Comparison of chromatographic stationary phases using Bayesian-based multilevel modeling

Paweł Wiczling, Agnieszka Kamedulska

2023-08-24

Table of contents

1 Introduction **3**

2 Experimental design **3**

3 Setup **3**

4 Data **3**
4.1 Prepare the data . . . 4
4.2 Exploratory data analysis . . . 4
4.2.1 Plots . . . 4
4.2.2 Summary of the data . . . 10

5 Methods **11**
5.1 Model . . . 11
5.2 Stan . . . 17
5.2.1 Initialize variables and parameters . . . 17
5.2.2 The Stan model: . . . 17
5.2.3 Fitting the model . . . 31

6 Results **32**
6.1 Summary of model parameters (table): . . . 32
6.2 Trace plots . . . 32
6.3 Summary of model parameters (figures) . . . 35
6.4 Isocratic predictions . . . 43

6.5 Individual Parameters . . . 61
6.5.1 Neutral Form . . . 61
6.5.2 Effects of dissociation . . . 65
6.5.3 pKa-related paraemters . . . 70
6.6 Eta plots . . . 70
6.6.1 Neutral Forms . . . 71
6.6.2 Effect of dissociation . . . 74
6.6.3 Effect of functional groups (exploratory) . . . 76

7 Comparison of observed and model predicted retention times 94
7.1 Individual predictions: . . . 95
7.2 Population predictions: . . . 101

8 Uncertainity chromatogram 106
8.1 Isocratic predictions (population predictions) . . . 108

9 Effect of logP 126

10 Goodness of fit plots 127

11 Predictions and decision making. Case Study 1 130
11.1 Utility maps . . . 131
11.1.1 Uncertainity chromatogram . . . 137

12 Predictions and decision making. Case 2 138
12.1 Summary of individual parameters . . . 139
12.2 Predicted retention times . . . 140
12.2.1 Plot the predicted vs. observed: . . . 140
12.3 Decision making . . . 145
12.3.1 Uncertainity chromatogram . . . 149
12.4 Utility maps . . . 150
12.4.1 Uncertainity chromatogram . . . 156

13 Conclusions 157

References 158

Licenses 158

Original Computing Environment 158

1 Introduction

In this work we applied the previously developed Bayesian multilevel framework Kubik, Kaliszan, and Wiczling (2018) to characterize chromatographic gradient retention time datasets collected using a multicomponent mixtures of analytes, five stationary phases, and a wide range of chromatographic conditions (pH, organic modifier, temperature, gradient program). Such datasets carry much information about chromatographic retention that, if extracted, can provide useful predictive information, i.e. a detailed multidimensional characterization of chromatographic stationary phases and ability to predict retention (along with uncertainty) based on various number of preliminary experiments (e.g. to predict retention time for a set of analytes given no, or several measurements collected using a different stationary phase).

In this case study, we compared five RP-HPLC stationary phases (XBridge Shield RP18, XTerra MS C18, XBridge Phenyl, XBridge C8, Xterra MS C8) based on LC-MS/TOF data.

2 Experimental design

The data was collected using a mixture of 300 analytes and 84 gradient liquid chromatography experiments for each column. The experiments differed in gradient duration (30, 90, and 270 min) and pH of the mobile phase (from 2.5 to 10.5). Experiments were conducted in MeOH or ACN as organic modifiers and at two temperatures (25^0C and 35^0C).

The molecular structure of the analytes was converted from SMILE format to MDL mol format using OpenBabel. The input molecules were then analyzed for the presence of approximately 204 functional groups and structural elements using Checkmol (version 0.5b N. Haider, University of Vienna, 2003-2018). Functional groups that were not present on any analyte and functional groups merging other simpler functional groups were excluded from the analysis. The lipophilicity (log P), dissociation constant (pKalit) were added to the dataset. They were calculated using ACD/Labs program based on the structures of analytes generated from smiles strings.

3 Setup

The packages we will use are listed below.

4 Data

Data can be accessed via github (data folder) or osf.io repositories Kubik et al. (2022b).

4.1 Prepare the data

We will begin with loading and merging all the required datasets:

The data requires some cleaning:

1. we remove measurements with low score, analytes with less than 42 measurements per column, analytes with less than 210 measurements out of total 420, and use only measurements with the highest score (if several are present)
2. we remove some outlying measurements (mismatch between literature pKa and observed data)
3. we also prepare available predictors: pKa, logP and functional_groups
4. finally, we select analytes with <=2 dissociation steps fot the analysis

4.2 Exploratory data analysis

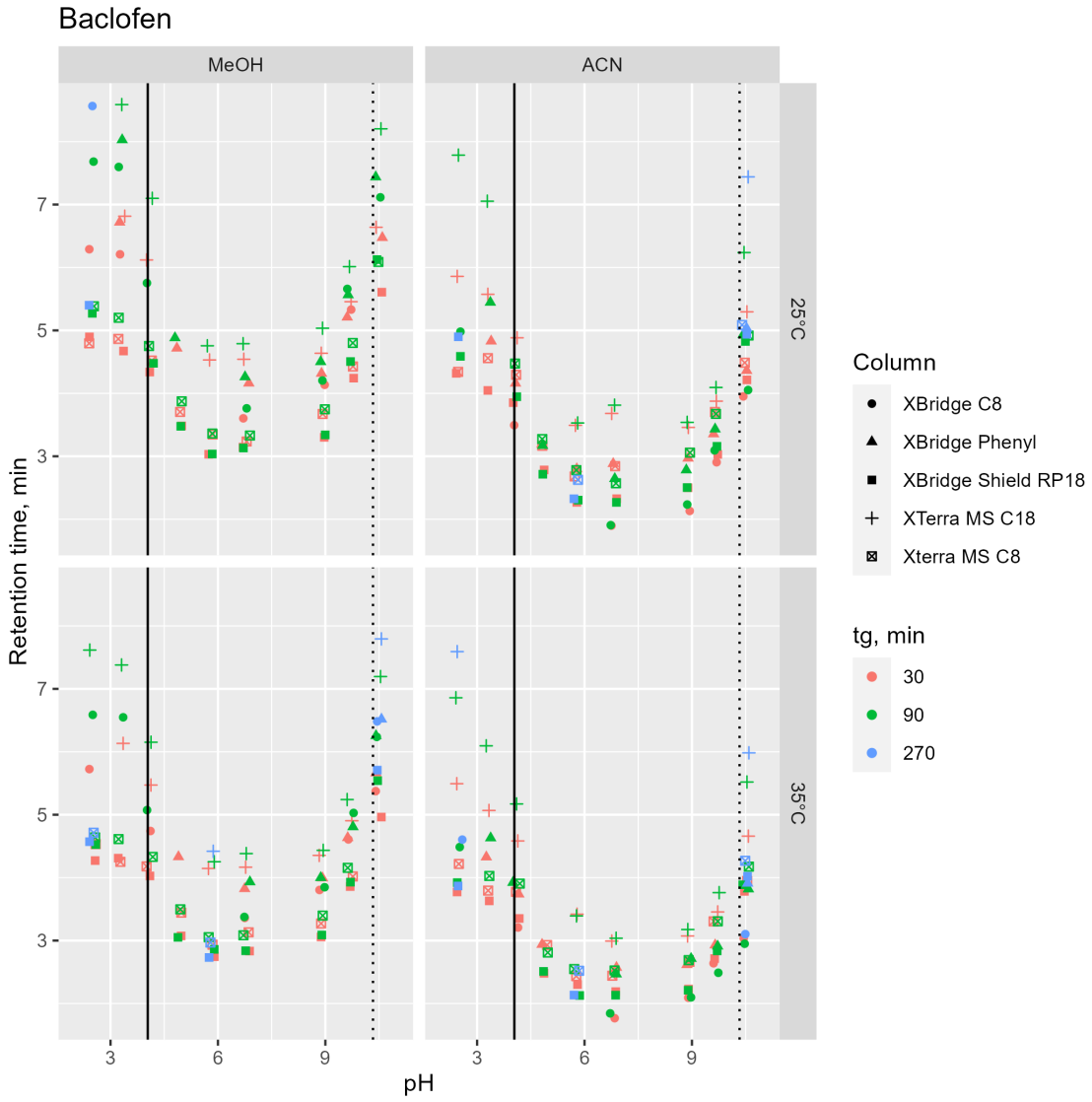
During the exploratory data analysis phase, we create a series of plots to better understand our data.

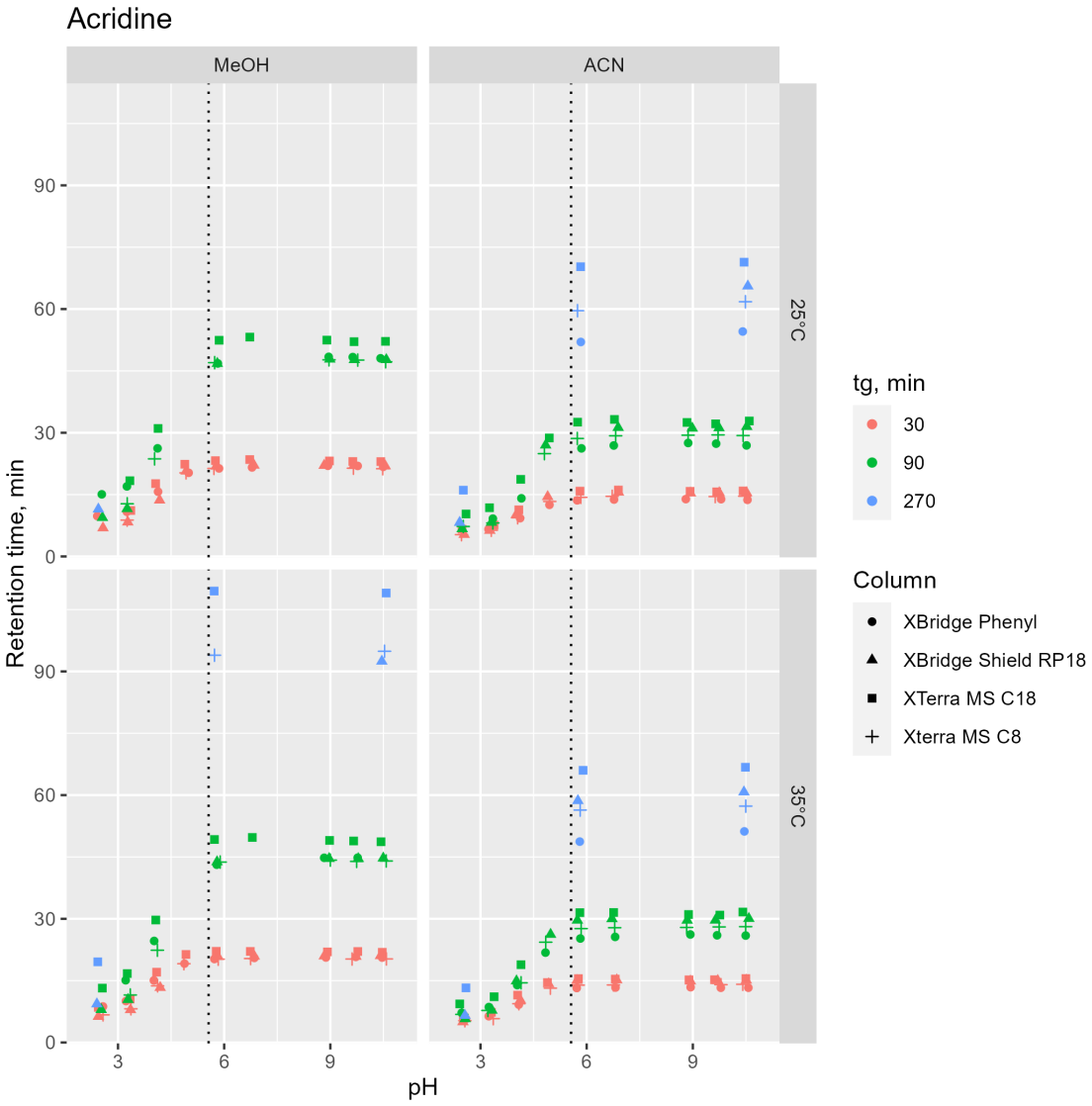
4.2.1 Plots

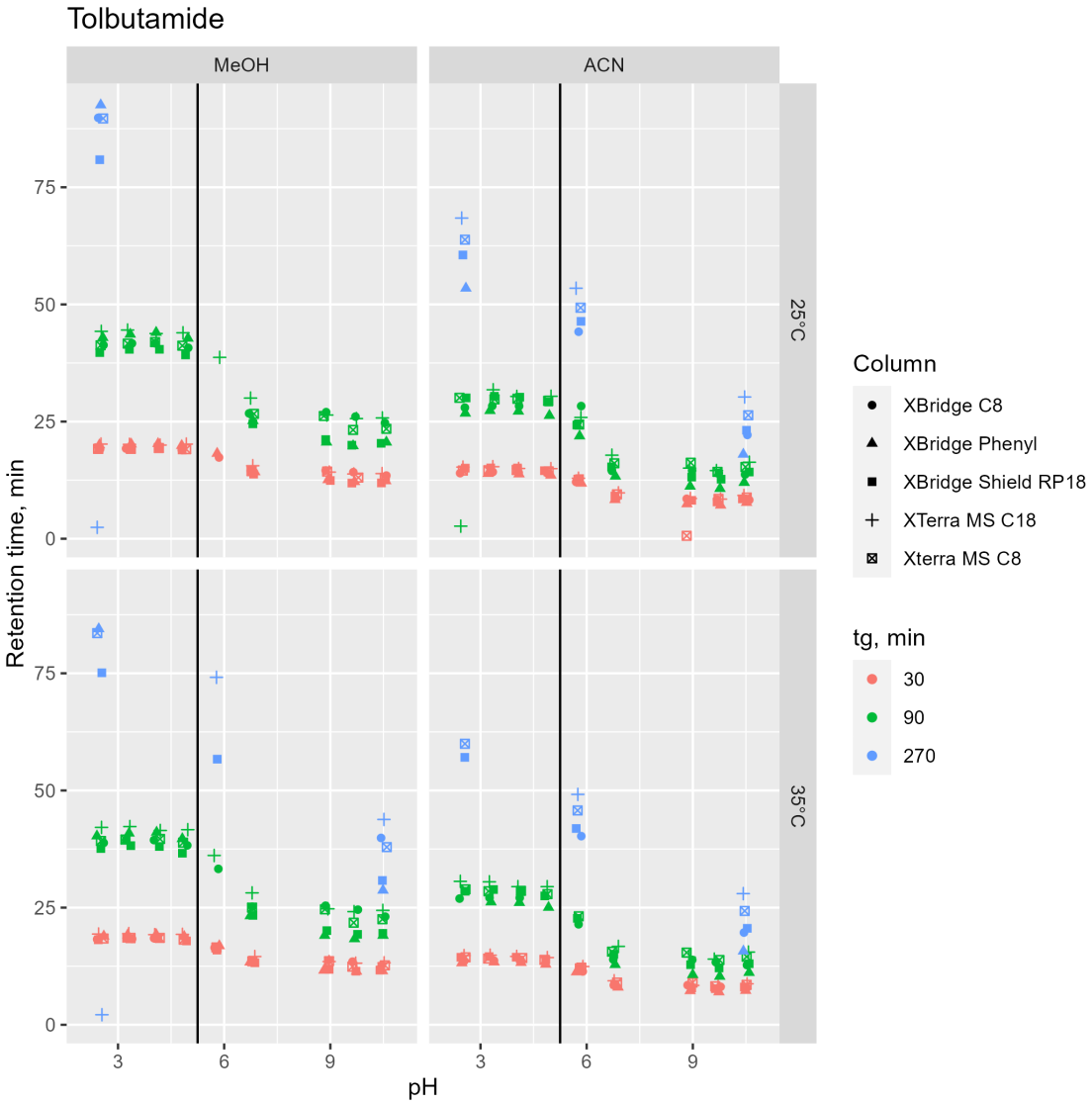
The following graphs present retention time profiles for 6 analytes:

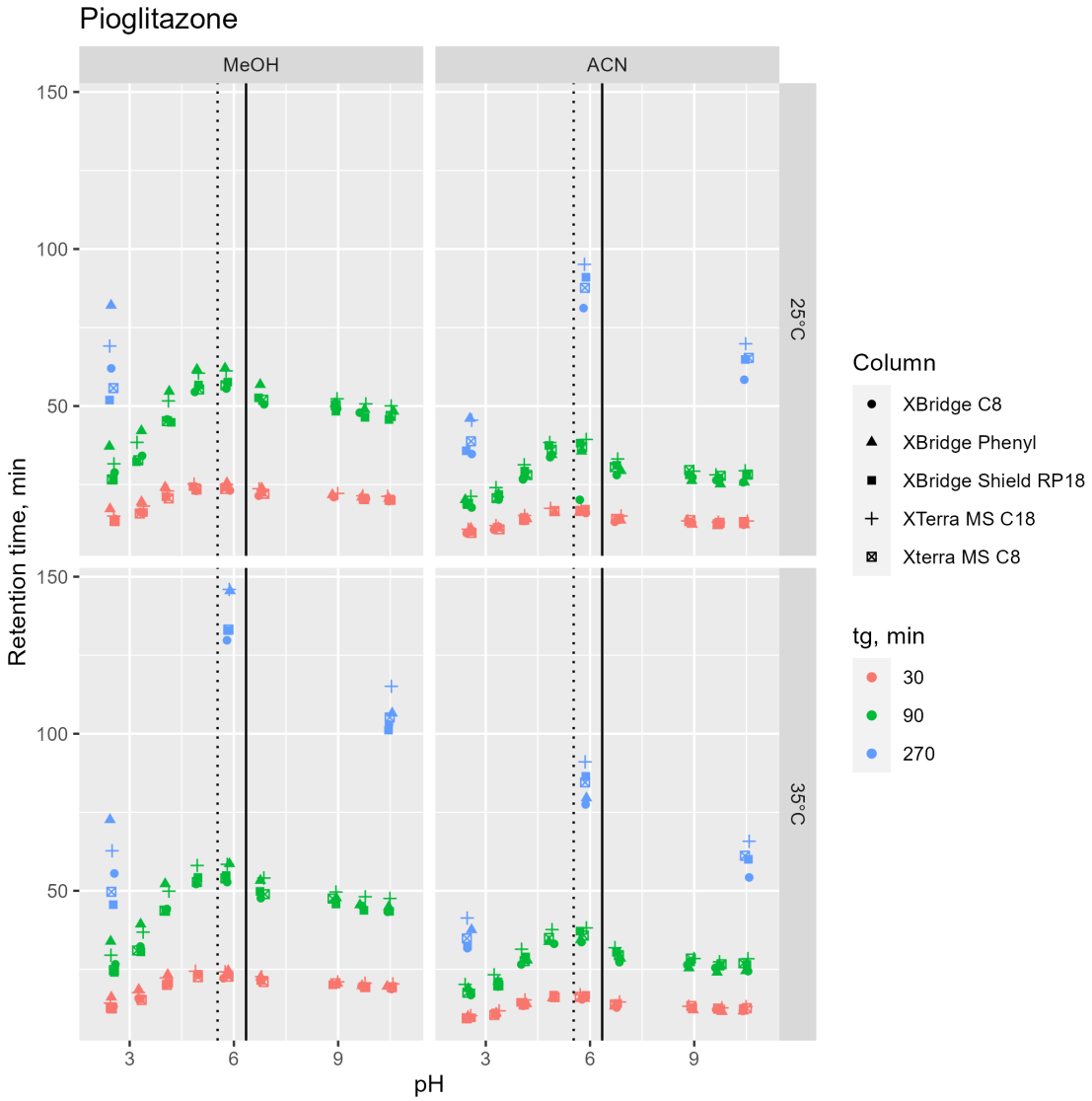
1. acridine (monoprotic acid)
2. baclofen (zwitterion: acidic and basic group)
3. hydrocortison (neutral)
4. pioglitazone (zwitterion: basic and acidic group)
5. quinine (diprotic: 2 basic groups),
6. tolbutamide (monoprotic base).

