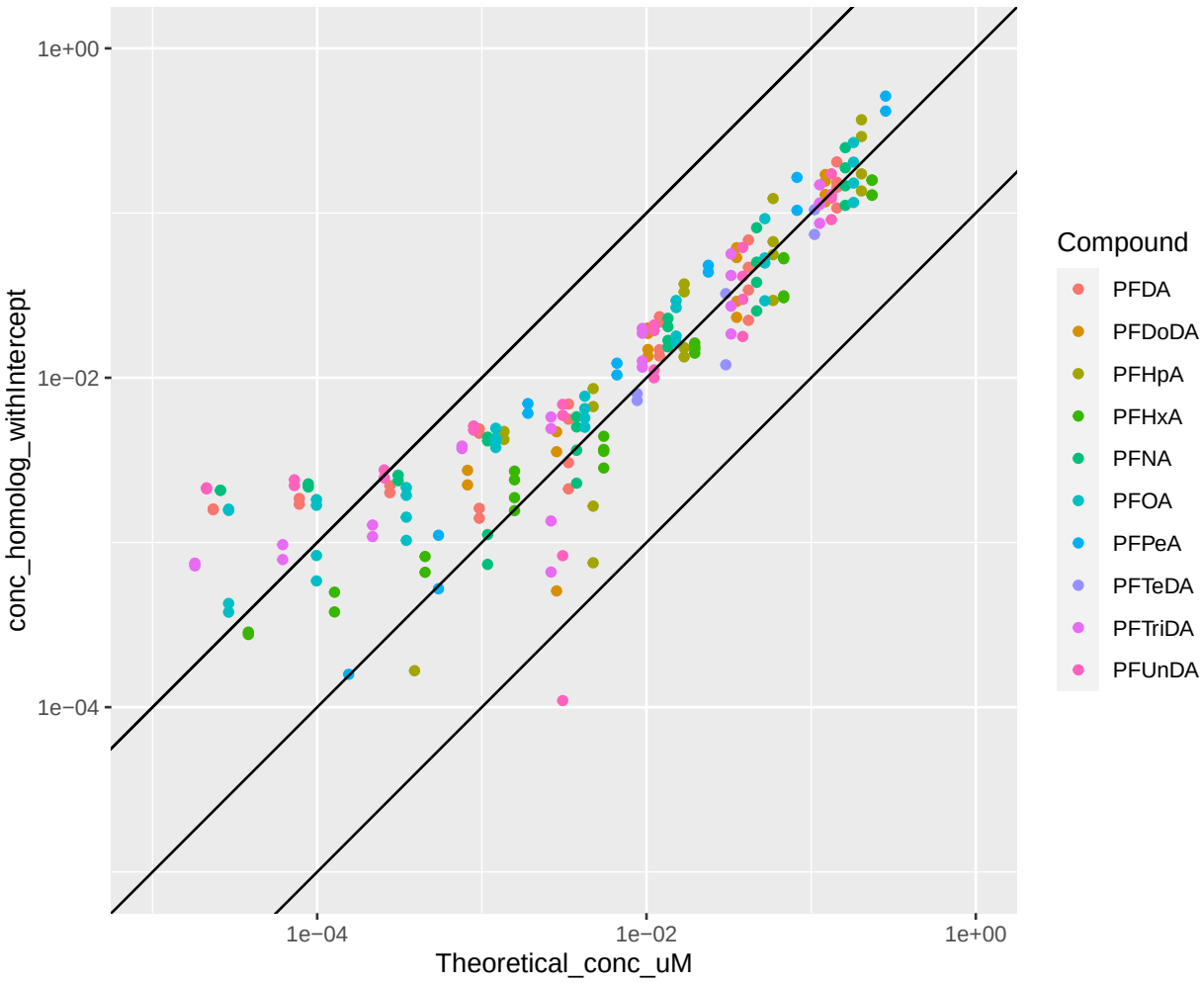
```
  geom_abline(slope = 1, intercept = 1) +
  geom_abline(slope = 1, intercept = -1) +
  geom_abline(slope = 1, intercept = 1) +
   theme(aspect.ratio = 1#,
         #legend.position = "none"
         )

plot(IE_c_plot)
```



```
plot(homolog_c_plot)
```




```
plot(homolog_c_plot_intercept)
```




```
# Comparison
# plot_comp <- plot_grid(IE_c_plot, homolog_c_plot, homolog_c_plot_intercept, ncol = 3)
# plot_comp


# Error calculations
summary_table_CF2_filtered = summary_table_CF2_filtered %>%
  mutate(error_IE = case_when(
    Theoretical_conc_uM > conc_pred ~ Theoretical_conc_uM/conc_pred,
    TRUE ~ conc_pred/Theoretical_conc_uM),
    error_homolog = case_when(
      Theoretical_conc_uM > conc_homolog ~ Theoretical_conc_uM/conc_homolog,
      TRUE ~ conc_homolog/Theoretical_conc_uM),)
```

```r
summary_table_CF2_filtered %>%
  na.omit() %>%
  group_by(pattern) %>%
  summarize(error_IE = mean(error_IE),
            error_homolog = mean(error_homolog)) %>%
  ungroup()
```

```
## # A tibble: 2 x 3
##   pattern error_IE error_homolog
##   <chr>      <dbl>         <dbl>
## 1 bigger      2.44          2.67
## 2 smaller     2.27          2.96
```