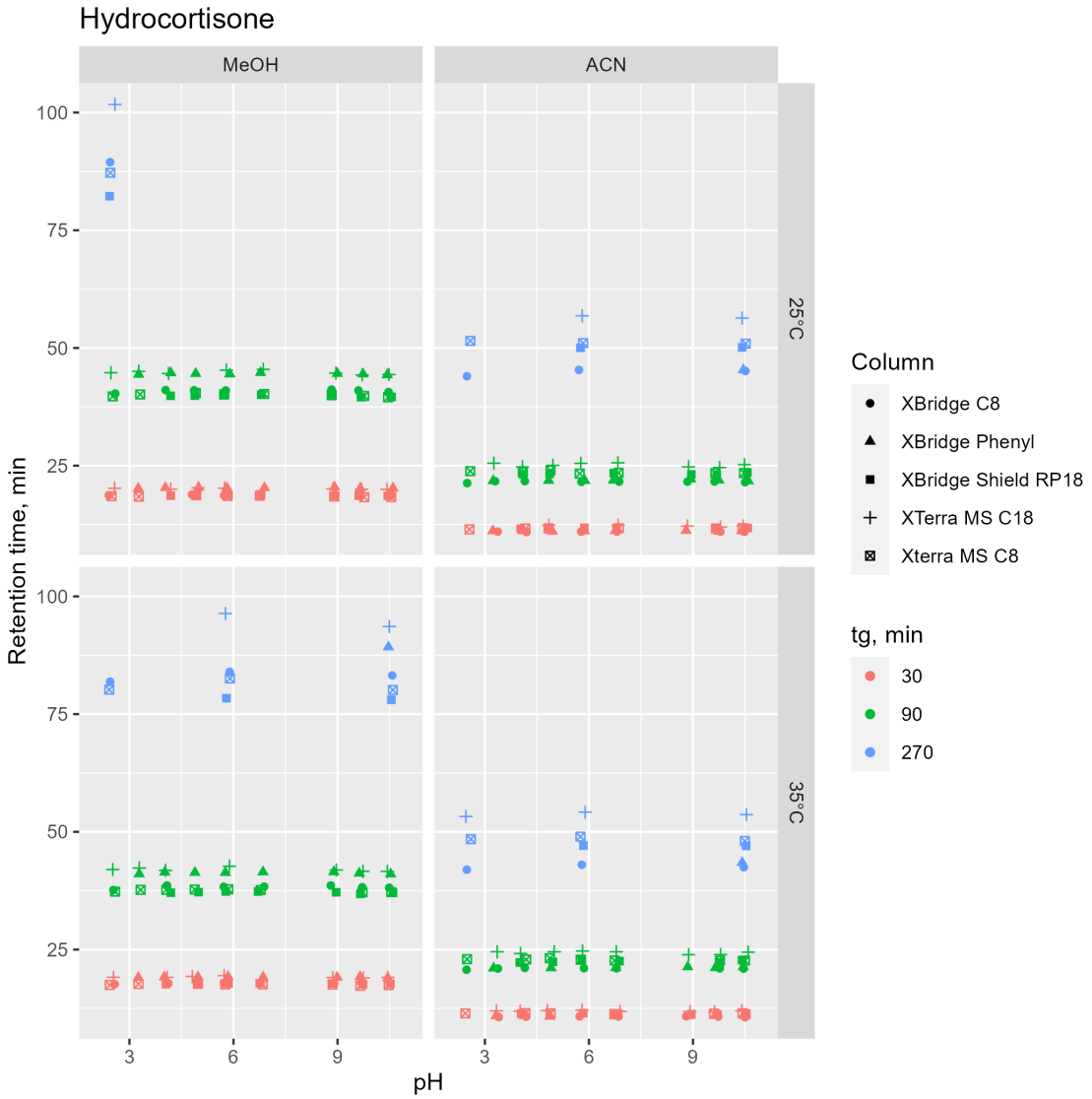
The vertical lines show ACD-based pKa values (solid line: acidic group; dotted line: basic group)

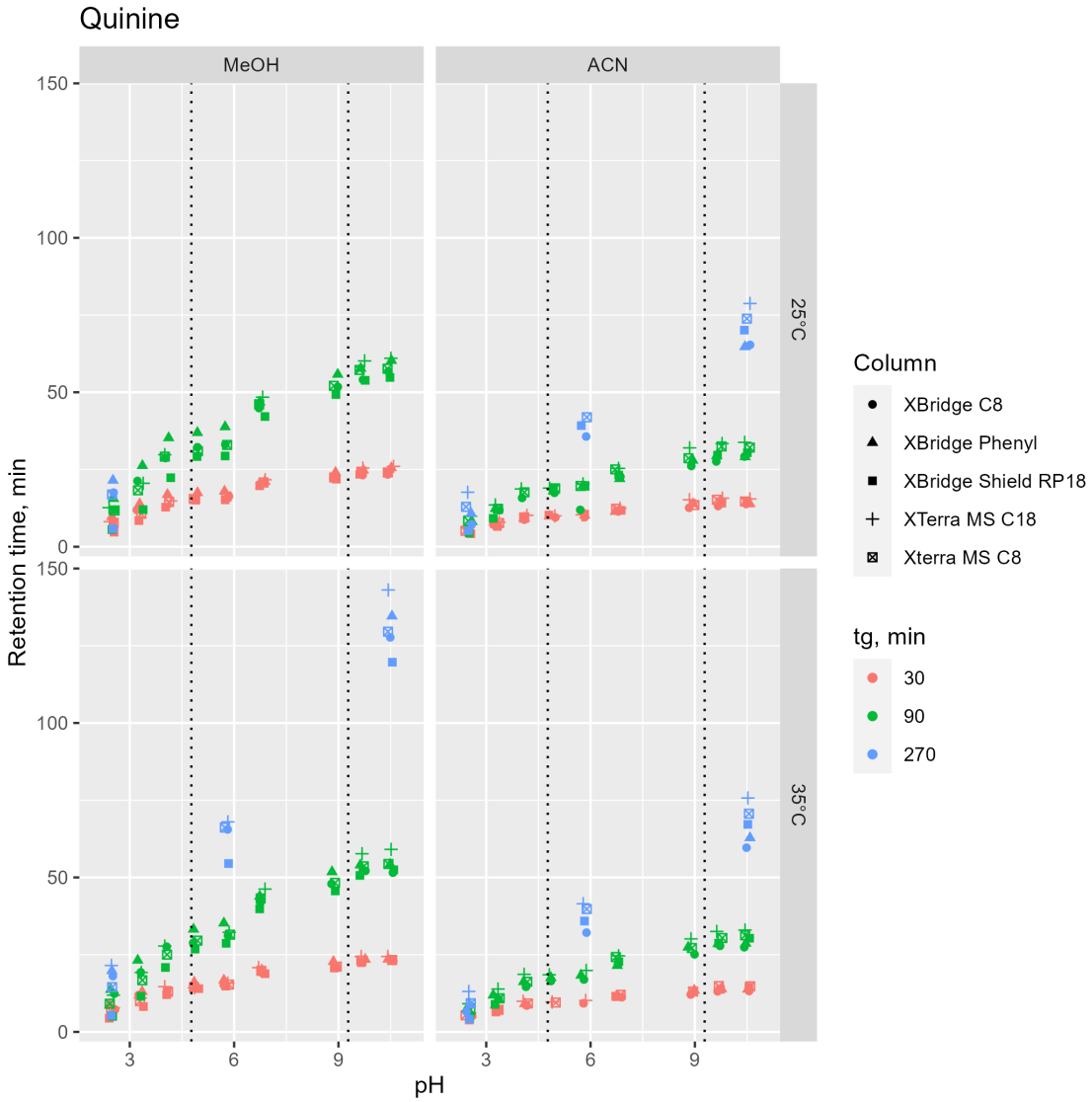

Baclofen
MeOH
ACN
25°C
35°C
Retention time, min
pH
Column
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
tg, min
30
90
270


Acridine
MeOH
ACN
25°C
35°C
Retention time, min
pH
0
30
60
90
3
6
9
tg, min
30
90
270
Column
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8


Tolbutamide
MeOH
ACN
25°C
35°C
Retention time, min
0
25
50
75
3
6
9
pH
Column
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
tg, min
30
90
270

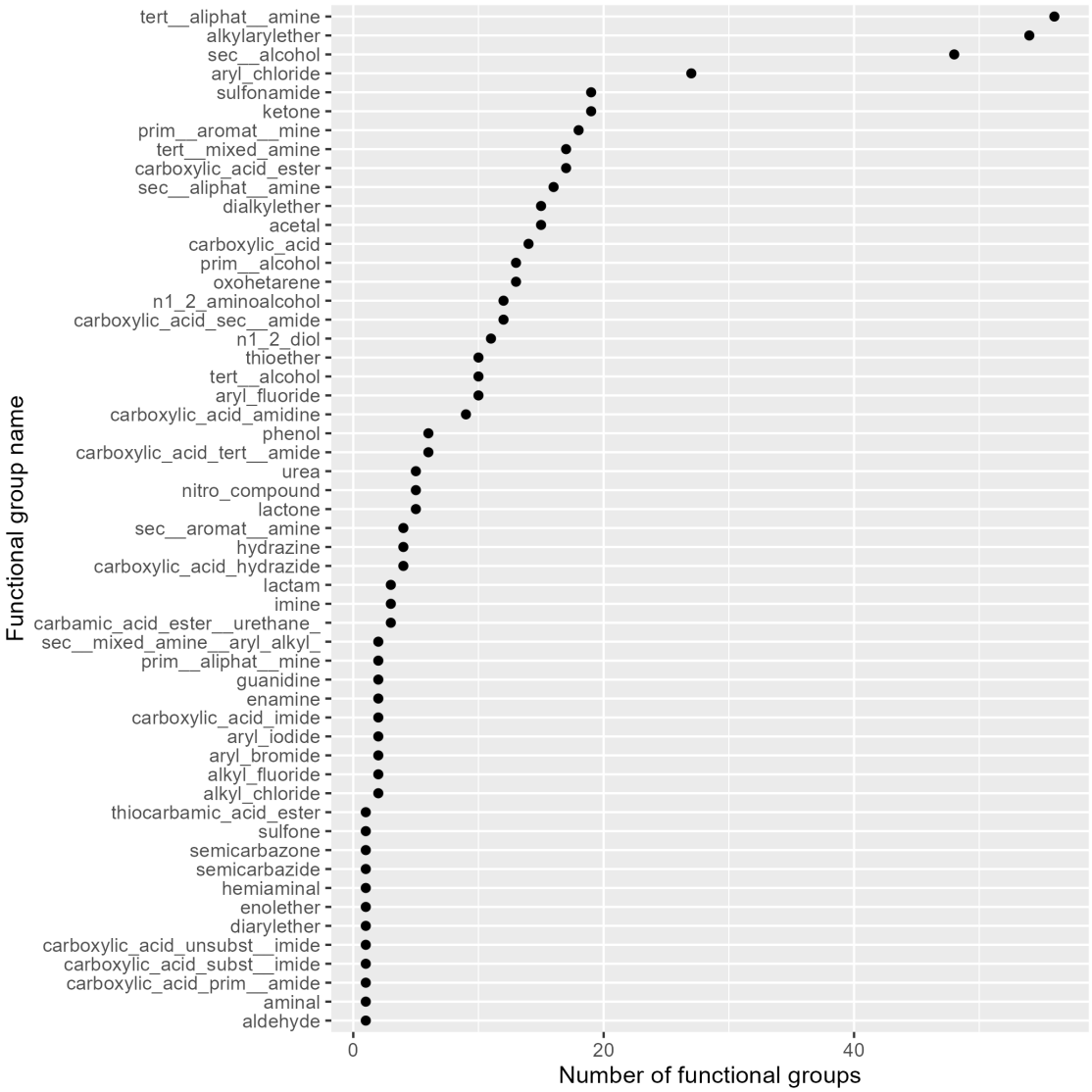

Pioglitazone
MeOH
ACN
25°C
35°C
Retention time, min
150
100
50
3
6
9
pH
Column
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
tg, min
30
90
270


Hydrocortisone
MeOH
ACN
25°C
35°C
Retention time, min
100
75
50
25
3
6
9
pH
Column
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
tg, min
30
90
270



4.2.2 Summary of the data

1. Number of identified analytes: 141
2. Number of observations: 51530
3. $logP$: 2.5 ± 1.82
4. Number of analytes with 0, 1 and 2 dissociation steps: 14, 88, 39
5. Number of acidic and basic groups: 46 and 120
6. Number of functional groups across all analytes included in the analysis:



5 Methods

5.1 Model

In this work z = 1..51530 denotes observation, i=1..141 denotes analyte, col=1..5 denotes column, m=1..2 denotes organic modifier and r=1..$R[i]$ denotes dissociation step for i-th analyte. The observed retention times ($t_{Robs,z}$) were described using the following model:

$$t_{Robs,z} \sim student_t(\nu, t_{R,z}, \sigma_{col[z],i[z]})$$

where z denotes z-th observation and student_t denotes the Student's t-distribution with the mean given by the predicted retention time $t_{R,z}$, scale $\sigma_{i,col}$ (analyte and column-specific), and normality parameter ν (set to 3).

Gradient retention time $t_{R,z}$ was calculated utilizing the well-known integral equation:

$$\int_0^{t_{R,z}-t_{0,z}-t_e} \frac{dt}{t_{0,z} \cdot ki_z(t)} = 1,$$

where $ki_z(t)$ denotes instantaneous isocratic retention factor corresponding to the mobile phase composition at time t at column inlet for analyte and conditions corresponding to the z-th observation, $t_{0,z}$ denotes column hold-up (dead) time and t_e denotes extra column-time. The numerical solution of this integral equation was carried out using method of steps with 4 and 10 steps for methanol and acetonitrile gradients using method proposed by Nikitas et al. (Nikitas and Pappa-Louisi 2002) The following function described the relationship between the isocratic retention factor and pH for an ith-analyte with $R[i]$ dissociation steps and $R[i]+1$ forms.

$$ki_z(t) = \frac{k_{z,i[z],1}(t) + \sum_{r=1}^{R[i[z]]} k_{z,i[z],r+1}(t) \cdot 10^{r \cdot pH_z(t) - \sum_{r=1}^{R[i]} pKa_{z,i[z],r}(t)}}{1 + \sum_{r=1}^{R} 10^{r \cdot pH_z(t) - \sum_{r=1}^{R[i[z]]} pKa_{z,i[z],r}(t)}}$$

$log(k_{z,i,r})$ was assumed to depend on the organic modifier content based on the Neue equation, on temperature assuming linear equation, and on the pH of the mobiles phase (for ionized forms of analytes).

$$log(k_{z,i,r}(t)) = logkw_{col[z],i,r} - \frac{S1_{m[z],col[z],i,r} \cdot (1 + S2_{m[z]}) \cdot \varphi_z(t)}{1 + S2_{m[z]} \cdot \varphi_z(t)} + ...$$
$$apH_{col[z],m[z],i,r} \cdot (pH_z(t) - 7) + dlogkT_{col[z],i} \cdot (T_z - 25)/10$$

where $logkw_{col,i,r}$ represents logarithm of retention factors extrapolated to 0% of organic modifier content for column col, i-th analyte and r-th analyte form; $S1_{i,m,col,r}$ and $S2_m$ are the slopes in the Neue equation for column c, modifier m, i-th analyte and r-th analyte form. In this parametrization of the Neue equation, $S1$ reflects the difference between logarithm of retention factors corresponding to water (0% of organic modifier content) and MeOH or ACN (100% of organic modifier content) as eluents. $dlogkT_{col,i}$ denotes the change in $logkw$ due to the increase in temperature by 10^oC. $apH_{col,m,i,r}$ denotes the pH effects on $logkw$;

Further a linear relationship between pKa values and the organic modifier content was assumed:

$$pKa_{z,i,r}(t) = pKaw_{i,r} + \alpha_{m[z],i,r} \cdot \varphi_z(t)$$

where $pKa_{z,i,r}(t)$ denotes dissociation constant of an i-th analyte and r-th dissociation step form and chromatographic conditions corresponding the z-th observation, $pKaw_{i,r}$ denotes aqueous pKa, and $\alpha_{m,i,r}$ denotes the slope for m-th modifier, i-th analyte and r-th form. The linear relationships is generally valid for $\varphi < 0.8$.

The relationship between pH and the organic modifier content for various combinations of organic modifier and buffer was experimentally determined prior to the chromatographic analysis. The obtained relationships was then described using quadratic equations for each nominal pH, temperature and organic modifier:

$$pH_z(t) = pHo_z + \alpha 1_z \cdot \varphi_z(t) + \alpha 2_z \cdot \varphi_z(t)^2$$

First, individual values of $logkw$, $S1$ are defined for the neutral form of an analyte in MeOH for the Xbridge Sheld RP18 column (denoted as $logkwN_i$ and $S1mN_i$). The effect of ACN was described as ($dS1N_i$), the effect of column ($c = 1..4$) was described by $clogkwN_{c,i}$, $cS1mN_{c,i}$, and $cdS1N_{c,i}$). The individual parameters for the neutral forms were described using the following equations:

$$\begin{bmatrix} logkwN_i \\ S1mN_i \end{bmatrix} \sim MVN(\begin{bmatrix} \widehat{logkw} + \beta_1 \cdot (logP_i - 2.2) \\ \widehat{S1m} + \beta_2 \cdot (logP_i - 2.2) \end{bmatrix}, \Omega)$$

$$dS1N_i \sim N(\widehat{dS1}, \omega_3)$$

$$\Omega = diag(\omega_{1:2}) \cdot \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \cdot diag(\omega_{1:2})$$

$$\begin{bmatrix} clogkwN_{1,i} \\ clogkwN_{2,i} \\ clogkwN_{3,i} \\ clogkwN_{4,i} \end{bmatrix} \sim MVN(\begin{bmatrix} \widehat{clogkw}_1 + c\beta_{1,1} \cdot (logP_i - 2.2) \\ \widehat{clogkw}_2 + c\beta_{2,1} \cdot (logP_i - 2.2) \\ \widehat{clogkw}_3 + c\beta_{3,1} \cdot (logP_i - 2.2) \\ \widehat{clogkw}_4 + c\beta_{4,1} \cdot (logP_i - 2.2) \end{bmatrix}, c\Omega)$$

$$c\Omega = diag(c\omega) \cdot \begin{bmatrix} 1 & c\rho_{12} & c\rho_{13} & c\rho_{14} \\ c\rho_{21} & 1 & c\rho_{23} & c\rho_{24} \\ c\rho_{31} & c\rho_{32} & 1 & c\rho_{34} \\ c\rho_{41} & c\rho_{42} & c\rho_{43} & 1 \end{bmatrix} \cdot diag(c\omega)$$

$$cS1mN_{c,i} \sim N(\widehat{cS1m} + c\beta_{c,2} \cdot (logP_i - 2.2), c\omega_{c,2}) \text{ for c=1...4}$$

$$cdS1N_{c,i} \sim N(\widehat{cdS1}_c, c\omega_{c,3}) \text{ for c=1...4}$$

$$dlogkT_{c,i} \sim N(\widehat{dlogkT}_c, \omega_{T,c}) \text{ for c=1...4}$$

were MVN denotes the multivariate normal distribution; $\widehat{logkw}$, $\widehat{S1m}$, $\widehat{dS1}$ are the mean values of individual chromatographic parameters that correspond to a typical neutral analyte with $logP$ =2.2 at 25^oC on Xbridge Shield RP18 stationary phase. βs are regression coefficients between the individual chromatographic parameters and the $logP_i$. ω denotes the standard deviation for between analyte variability (BAV). $\widehat{dlogkT}$ denotes the effect of temperature for a typical analyte and ω_T the standard deviation for between analyte variability for temperature effects. Similar set of equations was used for column effects. Here c denoted the column effect (4 differences with respect to he reference column).

The difference in retention between the ionized form of an analyte and the neutral form of an analyte was separately estimated for acids and bases for $logkw$, $S1m$, $dS1$ parameters. Similar set of equations was used for column effects.

$$
\begin{aligned}
&dlogkwA_a \sim N(\widehat{dlogkw}_1, \kappa_1),\\
&dlogkwB_b \sim N(\widehat{dlogkw}_2, \kappa_1),\\
&dS1mA_a \sim N(\widehat{dS1m}_1, \kappa_2),\\
&dS1mB_b \sim N(\widehat{dS1m}_2, \kappa_2),\\
&ddS1A_a \sim N(\widehat{ddS1}_1, \kappa_3),\\
&ddS1B_b \sim N(\widehat{ddS1}_2, \kappa_3),\\
&cdlogkwA_{c,a} \sim N(\widehat{cdlogkw}_{c,1}, c\kappa_{c,1}) \text{ for c=1...4},\\
&cdlogkwB_{c,b} \sim N(\widehat{cdlogkw}_{c,2}, c\kappa_{c,1}) \text{ for c=1...4},\\
&cdS1mA_{c,a} \sim N(\widehat{cdS1m}_{c,1}, c\kappa_{c,2}) \text{ for c=1...4},\\
&cdS1mB_{c,b} \sim N(\widehat{cdS1m}_{c,2}, c\kappa_{c,2}) \text{ for c=1...4},\\
&cddS1A_{c,a} \sim N(\widehat{cddS1}_{c,1}, c\kappa_{c,3}) \text{ for c=1...4},\\
&cddS1B_{c,b} \sim N(\widehat{cddS1}_{c,2}, c\kappa_{c,3}) \text{ for c=1...4}.
\end{aligned}
$$

where a=1..46 and b=1..120 denote the indexes of acidic and basic groups.

Similarly pKa an α parameters were described separately for acids and based:

$$
\begin{aligned}
&pKawA_a \sim N(pKaAlit_a, \tau_1),\\
&pKawB_b \sim N(pKaBlit_b, \tau_1),\\
&\alpha mA_a \sim N(\widehat{\alpha m}_1, \tau_2),\\
&\alpha mB_b \sim N(\widehat{\alpha m}_2, \tau_2),\\
&d\alpha A_a \sim N(\widehat{d\alpha}_1, \tau_3),\\
&d\alpha B_b \sim N(\widehat{d\alpha}_2, \tau_3).
\end{aligned}
$$

Further, we created the matrices containing the value of parameters for i-th analyte, col-th column, m-th modifier an r-th dissociation step based on the value of neutral form and effects of column, organic modifier, and dissociation. This transformation was denoted as f(.). The exact procedure can be found in the model code displayed later.

$$
\begin{aligned}
&logkw_{col,i,r} = f(logkwN_i, clogkwN_{c,i}, dlogkwA_a, cdlogkwA_{c,a}, dlogkwB_b, cdlogkwB_{c.b}, ...)\\
&S1_{m,col,i,r} = f(S1mN_i, cS1mN_{c,i}, dS1mA_a, cdS1mA_{c,a}, S1mB_b, cdS1mB_{c,b}, ...\\
&dS1N_i, cdS1N_{c,i}, ddS1A_a, cddS1A_{c,a}, ddS1B_b, cddS1B_{c,b}, ...)\\
&apH_{m,col,i,r} = f(\hat{apH}_1, \hat{capH}_{c,1}, \hat{apH}_2, \hat{capH}_{c,2}, ...)\\
&S2_m = 10^{f(\hat{logS2m}, \hat{dlogS2}, ...)}\\
&pKaw_{i,r} = f(pKawA_a, pKawB_b, ...)\\
&\alpha_{m,i,r} = f(\alpha mA_a, d\alpha A_a, \alpha mB_b, d\alpha B_b, ...)
\end{aligned}
$$

Residual error model assumes different parameters for each column and analyte:

$$
\begin{aligned}
&log(\sigma_{col,i}) = f(log\sigma_i, clog\sigma_{c,i})\\
&log\sigma_i \sim N(log(m\sigma), s\sigma)\\
&clog\sigma_{c,i} \sim N(clogm\sigma_c, cs\sigma_c) \text{ for c=1...4,}
\end{aligned}
$$

The detailed description of parameters and used priors is provided in the following table (BAV denotes between analyte variability):

Table 1: Desciption of model paramters

Name	Namecode	Description	Priors
XBridge Shield RP18 parameters			
$\hat{logkw}$	logkwHat	typical logkw [Neutral]	N(2.2,2)
$\hat{S1m}$	S1mHat	effect of MeOH on logkw [Neutral]	N(4, 1)
$\hat{dS1}$	dS1Hat	effect of ACN on S1m [Neutral]	N(1, 1)
β_1	beta[1]	effect of logP on logkw [Neutral]	N(1, 0.125)
β_2	beta[2]	effect of logP on S1m [Neutral]	N(0.5, 0.5)
$\hat{dlogkw}_1$	dlogkwHat[1]	effect of dissociation on logkw [Acids]	N(-1, 0.125)
$\hat{dlogkw}_2$	dlogkwHat[2]	effect of dissociation on logkw [Bases]	N(-1, 0.125)
$\hat{dS1m}_1$	dS1mHat[1]	effect of dissociation on S1m [Acids]	N(0, 0.5)
$\hat{dS1m}_2$	dS1mHat[2]	effect of dissociation on S1m [Bases]	N(0, 0.5)
$\hat{ddS1}_1$	ddS1Hat[1]	effect of dissociation on dS1 [Acids]	N(0, 0.25)
$\hat{ddS1}_2$	ddS1Hat[2]	effect of dissociation on dS1 [Bases]	N(0, 0.25)

Table 1: Desciption of model paramters *(continued)*

Name	Namecode	Description	Priors
$ap\widehat{H}_1$	apH[1]	effect of pH on logkw [Acids]	N(0, 0.1)
$ap\widehat{H}_2$	apH[2]	effect of pH on logkw [Bases]	N(0, 0.1)
$d\widehat{log}kT$	dlogkTHat	effect of temperature on logkw	N(-0.087, 0.022)
ω_1	omega[1]	sd of BAV for logkw [Neutral]	N+(0, 2)
ω_2	omega[2]	sd of BAV for S1 [Neutral]	N+(0, 2)
ω_3	omega[3]	sd of BAV for dS1 [Neutral]	N+(0, 2)
ρ	rho[2,1]	correlation logkw vs S1 [Neutral]	LKJCORRN(0.75, 0.125)
ω_T	omegaT	sd of BAV for dlogkT [Neutral]	N+(0, 0.022)
κ_1	kappa[1]	sd of BAV for dlogkw [Acids and Bases]	N+(0, 0.25)
κ_2	kappa[2]	sd of BAV for dS1m [Acids and Bases]	N+(0, 0.25)
κ_3	kappa[3]	sd of BAV for ddS1 [Acids and Bases]	N+(0, 0.25)
between column differences			
$c\widehat{logkw}_c$	clogkwHat[c]	effect of column c on logkw [Neutral]	N(0, 1)
$c\widehat{S1m}_c$	cS1mHat[c]	effect of column c on S1m [Neutral]	N(0, 0.5)
$cd\widehat{S1}_c$	cdS1Hat[c]	effect of column c on dS1 [Neutral]	N(0, 0.5)
$c\beta_{c,1}$	cbeta[c,1]	effect of column c on beta[1] [Neutral]	N(0, 0.25)
$c\beta_{c,2}$	cbeta[c,2]	effect of column c on beta[2] [Neutral]	N(0, 0.25)
$cd\widehat{logkw}_{c,1}$	cdlogkwHat[c,1]	effect of column c on dlogkw [Acids]	N(0, 0.0625)
$cd\widehat{logkw}_{c,2}$	cdlogkwHat[c,2]	effect of column c on dlogkw [Bases]	N(0, 0.0625)
$cd\widehat{S1m}_{c,1}$	cdS1mHat[c,1]	effect of column c on dS1m [Acids]	N(0, 0.25)
$cd\widehat{S1m}_{c,2}$	cdS1mHat[c,2]	effect of column c on dS1m [Bases]	N(0, 0.25)
$cdd\widehat{S1}_{c,1}$	cddS1Hat[c,1]	effect of column c on ddS1 [Acids]	N(0, 0.125)
$cdd\widehat{S1}_{c,2}$	cddS1Hat[c,2]	effect of column c on ddS1 [Bases]	N(0, 0.125)
$cd\widehat{logkT}_c$	cdlogkTHat[c]	effect of column c on dlogkwT	N(0, 0.011)
$capH_{c,1}$	capH[c,1]	effect of column c on apH [Acids]	N(0, 0.05)
$capH_{c,2}$	capH[c,2]	effect of column c on apH [Bases]	N(0, 0.05)
$c\omega_{c,1}$	comega[c,1]	sd of BAV for clogkw [Neutral]	N+(0, 1)
$c\omega_{c,2}$	comega[c,2]	sd of BAV for cS1 [Neutral]	N+(0, 1)
$c\omega_{c,3}$	comega[c,3]	sd of BAV for cdS1 [Neutral]	N+(0, 1)
$c\kappa_{c,1}$	ckappa[c,1]	sd of BAV for cdlogkw [Acids and Bases]	N+(0, 0.125)
$c\kappa_{c,2}$	ckappa[c,2]	sd of BAV for cdS1m [Acids and Bases]	N+(0, 0.125)
$c\kappa_{c,3}$	ckappa[c,3]	sd of BAV for cddS1 [Acids and Bases]	N+(0, 0.125)
$c\omega_{T,c}$	comegaT[c]	sd of BAV for dlogkT	N+(0, 0.011)
S2			
$log\widehat{S2m}$	logS2mHat	typical value of S2m (log10 scale)	N(-0.7, 0.125);
$d\widehat{logS2}$	dlogS2Hat	effect of ACN on logS2m	N(1, 0.125);
pKa			

Table 1: Desciption of model paramters *(continued)*

Name	Namecode	Description	Priors
$\alpha\hat{m}_1$	alphamHat[1]	effect of MeOH on pKa [Acids]	N(2, 0.25)
$\alpha\hat{m}_2$	alphamHat[2]	effect of MeOH on pKa [Bases]	N(-1, 0.25)
$d\hat{\alpha}_1$	dalphaHat[1]	effect of ACN on alpham [Acids]	N(0, 0.125)
$d\hat{\alpha}_2$	dalphaHat[2]	effect of ACN on alpham [Bases]	N(0, 0.125)
τ_1	tau[1]	sd of BAV for pKalit	N+(0, 0.25)
τ_2	tau[2]	sd of BAV for alpham	N+(0, 0.125)
τ_3	tau[3]	sd of BAV for dalpha	N+(0, 0.125)
Residuals			
$m\sigma$	msigma	typical sd of residuals for XBridge	N+(0,1)
$s\sigma$	ssigma	sd of BAV of residuals for XBridge	N(0,1)
$clogmsigma_c$	clogmsigma[c]	effect of column c on msigma (log scale)	N+(0,125)
$c\sigma_c$	cssigma[c]	sd of BAV of residuals for column c	N+(0,125)

5.2 Stan

Multilevel modeling was performed in Stan software linked with R/ cmdstanr. For the inference we used the follwoing settings: number of iterations = 500, warmup = 1000, and number of Markov chains = 8. The reduce_sum function, which was selected to accelerate the calculations, works by parallelizing the execution of a single Stan chain across multiple cores. Convergence diagnostics were checked using Gelman−Rubin statistics and trace plots.

5.2.1 Initialize variables and parameters

5.2.2 The Stan model:

```
functions {
  // credit http://srmart.in/informative-priors-for-correlation-matrices-an-easy-approach/
  real lkj_corr_point_lower_tri_lpdf(matrix rho, real point_mu_lower,
                                     real point_scale_lower) {
    // works only for [2x2 matrix]
    real lpdf = lkj_corr_lpdf(rho | 1)
                + normal_lpdf(rho[2, 1] | point_mu_lower, point_scale_lower);
    return lpdf;
  }


  // pH and fi at a given time at column inlet
  vector gra_state(real t, vector hplcparam) {
```

```
  vector[2] sol;
  real tg = hplcparam[1];
  real td = hplcparam[2];
  real fio = hplcparam[5];
  real fik = hplcparam[6];
  real pHo = hplcparam[8];
  real alpha1 = hplcparam[9];
  real alpha2 = hplcparam[10];
  real fi;

  fi = fio + (fik - fio) / tg * (t - td);

  if (t < td) {
    fi = fio;
  } else if (t > tg + td) {
    fi = fik;
  }

  sol[1] = fi;
  sol[2] = pHo + alpha1 * fi + alpha2 * fi ^ 2;

  return sol;
}

real funlnki(vector logkw, vector apH, vector S1, real S2, vector pKaw, vector alpha,
                int nDiss, real fi, real pH) {
  real lnki;
  vector[3] logkix;
  vector[2] pHmpKa;

  logkix = log(10) *(logkw - S1*(1+S2) * fi / (1 + S2 * fi) + apH * (pH - 7));
  pHmpKa = log(10) *(pH - (pKaw + alpha * fi));

  if (nDiss == 0) {
    lnki = logkix[1];
  } else if (nDiss == 1) {
    lnki = logkix[1] +
           log1p_exp(pHmpKa[1] + logkix[2] - logkix[1]) -
           log1p_exp(pHmpKa[1]);
  } else if (nDiss == 2) {
    lnki = logkix[1] +
           log1p_exp(pHmpKa[1]+logkix[2]-logkix[1] + log1p_exp(pHmpKa[2]+logkix[3]-logkix[
           log1p_exp(pHmpKa[1] + log1p_exp(pHmpKa[2]));
```

```
  }

  return lnki;
}

vector areaandslope(real dt, real lnki1, real lnki2, real invki1, real invki2) {
  vector[2] cki_b;
  real bo;
  real cki;

  if (invki2 > 1.001 * invki1) {
    bo = (lnki1 - lnki2) / dt;
    cki = (invki2 - invki1) / bo;
  }

  else if (invki1 > 1.001 * invki2) {
    bo = (lnki2 - lnki1) / dt;
    cki = (invki1 - invki2) / bo;
  }
  else {
    bo = 0.001 / dt;
    cki = dt * (invki2 + invki1) / 2;
  }

  cki_b[1] = cki;
  cki_b[2] = bo;

  return cki_b;
}

real chromgratrapz(int steps, vector logkw, vector apH, vector S1,  real S2, vector pKaw,
                   vector alpha, int nDiss, vector hplcparam) {

  real tg = hplcparam[1];
  real td = hplcparam[2];
  real to = hplcparam[3];
  real te = hplcparam[4];

  vector[1] sol;
  real time1;
  real time2;
  vector[2] fipH1;
  vector[2] fipH2;
```

```
real lnki1;
real lnki2;
real invki1;
real invki2;
vector[2] cki_b;
real cumki1;
real cumki2;
real bo;
real tr;
real dt;

dt = tg / steps;

time1 = 0;
time2 = td;

fipH1 = gra_state(time1, hplcparam);
lnki1 = funlnki(logkw, apH, S1, S2, pKaw, alpha, nDiss, fipH1[1], fipH1[2]);
lnki2=lnki1;

invki1 = exp(-lnki1)/to;
invki2 = invki1;

cumki1 = 0;
cumki2 = td*invki1;

bo = 0.001 / td;

for (x in 1 : steps) {
  if (cumki2 >= 1) {
    continue;
  }
  time1 = time2;
  time2 += dt;
  fipH2 = gra_state(time2, hplcparam);
  lnki1 = lnki2;
  lnki2 = funlnki(logkw, apH, S1, S2, pKaw, alpha, nDiss, fipH2[1], fipH2[2]);
  invki1 = invki2;
  invki2 = exp(-lnki2)/to;
  cki_b = areaandslope(dt, lnki1, lnki2, invki1, invki2);
  cumki1 = cumki2;
  cumki2 += cki_b[1];
  bo = cki_b[2];
```

```
  }

  if (cumki2 >= 1 && cumki1==0) {
    tr = te+to+1/invki2;
  } else if (cumki2 >= 1) {
    tr = te+to+time1+log1p_exp(log((1-cumki1)*bo*to) + lnki1)/bo;
  } else if (cumki2 < 1) {
    tr = te+to+time2+(1-cumki2)/invki2;
  }

  return tr;

}

real partial_sum(array[] int ind, int start, int end,
                 vector trobs,
                 array[] int steps,
                 array[] vector hplcparam,
                 array[] int analyte,
                 array[] int column,
                 array[] int modifier,
                 array[] int R,
                 array[,] vector logkw,
                 array[,] vector apH,
                 array[, ,] vector S1,
                 array[,] real S2,
                 array[] vector pKaw,
                 array[,] vector alpha,
                 array[,] real dlogkT,
                 array[] vector sigma) {

  real lp = 0;

  for (z in start : end) {

  real y_hat = chromgratrapz(steps[z],
                             logkw[analyte[z], column[z],  : ] + dlogkT[analyte[z],column
                             apH[analyte[z], column[z], : ],
                             S1[analyte[z], modifier[z], column[z],  : ],
                             S2[modifier[z], column[z]],
                             pKaw[analyte[z],  : ],
                             alpha[analyte[z],modifier[z],  : ],
                             R[analyte[z]],
```

```
                                hplcparam[z]);

     lp = lp + student_t_lpdf(trobs[z] | 3, y_hat,  sigma[analyte[z],column[z]]);

    }
    return lp;
  }
}

data {
  int nAnalytes;            // number of analytes
  int nColumns;             // number of columns
  int nModifiers;           // number of org. modifiers
  int nObs;                 // number of observations
  array[nObs] int analyte;   // analyte indexes
  array[nObs] int column;   // column indexes
  array[nObs] int modifier;  // modifier indexes
  array[nObs] int<lower=1> steps;  // steps for gradient retention time aproximation
  array[nObs] vector[12] hplcparam; // [tg, td, to, te, fio, fik, org modifier, pHo, alpha1,
  vector[nAnalytes] logPobs;
  int<lower=0, upper=2> maxR; //
  array[nAnalytes] int<lower=0, upper=2> R;
  int<lower=1> nGroupsA;
  int<lower=1> nGroupsB;
  vector[nGroupsA] pKaslitA;
  vector[nGroupsB] pKaslitB;
  array[nGroupsA,2] int idxGroupsA;
  array[nGroupsB,2] int idxGroupsB;

  vector[nObs] trobs;                   // observed retention factors
  int<lower=0, upper=1> run_estimation; // 0 for prior predictive, 1 for estimation
}
transformed data {
  int grainsize = 1;
  array[nObs] int ind = rep_array(1, nObs);
}

parameters {

  real logkwHat; // typical logkw [Neutral]
  real S1mHat;   // effect of MeOH on logkw [Neutral]
  real dS1Hat;   // effect of ACN on S1m [Neutral]
  array[2] real dlogkwHat; // effect of dissociation on logkw [Acids, Bases]
```

```
array[2] real dS1mHat;   // effect of dissociation on S1m [Acids, Bases]
array[2] real ddS1Hat;   // effect of dissociation on dS1 [Acids, Bases]
real logS2mHat; // typical value of S2m (log10 scale)
real dlogS2Hat; // effect of ACN on logS2m
vector[2] beta; // effect of logP on logkw and S1m
real dlogkTHat; // effect of temperature on logkw
vector[2] apH;  // effect of pH on logkw [Acids, Bases]
array[2] real alphamHat;   // effect of MeOH on pKa [Acids, Bases]
array[2] real dalphaHat;   // effect of ACN on alpham [Acids, Bases]
vector<lower=0>[3] tau;   // sd for between analyte variability of pKa's
vector<lower=0>[3] omega; // sd of BAV [logkw,S1m, dS1]
corr_matrix[2] rho;      // correlation matrix [logkw vs. S1m]
real<lower=0> omegaT;     // sd of BAV [dlogkT]
vector<lower=0>[3] kappa; // sd of BAV [dlogkw,dS1m,ddS1]

// 2nd column
vector[nColumns-1] clogkwHat;          // effect of column on logkw [Neutral]
vector[nColumns-1] cS1mHat;            // effect of column on S1m [Neutral]
vector[nColumns-1] cdS1Hat;            // effect of column on dS1 [Neutral]
matrix[nColumns-1,2] cdlogkwHat;      // effect of column on logkw [Acids, Bases]
matrix[nColumns-1,2] cdS1mHat;        // effect of column on dS1m [Acids, Bases]
matrix[nColumns-1,2] cddS1Hat;        // effect of column on ddS1 [Acids, Bases]
matrix[nColumns-1,2]  cbeta;
vector[nColumns-1]   cdlogkTHat;       // effect of column on dlogkTHat
matrix[nColumns-1,2] capH;             // effect of column on apH  [Acids, Bases]
matrix<lower=0>[nColumns-1,3] comega;  // sd of BAV [clogkw,cS1m,cdS1]
vector<lower=0>[nColumns-1]  comegaT;  // sd of BAV [cdlogkT]
matrix<lower=0>[nColumns-1,3] ckappa;  // sd of BAV [cdlogkw,cdS1m,cddS1]
cholesky_factor_corr[nColumns-1] corr_L;   // cholesky factor correlation matrix

// 1st column
array[nAnalytes] vector[2] paramN;
vector[nAnalytes] dS1N;
vector[nAnalytes] dlogkT;
vector[nGroupsA] dlogkwA;
vector[nGroupsB] dlogkwB;
vector[nGroupsA] dS1mA;
vector[nGroupsB] dS1mB;
vector[nGroupsA] dS1A;
vector[nGroupsB] dS1B;

// 2nd column
matrix[nColumns-1, nAnalytes] etaclogkwNStd;
```

```
  array[nColumns-1] vector[nAnalytes] etacS1mN;
  array[nColumns-1] vector[nAnalytes] etacdS1N;
  array[nColumns-1] vector[nAnalytes] etacdlogkT;
  array[nColumns-1] vector[nGroupsA] etacdlogkwA;
  array[nColumns-1] vector[nGroupsB] etacdlogkwB;
  array[nColumns-1] vector[nGroupsA] etacdS1mA;
  array[nColumns-1] vector[nGroupsB] etacdS1mB;
  array[nColumns-1] vector[nGroupsA] etacdS1A;
  array[nColumns-1] vector[nGroupsB] etacdS1B;

  // Dissociation
  vector[nGroupsA] pKawA;
  vector[nGroupsB] pKawB;
  vector[nGroupsA] etaalphamA;
  vector[nGroupsB] etaalphamB;
  vector[nGroupsA] etadalphaA;
  vector[nGroupsB] etadalphaB;

  // residual variability for the 1st and 2nd column
  real<lower=0> msigma;   // typical sigma for the 1st column]
  real<lower=0> ssigma;
  vector[nAnalytes] logsigma;

  vector[nColumns-1] clogmsigma; // effect of column on log(msigma)]
  vector<lower=0>[nColumns-1] cssigma; ; //sd of residual [1st,2nd column]
  array[nColumns-1] vector[nAnalytes] etaclogsigma;
}
transformed parameters {

  cov_matrix[2] Omega;

  array[nAnalytes] vector[3] miu;
  array[nAnalytes,3] vector[nColumns-1] cmiu;
  array[nAnalytes,nColumns]   vector[maxR + 1] logkwx;
  array[nAnalytes,nModifiers, nColumns] vector[maxR + 1] S1x;
  array[nModifiers,nColumns]  real S2x;
  array[nAnalytes,nColumns]   vector[maxR + 1] apHx;
  array[nAnalytes,nModifiers] vector[maxR] alphax;
  array[nAnalytes, nColumns]  real dlogkTx;
  array[nAnalytes] vector[maxR] pKawx;
  array[nAnalytes] vector[nColumns] sigmax;

  array[nColumns-1] vector[nAnalytes] clogkwN;
```

```
matrix[nColumns-1, nAnalytes] etaclogkwN;
array[nColumns-1] vector[nAnalytes] cS1mN;
array[nColumns-1] vector[nAnalytes] cdS1N;
array[nColumns-1] vector[nGroupsA] cdlogkwA;
array[nColumns-1] vector[nGroupsB] cdlogkwB;
array[nColumns-1] vector[nGroupsA] cdS1mA;
array[nColumns-1] vector[nGroupsB] cdS1mB;
array[nColumns-1] vector[nGroupsA] cdS1A;
array[nColumns-1] vector[nGroupsB] cdS1B;
 array[nColumns-1] vector[nAnalytes] cdlogkT;
 array[nColumns-1] vector[nAnalytes] clogsigma;
vector[nGroupsA] alphamA;
vector[nGroupsB] alphamB;
vector[nGroupsA] dalphaA;
vector[nGroupsB] dalphaB;

Omega = quad_form_diag(rho, omega[1 : 2]); // diag_matrix(omega) * rho * diag_matrix(omega)

for (i in 1 : nAnalytes) {
  miu[i, 1] = logkwHat + beta[1] * (logPobs[i] - 2.2);
  miu[i, 2] = S1mHat   + beta[2] * (logPobs[i] - 2.2);
  miu[i, 3] = dS1Hat;
}

 for (i in 1 : nAnalytes) {
  for (c in 1 : (nColumns-1)) {
  cmiu[i, 1, c] = clogkwHat[c] + cbeta[c,1] * (logPobs[i] - 2.2);
  cmiu[i, 2, c] = cS1mHat[c]   + cbeta[c,2] * (logPobs[i] - 2.2);
  cmiu[i, 3, c] = cdS1Hat[c];
}}

// Matt's trick to use unit scale
etaclogkwN = diag_pre_multiply(comega[,1], corr_L * etaclogkwNStd);

for (i in 1 : nAnalytes) {
 logkwx[i, 1, : ]  =  paramN[i,1]*[1,1,1]';
for (c in 1 : (nColumns-1)) {
 clogkwN[c,i] = cmiu[i, 1, c]  + etaclogkwN[c,i];
 logkwx[i, c+1, : ]  =  (paramN[i,1]+clogkwN[c,i])*[1,1,1]';
}}

for (d in 1 : nGroupsA) {
  logkwx[idxGroupsA[d,1], 1, 1+idxGroupsA[d,2]] += dlogkwA[d];
```

```
  if (idxGroupsA[d,2]==1) {
 logkwx[idxGroupsA[d,1], 1, 3] += dlogkwA[d];
 }
  for (c in 1 : (nColumns-1)) {
 cdlogkwA[c,d] = cdlogkwHat[c,1] + ckappa[c,1]*etacdlogkwA[c,d];
 logkwx[idxGroupsA[d,1], c+1, 1+idxGroupsA[d,2]] += dlogkwA[d]+cdlogkwA[c,d];
 if (idxGroupsA[d,2]==1) {
 logkwx[idxGroupsA[d,1], c+1, 3] += dlogkwA[d]+cdlogkwA[c,d];
 }}}

for (d in 1 : nGroupsB) {
  logkwx[idxGroupsB[d,1], 1, idxGroupsB[d,2]] += dlogkwB[d];
  if (idxGroupsB[d,2]==2) {
  logkwx[idxGroupsB[d,1], 1, 1] += dlogkwB[d];
  }
  for (c in 1 : (nColumns-1)) {
  cdlogkwB[c,d] = cdlogkwHat[c,2]+ ckappa[c,1]*etacdlogkwB[c,d];
  logkwx[idxGroupsB[d,1], c+1, idxGroupsB[d,2]] += dlogkwB[d]+cdlogkwB[c,d];
  if (idxGroupsB[d,2]==2) {
  logkwx[idxGroupsB[d,1], c+1, 1] += dlogkwB[d]+cdlogkwB[c,d];
  }}}

for (i in 1 : nAnalytes) {
  apHx[i, 1, : ] = [0,0,0]';
  for (c in 1 : (nColumns-1)) {
  apHx[i, c+1, : ] = [0,0,0]';
}}

for (d in 1 : nGroupsA) {
  apHx[idxGroupsA[d,1], 1, 1+idxGroupsA[d,2]] += apH[1];
if (idxGroupsA[d,2]==1) {
 apHx[idxGroupsA[d,1], 1, 3] += apH[1];
 }
for (c in 1 : (nColumns-1)) {
 apHx[idxGroupsA[d,1], c+1, 1+idxGroupsA[d,2]] += apH[1]+capH[c,1];
if (idxGroupsA[d,2]==1) {
 apHx[idxGroupsA[d,1], c+1, 3] += apH[1]+capH[c,1];
 }}}

for (d in 1 : nGroupsB) {
  apHx[idxGroupsB[d,1], 1, idxGroupsB[d,2]] += apH[2];
       if (idxGroupsB[d,2]==2) {
  apHx[idxGroupsB[d,1], 1, 1] += apH[2];
```

```
 }
 for (c in 1 : (nColumns-1)) {
  apHx[idxGroupsB[d,1], c+1, idxGroupsB[d,2]] += apH[2]+capH[c,2];
  if (idxGroupsB[d,2]==2) {
  apHx[idxGroupsB[d,1], c+1, 1] += apH[2]+capH[c,2];
 }}}


for (i in 1 : nAnalytes) {
 S1x[i, 1, 1, : ]  =  paramN[i,2]*[1,1,1]';
 S1x[i, 2, 1, : ]  =  paramN[i,2]*[1,1,1]' + dS1N[i];
for (c in 1 : (nColumns-1)) {
 cS1mN[c,i] = cmiu[i,2,c] + comega[c,2]*etacS1mN[c,i];
 cdS1N[c,i] = cmiu[i,3,c] + comega[c,3]*etacdS1N[c,i];
 S1x[i, 1, c+1, : ]  =  (paramN[i,2]+cS1mN[c,i])*[1,1,1]' ;
 S1x[i, 2, c+1, : ]  =  (paramN[i,2]+cS1mN[c,i])*[1,1,1]' + (dS1N[i]+cdS1N[c,i]);
}}

for (d in 1 : nGroupsA) {
 S1x[idxGroupsA[d,1], 1, 1, 1+idxGroupsA[d,2]] += dS1mA[d];
 S1x[idxGroupsA[d,1], 2, 1, 1+idxGroupsA[d,2]] += dS1mA[d]+dS1A[d];

if (idxGroupsA[d,2]==1) {
 S1x[idxGroupsA[d,1], 1, 1, 3] += dS1mA[d];
 S1x[idxGroupsA[d,1], 2, 1, 3] += dS1mA[d]+dS1A[d];
 }

for (c in 1 : (nColumns-1)) {
 cdS1mA[c,d]= cdS1mHat[c,1] + ckappa[c,2]*etacdS1mA[c,d];
 cdS1A[c,d] = cddS1Hat[c,1] + ckappa[c,3]*etacdS1A[c,d];
 S1x[idxGroupsA[d,1], 1, c+1, 1+idxGroupsA[d,2]] += dS1mA[d]        +cdS1mA[c,d];
 S1x[idxGroupsA[d,1], 2, c+1, 1+idxGroupsA[d,2]] += dS1mA[d]+dS1A[d]+cdS1mA[c,d]+cdS1A[c,d]
if (idxGroupsA[d,2]==1) {
 S1x[idxGroupsA[d,1], 1, c+1, 3] += dS1mA[d]        +cdS1mA[c,d];
 S1x[idxGroupsA[d,1], 2, c+1, 3] += dS1mA[d]+dS1A[d]+cdS1mA[c,d]+cdS1A[c,d];
 }}}

for (d in 1 : nGroupsB) {
  S1x[idxGroupsB[d,1], 1, 1, idxGroupsB[d,2]] += dS1mB[d];
  S1x[idxGroupsB[d,1], 2, 1, idxGroupsB[d,2]] += dS1mB[d]+dS1B[d];
  if (idxGroupsB[d,2]==2) {
  S1x[idxGroupsB[d,1], 1, 1, 1] += dS1mB[d];
  S1x[idxGroupsB[d,1], 2, 1, 1] += dS1mB[d]+dS1B[d];
```

```
 }
  for (c in 1 : (nColumns-1)) {
  cdS1mB[c,d]= cdS1mHat[c,2] + ckappa[c,2]*etacdS1mB[c,d];
  cdS1B[c,d] = cddS1Hat[c,2] + ckappa[c,3]*etacdS1B[c,d];

  S1x[idxGroupsB[d,1], 1, c+1, idxGroupsB[d,2]] += dS1mB[d]        +cdS1mB[c,d];
  S1x[idxGroupsB[d,1], 2, c+1, idxGroupsB[d,2]] += dS1mB[d]+dS1B[d]+cdS1mB[c,d]+cdS1B[c,d]
if (idxGroupsB[d,2]==2) {
  S1x[idxGroupsB[d,1], 1, c+1, 1] += dS1mB[d]        +cdS1mB[c,d];
  S1x[idxGroupsB[d,1], 2, c+1, 1] += dS1mB[d]+dS1B[d]+cdS1mB[c,d]+cdS1B[c,d];
 }}}

 S2x[1,1] = 10^(logS2mHat);
 S2x[2,1] = 10^(logS2mHat + dlogS2Hat);

 for (c in 1 : (nColumns-1)) {
 S2x[1,c+1] = S2x[1,1];
 S2x[2,c+1] = S2x[2,1];
 }

for (i in 1 : nAnalytes) {
 dlogkTx[i, 1]  = dlogkT[i];
for (c in 1 : (nColumns-1)) {
 cdlogkT[c,i] = cdlogkTHat[c]+comegaT[c]*etacdlogkT[c,i];
 dlogkTx[i, c+1]  =  dlogkT[i] + cdlogkT[c,i];
}}

for (i in 1 : nAnalytes) {
 pKawx[i, : ]  = [0,0]';
 alphax[i, 1, : ]  = [0,0]';
 alphax[i, 2, : ]  = [0,0]';
}

for (d in 1 : nGroupsA) {

alphamA[d] = alphamHat[1] + tau[2]*etaalphamA[d];
dalphaA[d] = dalphaHat[1] + tau[3]*etadalphaA[d];

pKawx[idxGroupsA[d,1], idxGroupsA[d,2]] = pKawA[d];
alphax[idxGroupsA[d,1], 1, idxGroupsA[d,2]]= alphamA[d];
alphax[idxGroupsA[d,1], 2, idxGroupsA[d,2]]= alphamA[d]+dalphaA[d];
 }
```

```
  for (d in 1 : nGroupsB) {
    alphamB[d] = alphamHat[2] + tau[2]*etaalphamB[d];
    dalphaB[d] = dalphaHat[2] + tau[3]*etadalphaB[d];
    pKawx[idxGroupsB[d,1], idxGroupsB[d,2]] = pKawB[d];
    alphax[idxGroupsB[d,1], 1, idxGroupsB[d,2]]= alphamB[d];
    alphax[idxGroupsB[d,1], 2, idxGroupsB[d,2]]= alphamB[d]+dalphaB[d];
   }

  for (i in 1 : nAnalytes) {
   sigmax[i, 1] = exp(logsigma[i]);
   for (c in 1 : (nColumns-1)) {
   clogsigma[c,i] = clogmsigma[c] + cssigma[c]*etaclogsigma[c,i];
   sigmax[i, c+1] = exp(logsigma[i]+clogsigma[c,i]);
  }}
}

model {
  logkwHat ~ normal(2.2, 2);
  S1mHat ~ normal(4, 1);
  dS1Hat ~ normal(1, 0.5);
  dlogkwHat ~ normal(-1, 0.125);
  dS1mHat   ~ normal(0, 0.5);
  ddS1Hat   ~ normal(0, 0.25);
  logS2mHat ~ normal(-0.7, 0.125);
  dlogS2Hat ~ normal(1, 0.125);
  beta[{1}] ~ normal(1, 0.125);
  beta[{2}] ~ normal(0.5, 0.5);
  dlogkTHat ~ normal(-0.087, 0.022);
  apH       ~ normal(0, 0.1);

  alphamHat[{1}] ~ normal(2, 0.25);
  alphamHat[{2}] ~ normal(-1, 0.25);
  dalphaHat ~ normal(0, 0.125);
  tau[{1}] ~ normal(0, 0.25);
  tau[{2,3}] ~ normal(0, 0.125);
  omega ~ normal(0, 2);
  rho ~ lkj_corr_point_lower_tri(0.75, 0.125);
  omegaT ~ normal(0, 0.022);
  kappa ~ normal(0, 0.25);

  clogkwHat ~ normal(0, 1);
  cS1mHat ~ normal(0, 0.5);
  cdS1Hat ~ normal(0, 0.25);
```

```
 to_vector(cdlogkwHat) ~ normal(0, 0.0625);
 to_vector(cdS1mHat) ~ normal(0, 0.25);
 to_vector(cddS1Hat) ~ normal(0, 0.125);
 to_vector(cbeta) ~ normal(0, 0.25);
 cdlogkTHat ~ normal(0, 0.011);
 to_vector(capH) ~ normal(0, 0.05);
 to_vector(comega) ~ normal(0, 0.5);
 comegaT ~ normal(0, 0.011);
 to_vector(ckappa) ~ normal(0, 0.125);

 corr_L~lkj_corr_cholesky(2.0);

 for (i in 1 : nAnalytes) {
 paramN[i] ~ multi_normal(miu[i,1:2], Omega);
 }

 dS1N ~ normal(miu[,3], omega[3]);
 dlogkT ~ normal(dlogkTHat, omegaT);
 dlogkwA ~ normal(dlogkwHat[1], kappa[1]);
 dlogkwB ~ normal(dlogkwHat[2], kappa[1]);
 dS1mA ~ normal(dS1mHat[1], kappa[2]);
 dS1mB ~ normal(dS1mHat[2], kappa[2]);
 dS1A ~ normal(ddS1Hat[1], kappa[3]);
 dS1B ~ normal(ddS1Hat[2], kappa[3]);

 to_vector(etaclogkwNStd) ~ normal(0, 1);

for (c in 1 : (nColumns-1)) {
 etacS1mN[c] ~ normal(0,1);
 etacdS1N[c] ~ normal(0,1);
 etacdlogkT[c] ~ normal(0,1);
 etacdlogkwA[c] ~ normal(0,1);
 etacdlogkwB[c] ~ normal(0,1);
 etacdS1mA[c] ~ normal(0,1);
 etacdS1mB[c] ~ normal(0,1);
 etacdS1A[c] ~ normal(0,1);
 etacdS1B[c] ~ normal(0,1);
}

 pKawA ~ normal(pKaslitA, tau[1]);
 pKawB ~ normal(pKaslitB, tau[1]);
 etaalphamA ~ normal(0,1);
 etaalphamB ~ normal(0,1);
```

```
  etadalphaA ~ normal(0,1);
  etadalphaB ~ normal(0,1);

  msigma ~ normal(0,1);
  ssigma ~ normal(0,1);
  logsigma  ~ normal(log(msigma),ssigma);

  clogmsigma ~ normal(0,0.125);
  cssigma ~ normal(0,0.125);

  for (c in 1 : (nColumns-1)) {
  etaclogsigma[c]  ~ normal(0,1);
  }

  if (run_estimation == 1) {

    target += reduce_sum(partial_sum, ind, grainsize, trobs, steps,
                         hplcparam, analyte, column, modifier, R, logkwx, apHx,
                         S1x, S2x, pKawx, alphax, dlogkTx, sigmax);
  }
}

generated quantities {
   corr_matrix[nColumns-1] crho;
   crho = corr_L * corr_L';
}
```

5.2.3 Fitting the model

Compile the model:

The optimization was used for initial testing:

For local computations one can use cmdstanr:

We performed computations at the Academic Computer Center in Gdańsk, Tryton Cluster. In this case:

1. we dumped the necessary data to .json format
2. we run the model using the batch file:
3. After calculations, we loaded the output files using cmdstanr. The files are accessible through stan output files.

4. finally we checked the diagnostics of Monte Carlo inferences based on the Stan documentation described here .

The diagnostics are reasonable given model complexity. Output copied here to save time:

6 Results

6.1 Summary of model parameters (table):

The summary can also be extracted for individual parameters. Here presented for analyte 9 (Baclofen)

6.2 Trace plots

Several trace plots are shown.

Table 2: **?(caption)**

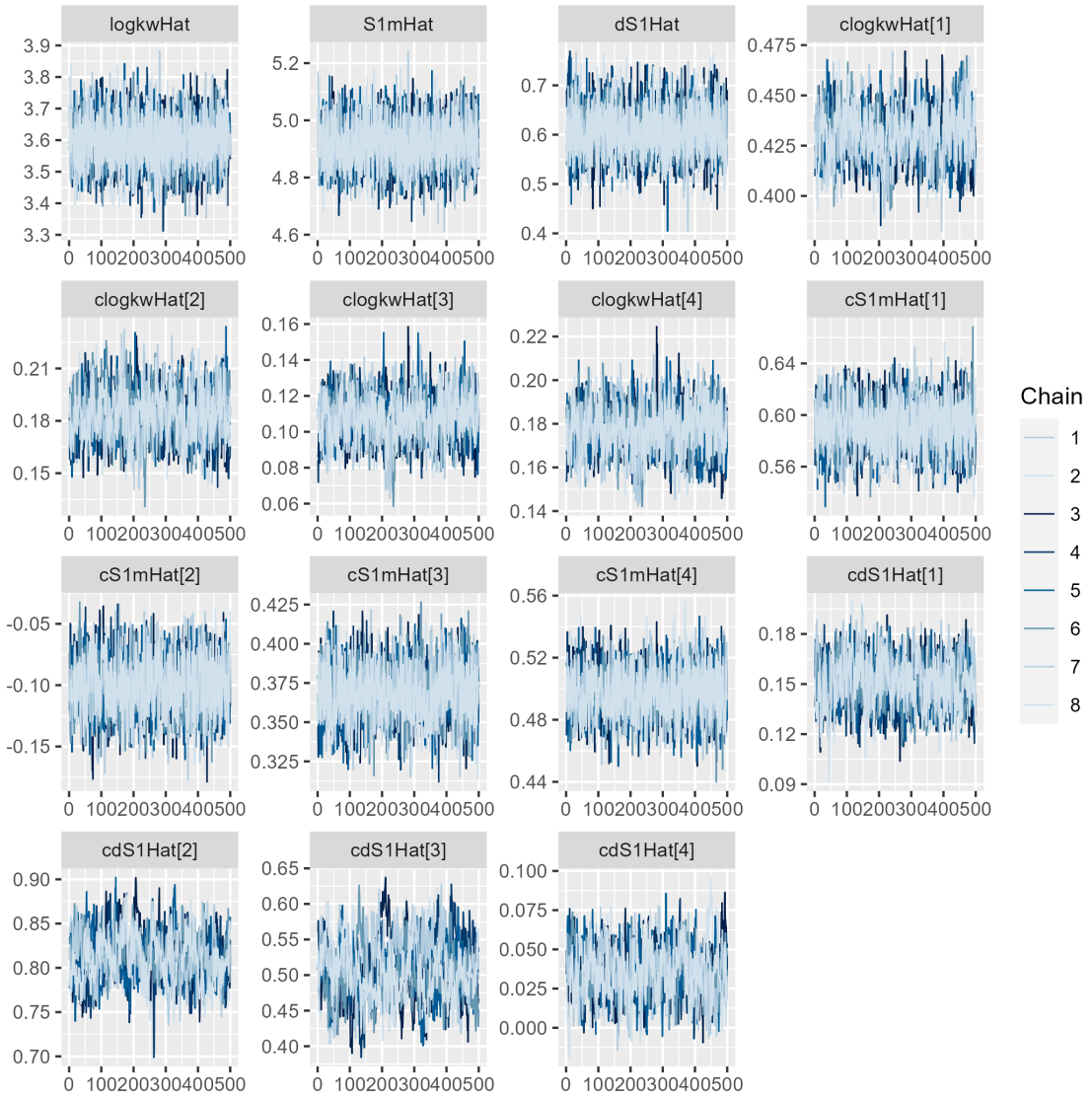
variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
logkwHat	3.60	3.60	0.08	0.08	3.47	3.72	1.00	5947	3030
S1mHat	4.92	4.92	0.08	0.08	4.79	5.05	1.00	5251	3063
dS1Hat	0.61	0.61	0.05	0.05	0.53	0.69	1.00	3259	2365
dlogkwHat[1]	-0.79	-0.79	0.07	0.07	-0.91	-0.67	1.00	7672	3143
dlogkwHat[2]	-0.97	-0.97	0.05	0.05	-1.05	-0.88	1.00	5300	2998
dS1mHat[1]	0.17	0.17	0.12	0.12	-0.03	0.36	1.00	4231	3375
dS1mHat[2]	0.12	0.11	0.07	0.07	0.00	0.24	1.00	2482	2783
ddS1Hat[1]	0.28	0.28	0.08	0.08	0.15	0.41	1.00	6200	3217
ddS1Hat[2]	-0.67	-0.67	0.05	0.06	-0.76	-0.58	1.00	6993	3444
logS2mHat	-0.31	-0.31	0.01	0.01	-0.33	-0.28	1.02	432	986
dlogS2Hat	0.42	0.42	0.01	0.01	0.41	0.43	1.01	552	1093
beta[1]	0.84	0.84	0.04	0.04	0.77	0.91	1.00	7278	3170
beta[2]	0.51	0.51	0.05	0.05	0.43	0.58	1.00	3976	2814
dlogkTHat	-0.09	-0.09	0.00	0.00	-0.09	-0.08	1.00	4047	2966
apH[1]	-0.03	-0.03	0.00	0.00	-0.03	-0.03	1.00	4583	3724
apH[2]	0.08	0.08	0.00	0.00	0.08	0.08	1.00	3114	3288
omega[1]	0.92	0.92	0.06	0.06	0.83	1.02	1.00	5132	2669
omega[2]	0.93	0.93	0.06	0.06	0.84	1.03	1.00	4993	3096
omega[3]	0.55	0.55	0.03	0.03	0.50	0.61	1.00	8536	3214
omegaT	0.03	0.03	0.00	0.00	0.03	0.04	1.00	7750	3273
kappa[1]	0.59	0.58	0.03	0.03	0.53	0.65	1.00	4404	3301
kappa[2]	0.69	0.69	0.05	0.05	0.62	0.77	1.00	3416	3502
kappa[3]	0.55	0.55	0.04	0.04	0.49	0.61	1.00	3604	3242
rho[2,1]	0.87	0.87	0.02	0.02	0.83	0.91	1.00	3500	3223
msigma	0.39	0.39	0.03	0.03	0.35	0.44	1.00	9144	3009
ssigma	0.81	0.81	0.05	0.05	0.74	0.90	1.00	12125	2859

variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
alphamHat[1]	2.22	2.22	0.15	0.15	1.96	2.47	1.00	1721	2559
alphamHat[2]	-1.35	-1.35	0.10	0.10	-1.51	-1.18	1.01	1282	2346
dalphaHat[1]	0.22	0.22	0.10	0.10	0.06	0.38	1.00	2171	2544
dalphaHat[2]	-0.20	-0.20	0.07	0.07	-0.32	-0.08	1.01	1514	2412
tau[1]	0.88	0.88	0.05	0.05	0.81	0.97	1.00	6372	3337
tau[2]	0.96	0.96	0.06	0.06	0.87	1.06	1.00	2193	2683
tau[3]	0.79	0.79	0.05	0.05	0.71	0.88	1.00	1967	2870

variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
clogkwHat[1]	0.43	0.43	0.01	0.01	0.41	0.45	1.01	554	1048
clogkwHat[2]	0.18	0.18	0.01	0.01	0.16	0.21	1.01	734	1437
clogkwHat[3]	0.11	0.11	0.01	0.01	0.09	0.13	1.01	612	1192
clogkwHat[4]	0.18	0.18	0.01	0.01	0.16	0.19	1.01	726	1350
cS1mHat[1]	0.59	0.59	0.02	0.02	0.56	0.62	1.00	2127	2781
cS1mHat[2]	-0.10	-0.10	0.02	0.02	-0.14	-0.06	1.00	2010	2704
cS1mHat[3]	0.37	0.37	0.02	0.02	0.34	0.40	1.00	1308	3004
cS1mHat[4]	0.50	0.50	0.02	0.02	0.47	0.52	1.00	1837	3007
cdS1Hat[1]	0.15	0.15	0.01	0.01	0.13	0.17	1.01	928	1643
cdS1Hat[2]	0.81	0.81	0.03	0.02	0.77	0.85	1.01	379	872
cdS1Hat[3]	0.51	0.51	0.04	0.04	0.44	0.58	1.05	153	462
cdS1Hat[4]	0.04	0.04	0.02	0.02	0.01	0.06	1.00	607	1094
cdlogkwHat[1,1]	0.04	0.04	0.02	0.02	0.02	0.07	1.00	2224	2870
cdlogkwHat[2,1]	-0.05	-0.05	0.02	0.02	-0.08	-0.02	1.00	1549	2160
cdlogkwHat[3,1]	-0.03	-0.03	0.02	0.02	-0.06	0.00	1.00	1668	2740
cdlogkwHat[4,1]	0.04	0.04	0.02	0.02	0.02	0.07	1.00	1319	2124

Table 3: **?(caption)**

variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
logkwx[3,1,1]	1.33	1.33	0.03	0.03	1.28	1.38	1.00	6062	3190
logkwx[3,1,2]	0.68	0.68	0.07	0.07	0.58	0.79	1.00	5550	3342
logkwx[3,1,3]	0.68	0.68	0.07	0.07	0.58	0.79	1.00	5550	3342
logkwx[3,2,1]	1.64	1.64	0.04	0.04	1.58	1.70	1.00	6117	3234
logkwx[3,2,2]	1.04	1.04	0.06	0.06	0.94	1.14	1.00	6694	3794
logkwx[3,2,3]	1.04	1.04	0.06	0.06	0.94	1.14	1.00	6694	3794
S1x[3,1,1,1]	3.78	3.79	0.31	0.31	3.28	4.28	1.00	5248	2966
S1x[3,1,1,2]	4.74	4.72	0.71	0.71	3.59	5.95	1.00	6928	3011
S1x[3,1,1,3]	4.74	4.72	0.71	0.71	3.59	5.95	1.00	6928	3011
S1x[3,2,1,1]	3.35	3.35	0.26	0.27	2.92	3.78	1.00	5914	3292
S1x[3,2,1,2]	5.74	5.74	0.63	0.63	4.73	6.78	1.00	6900	3283
S1x[3,2,1,3]	5.74	5.74	0.63	0.63	4.73	6.78	1.00	6900	3283
S1x[3,1,2,1]	4.53	4.53	0.31	0.32	4.02	5.03	1.00	5363	3001
S1x[3,1,2,2]	5.48	5.46	0.72	0.71	4.30	6.69	1.00	7097	3287
S1x[3,1,2,3]	5.48	5.46	0.72	0.71	4.30	6.69	1.00	7097	3287
S1x[3,2,2,1]	4.20	4.20	0.27	0.28	3.76	4.62	1.00	5462	3331
S1x[3,2,2,2]	6.62	6.62	0.64	0.64	5.58	7.68	1.00	6955	3383
S1x[3,2,2,3]	6.62	6.62	0.64	0.64	5.58	7.68	1.00	6955	3383
apHx[3,1,1]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
apHx[3,1,2]	-0.03	-0.03	0.00	0.00	-0.03	-0.03	1.00	4583	3724
apHx[3,1,3]	-0.03	-0.03	0.00	0.00	-0.03	-0.03	1.00	4583	3724
pKawx[3,1]	6.74	6.74	0.08	0.08	6.61	6.86	1.00	8439	3301
pKawx[3,2]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
alphax[3,1,1]	2.14	2.14	0.90	0.90	0.67	3.64	1.00	8395	3357
alphax[3,1,2]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
alphax[3,2,1]	2.49	2.47	1.14	1.11	0.59	4.39	1.00	7846	3292
alphax[3,2,2]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
S2x[1,1]	0.49	0.49	0.02	0.02	0.47	0.52	1.02	432	986
S2x[2,1]	1.30	1.30	0.03	0.03	1.26	1.34	1.02	355	754
sigmax[3,1]	0.39	0.39	0.03	0.03	0.34	0.45	1.00	4626	3538
sigmax[3,2]	0.45	0.45	0.04	0.04	0.38	0.52	1.00	2663	3556



6.3 Summary of model parameters (figures)

First plot presents parameter specific for XBridge Shield RP18 column and parameters common for all the columns:

Column effects relative to XBridge Shield RP18 column:

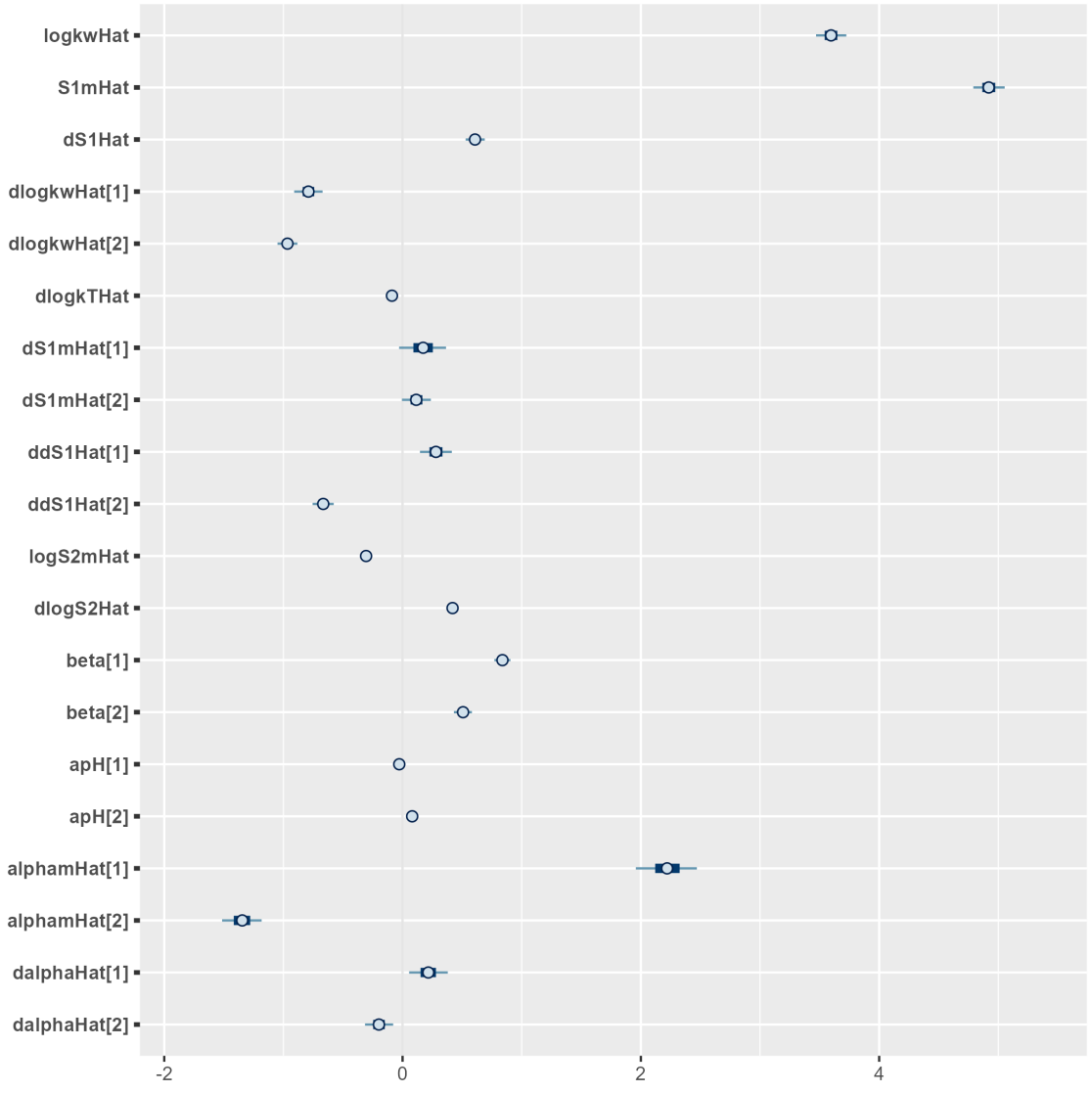


Figure 1: **?(caption)**

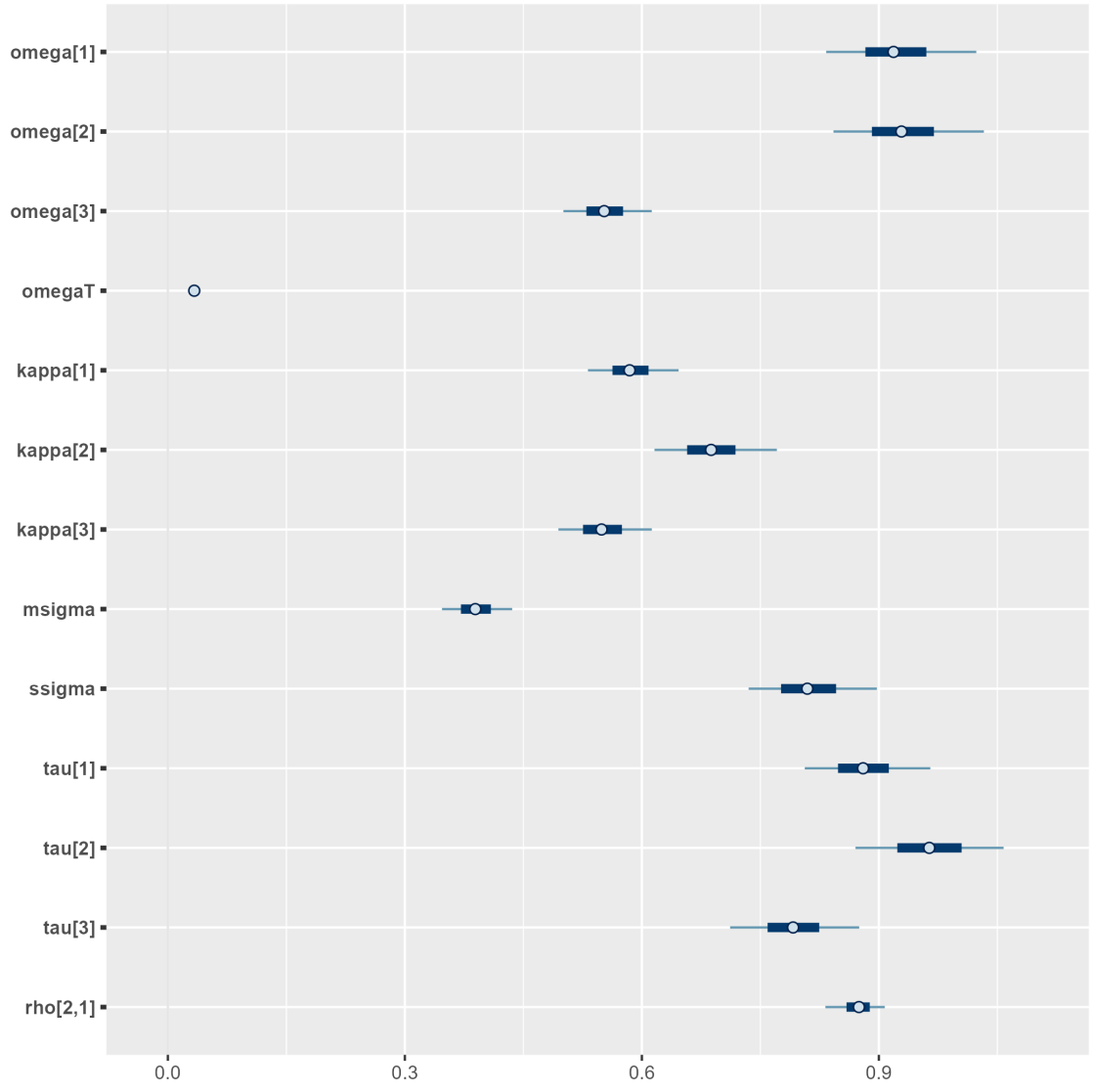
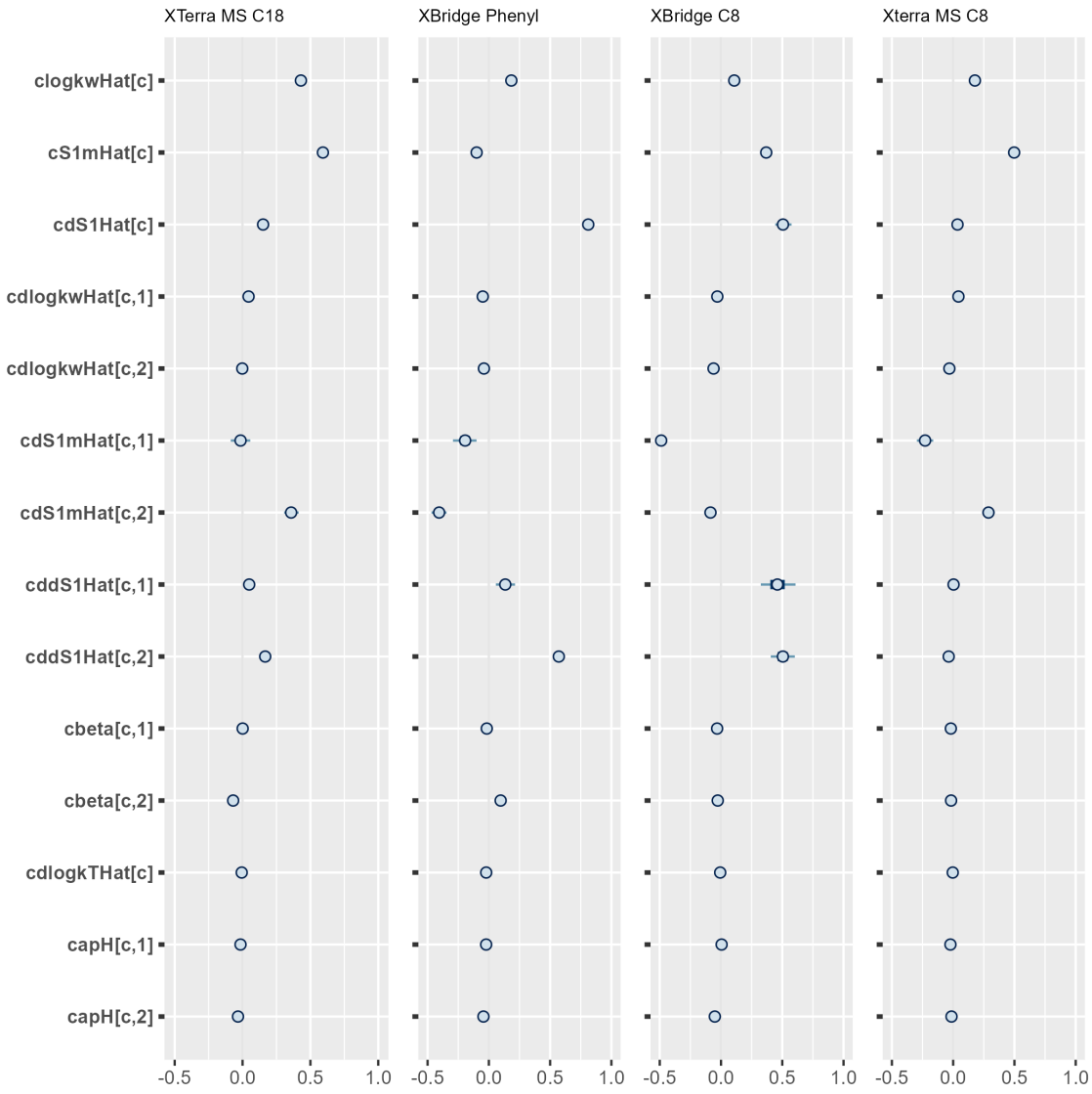
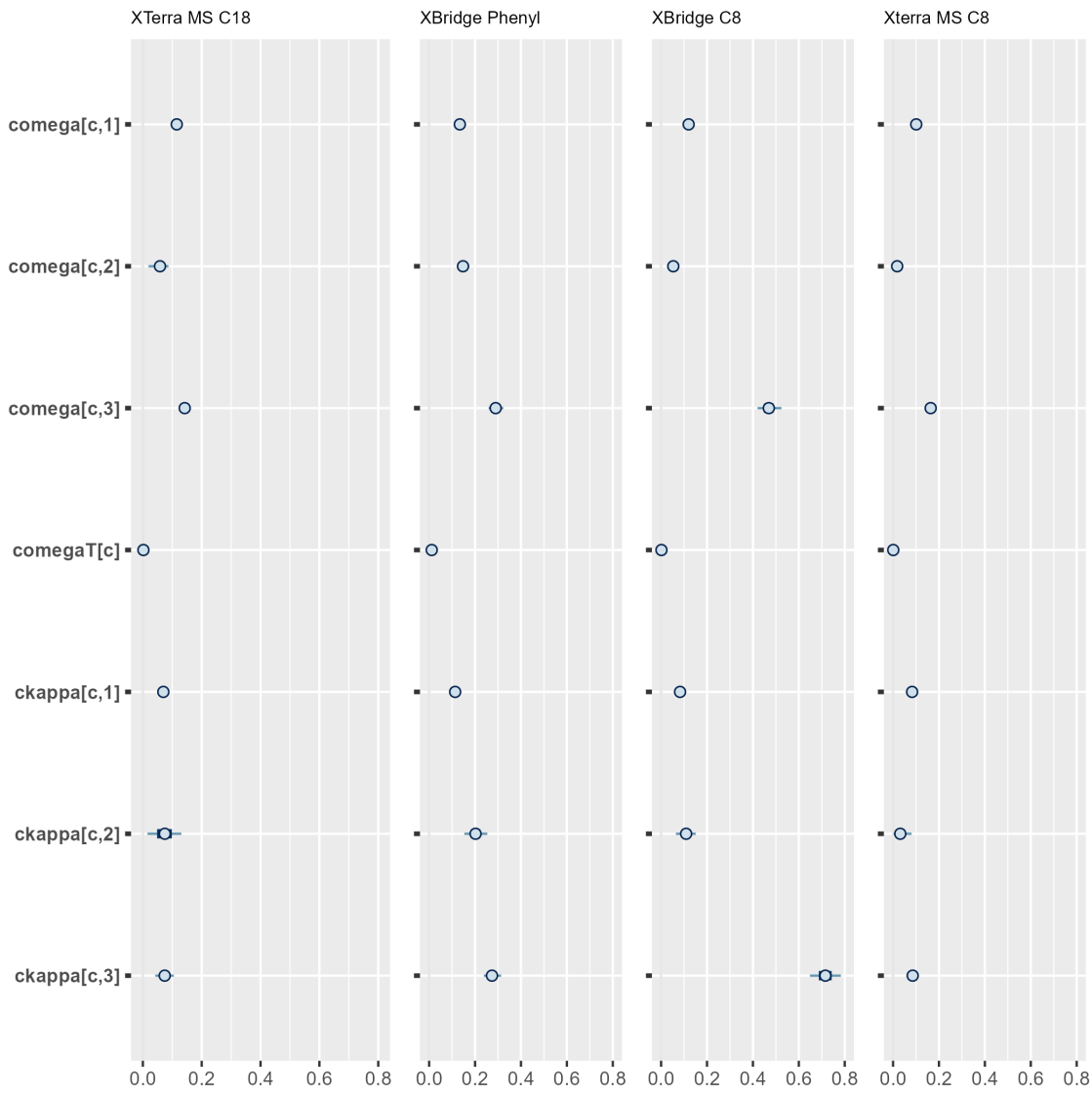


Figure 2: **?(caption)**


XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8
clogkwHat[c]
cS1mHat[c]
cdS1Hat[c]
cdlogkwHat[c,1]
cdlogkwHat[c,2]
cdS1mHat[c,1]
cdS1mHat[c,2]
cddS1Hat[c,1]
cddS1Hat[c,2]
cbeta[c,1]
cbeta[c,2]
cdlogkTHat[c]
capH[c,1]
capH[c,2]
-0.5
0.0
0.5
1.0



Parameters characterizing columns:

Parameters characterizing columns:

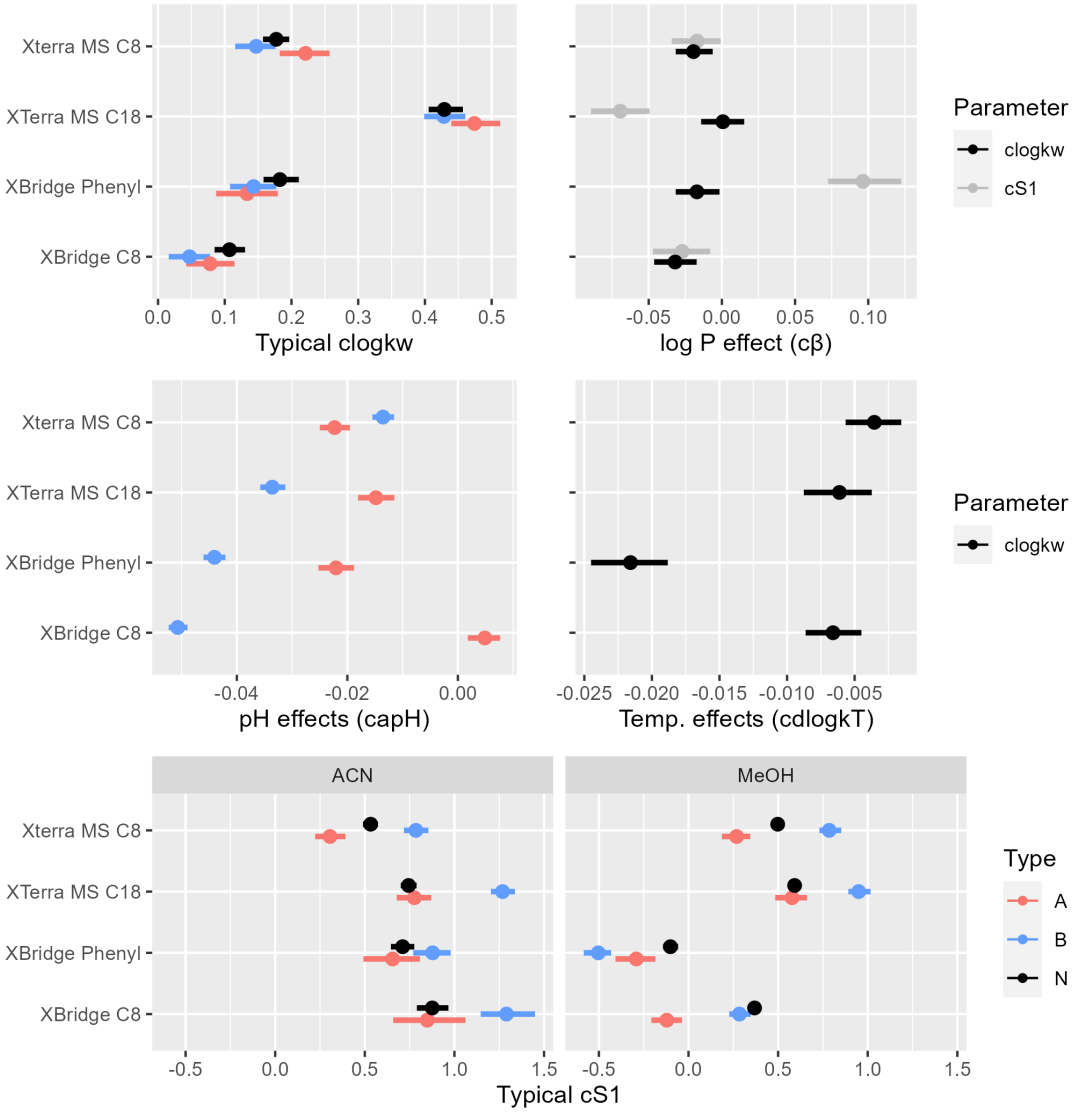
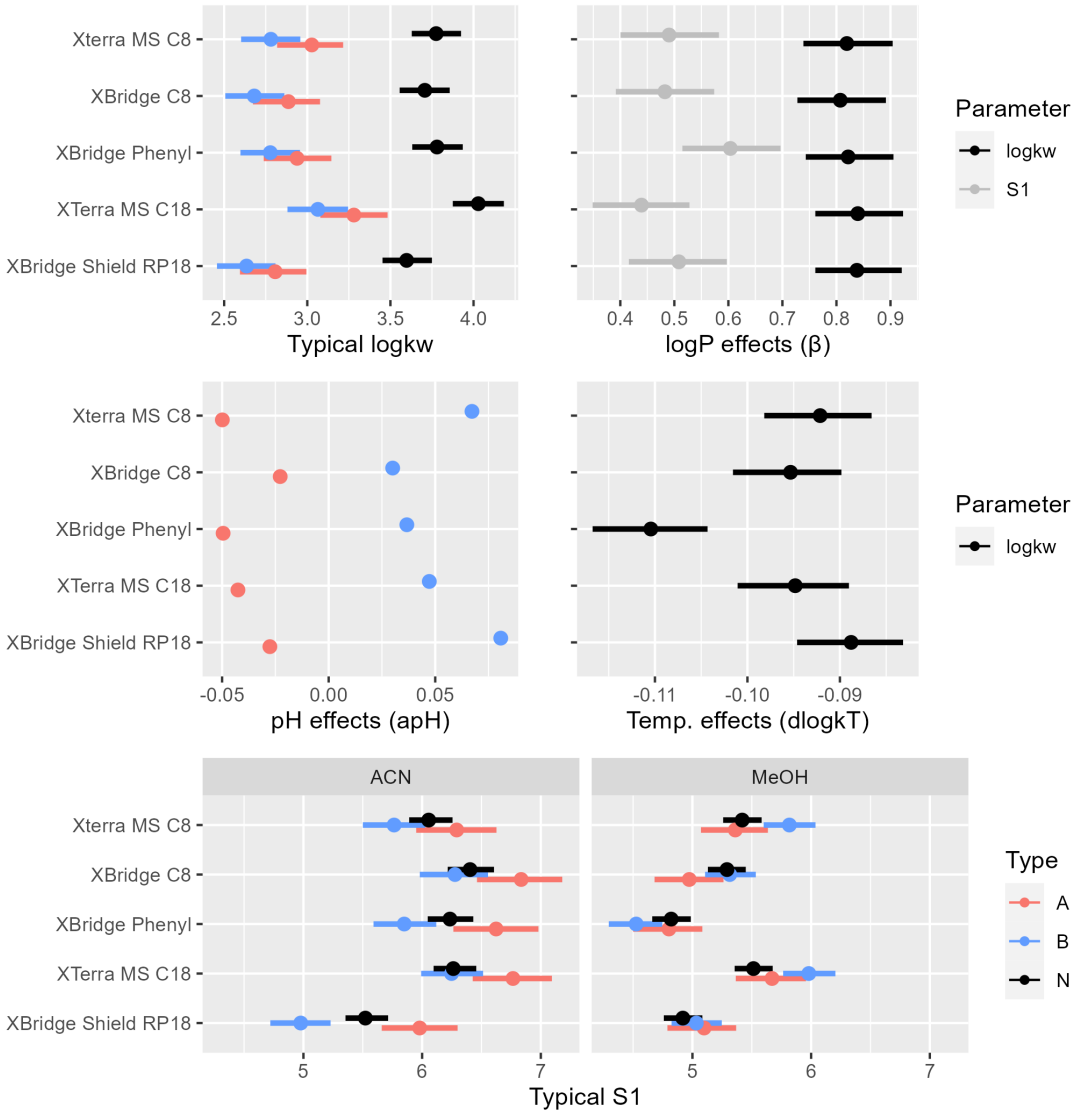
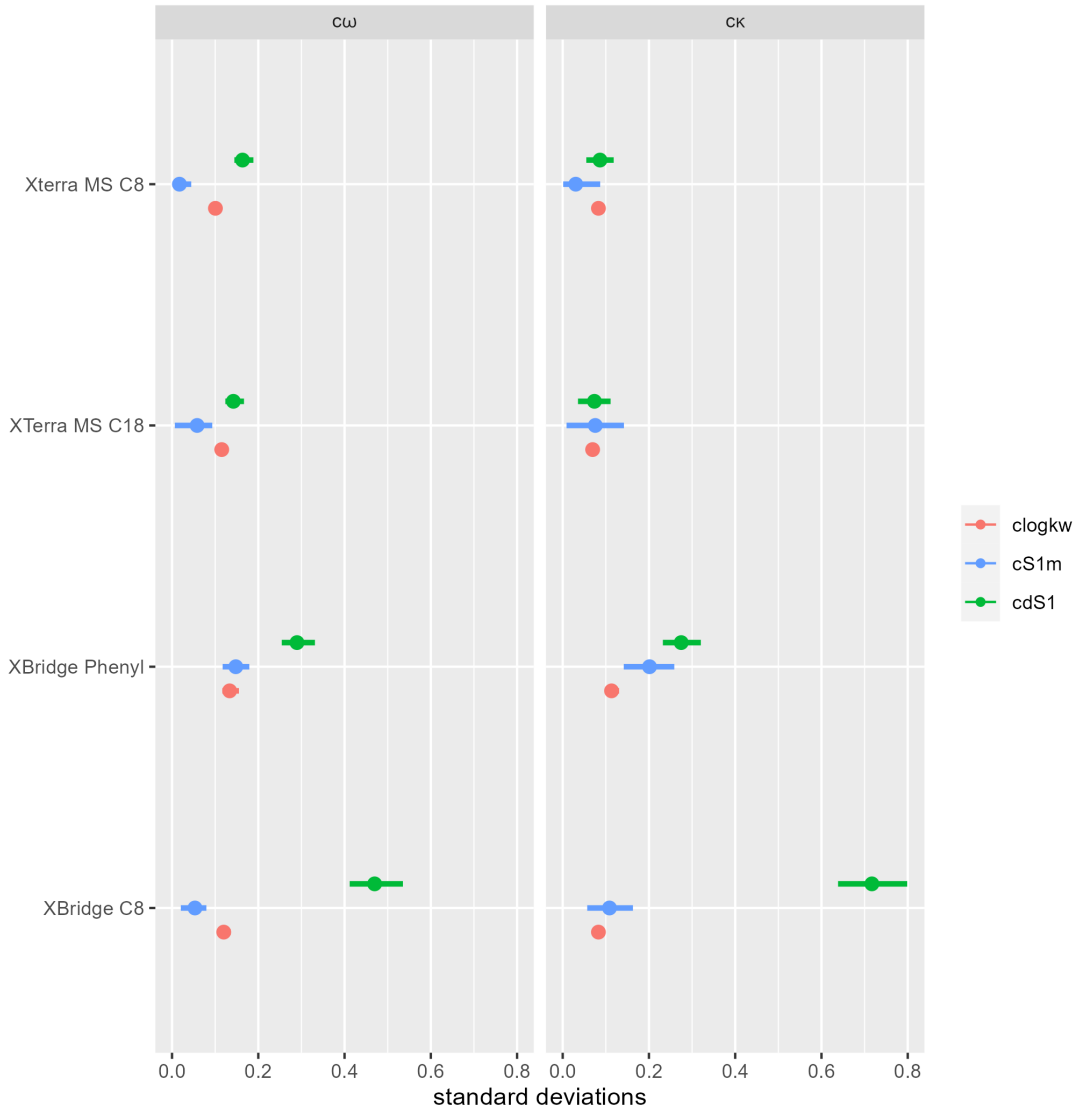
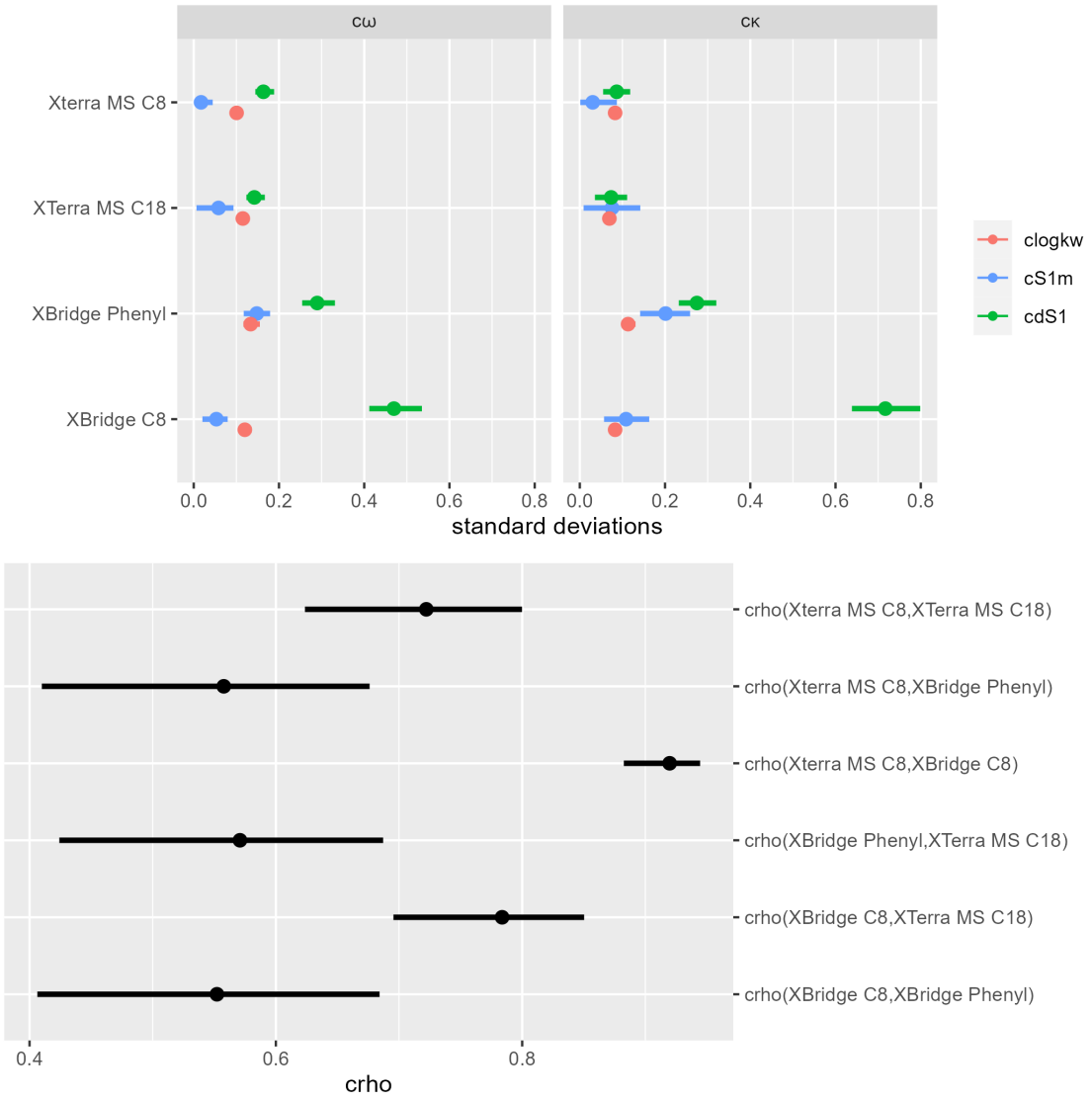


Figure 3: **?(caption)**



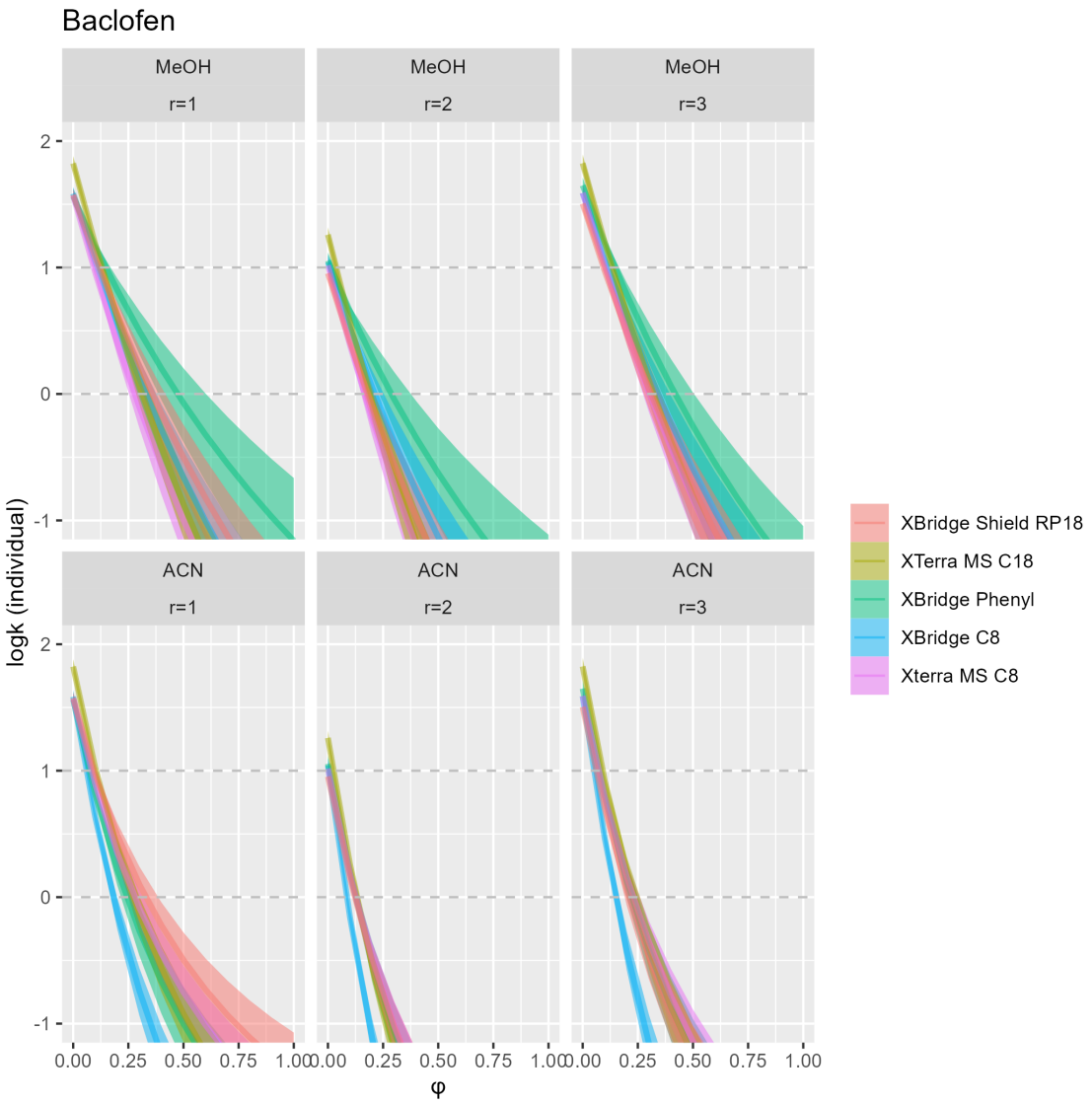
Standard deviations:

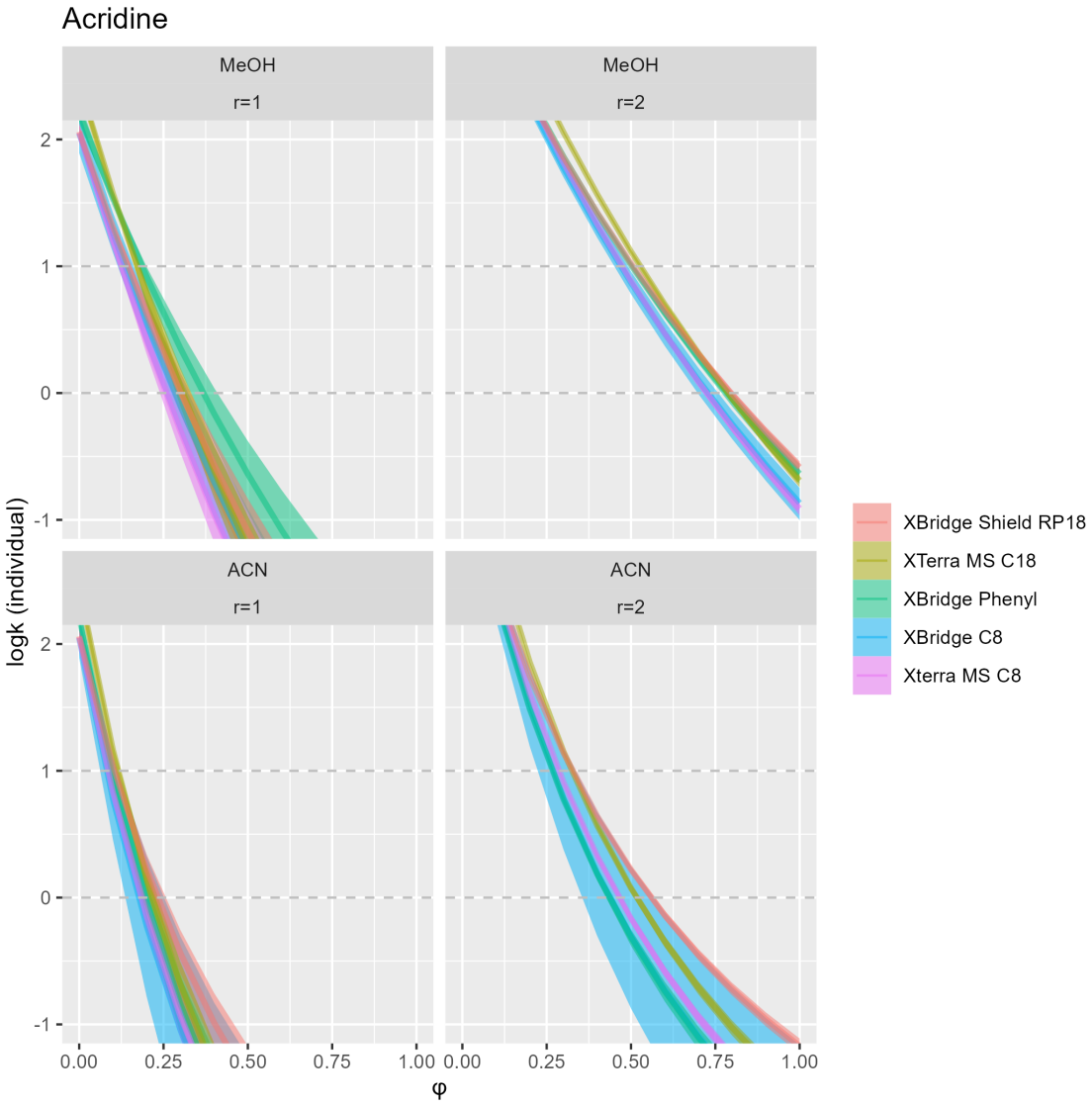

cω
cκ
Xterra MS C8
XTerra MS C18
XBridge Phenyl
XBridge C8
0.0
0.2
0.4
0.6
0.8
standard deviations
clogkw
cS1m
cdS1

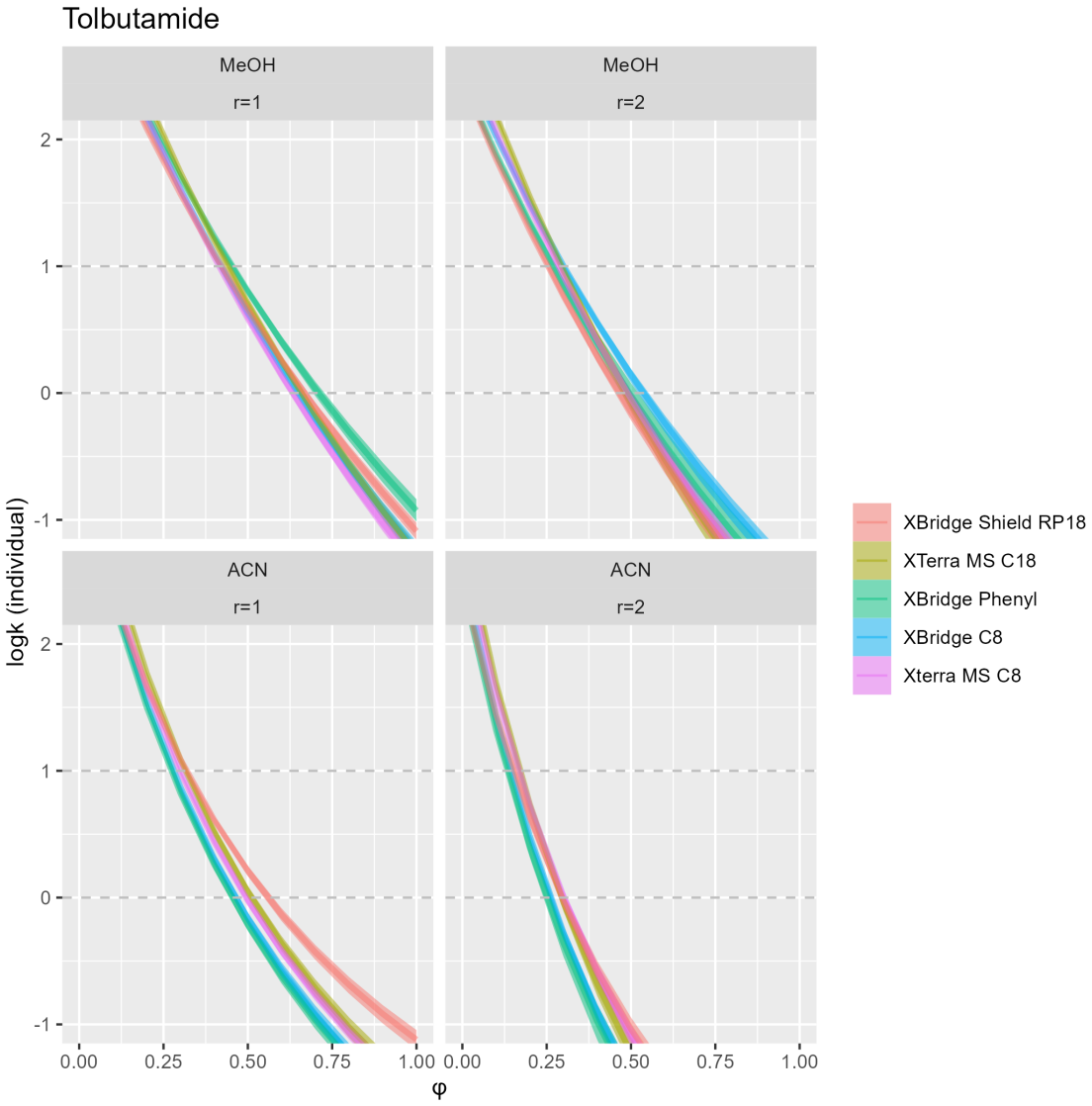


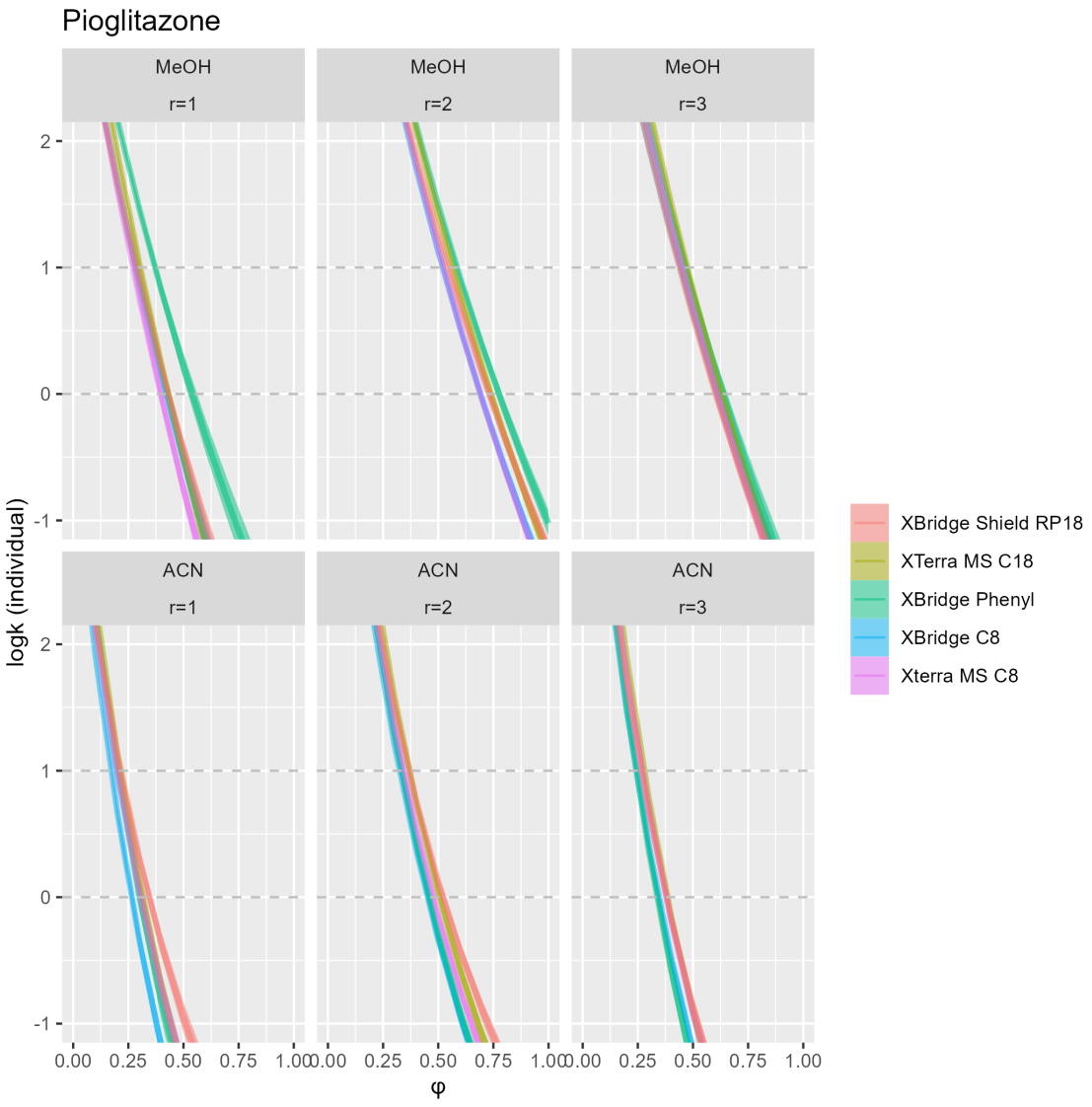
6.4 Isocratic predictions

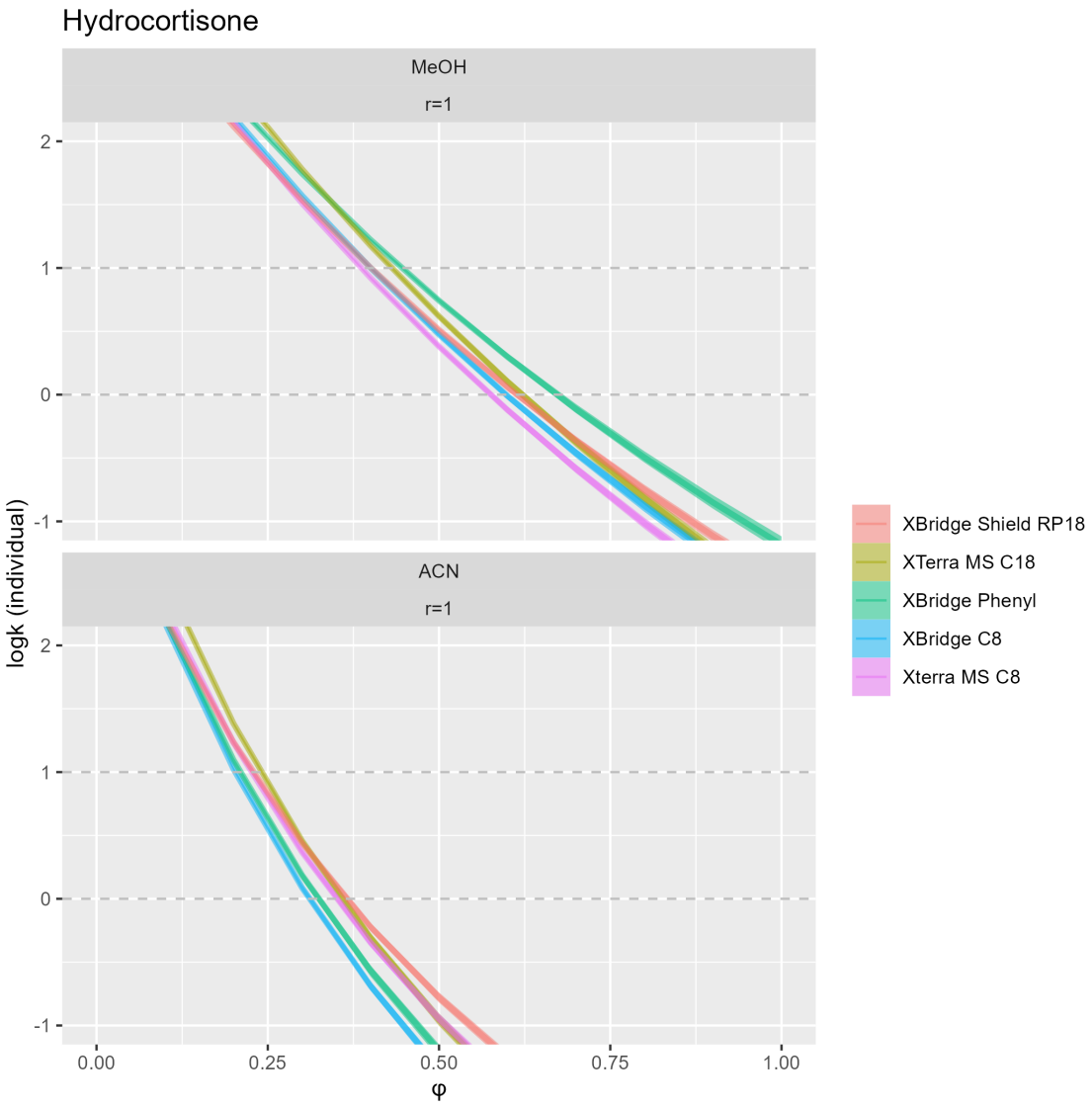
To better asses the impact of parameters on retention we created graphs presenting the isocratic logarithm of retention factor vs. φ for selected analytes. Separate graphs are shown for each dissociation form (r=1,r=2,r=3). Here the individual predictions (given all the data) are shown:

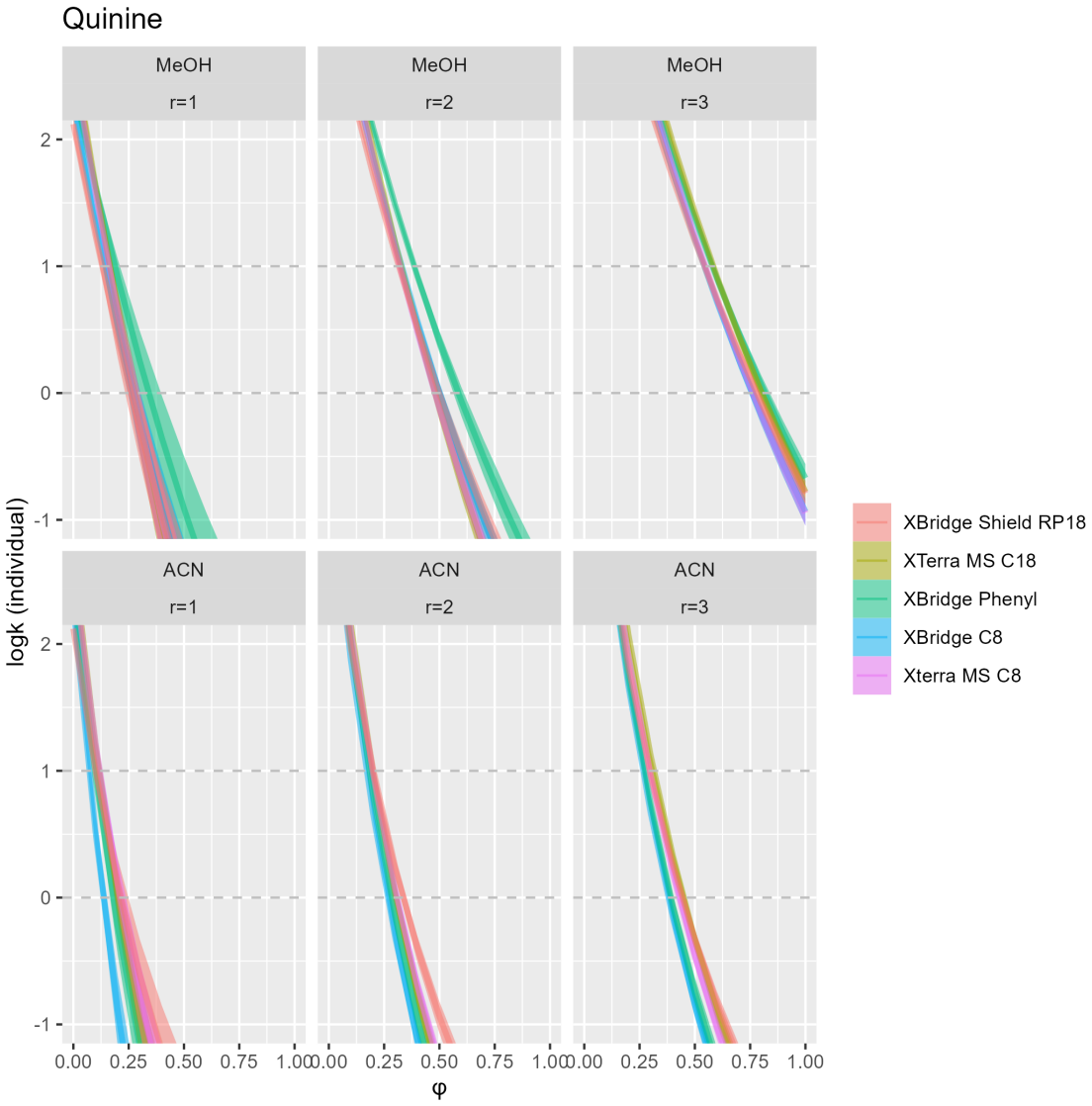

Baclofen
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
logk (individual)
2
1
0
-1
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Acridine
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
logk (individual)
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

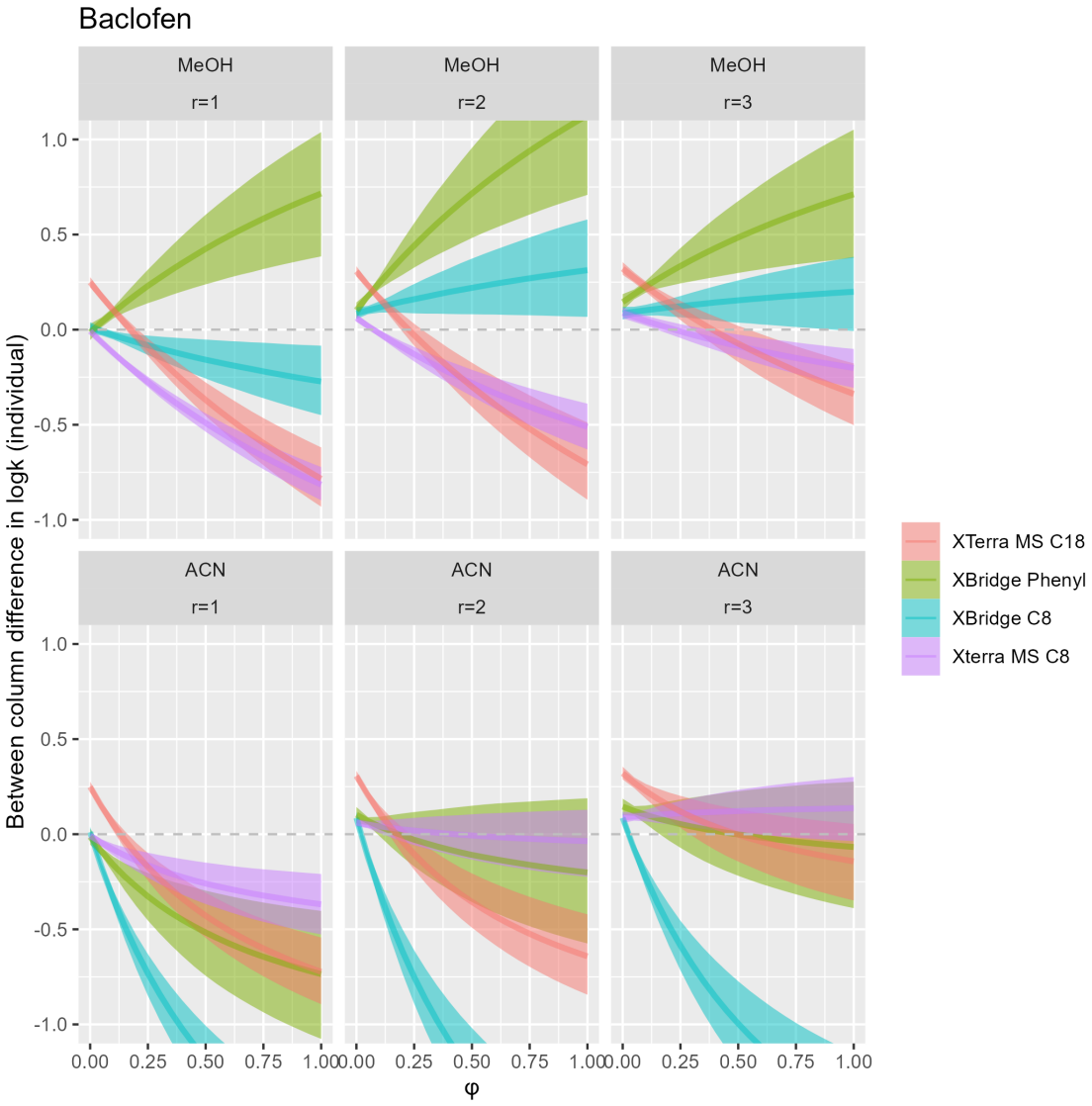

Tolbutamide
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
logk (individual)
φ
2
1
0
-1
0.00
0.25
0.50
0.75
1.00
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

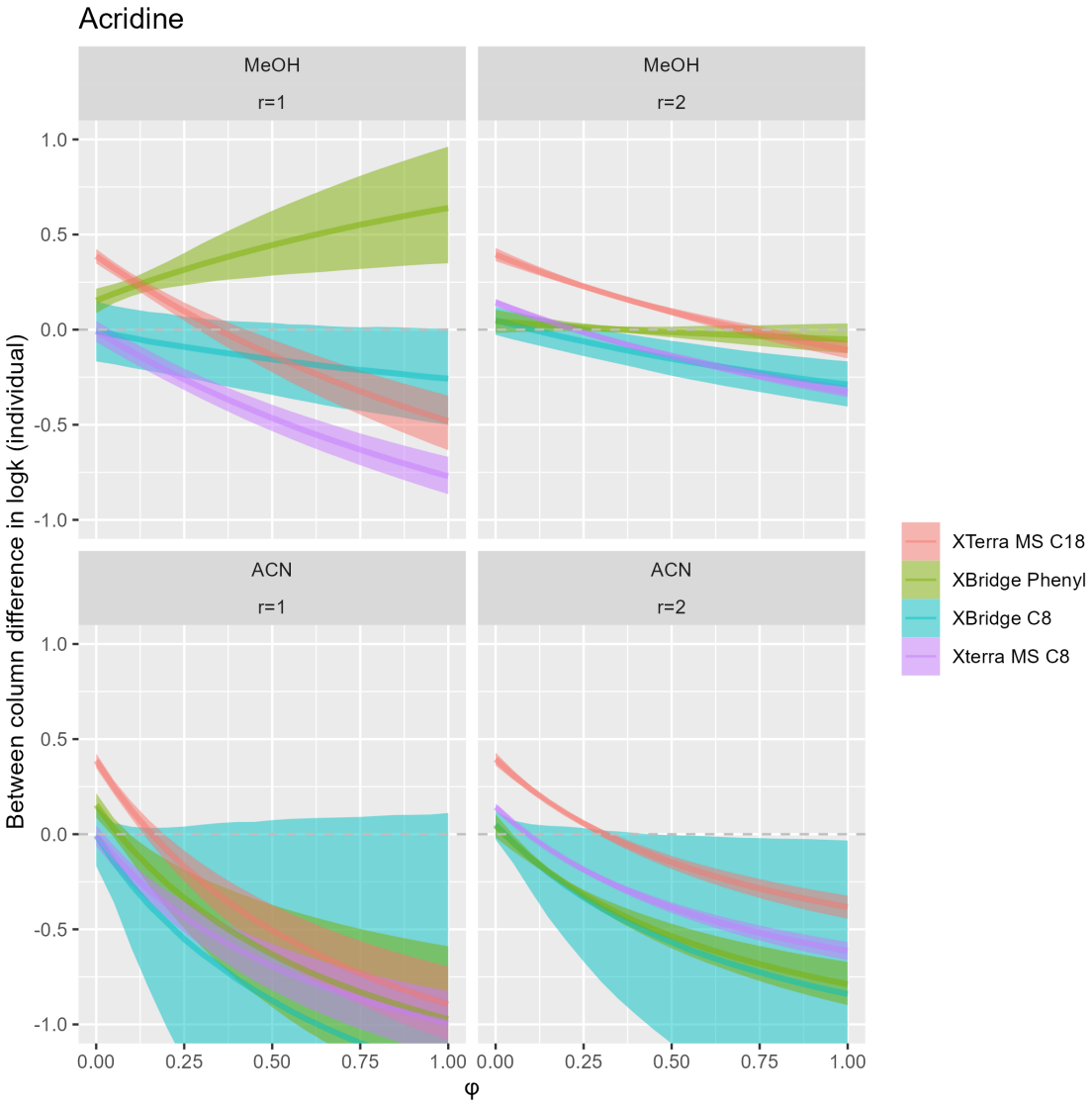

Pioglitazone
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
logk (individual)
2
1
0
-1
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

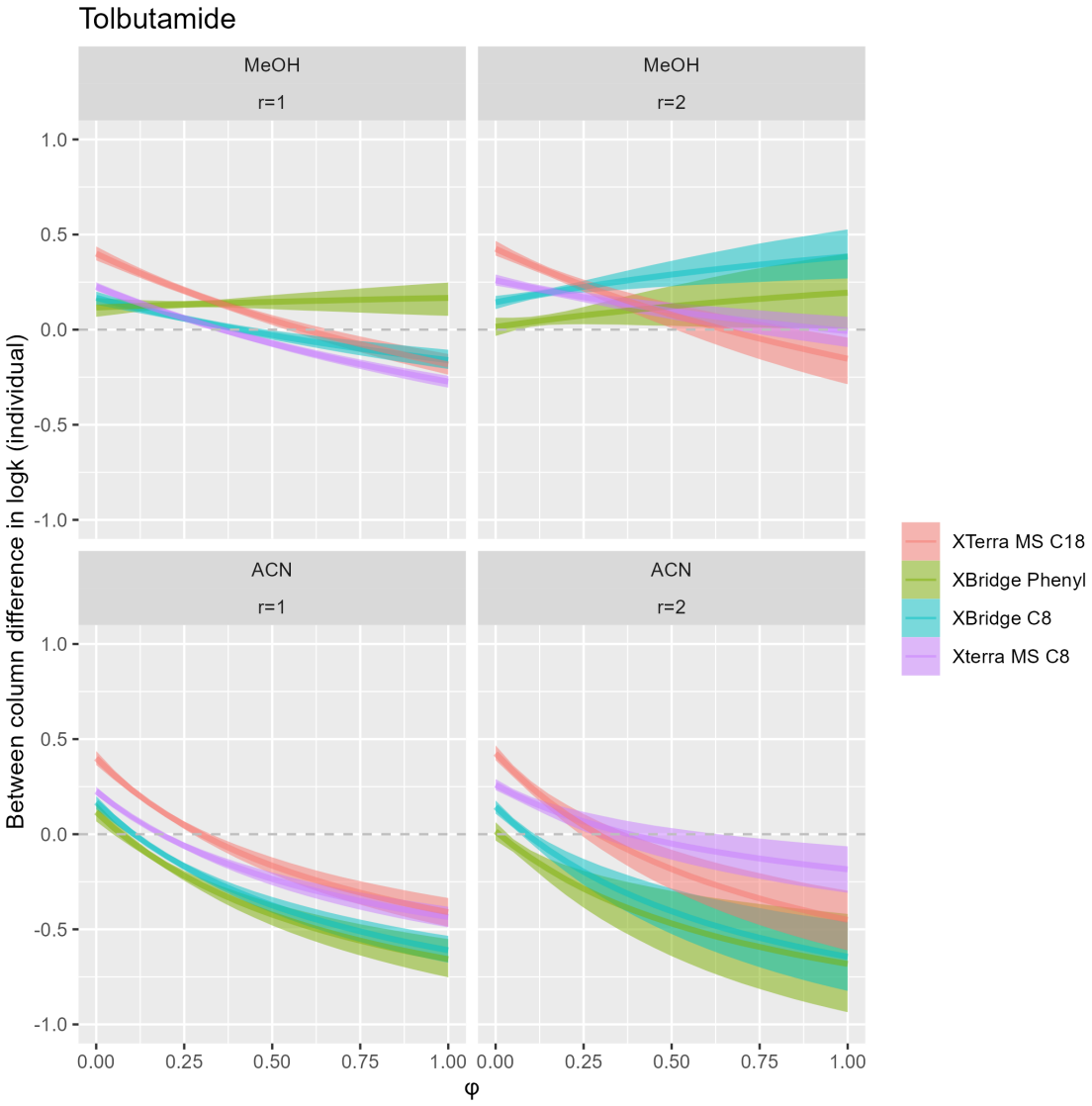

Hydrocortisone
MeOH
r=1
ACN
r=1
logk (individual)
2
1
0
-1
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

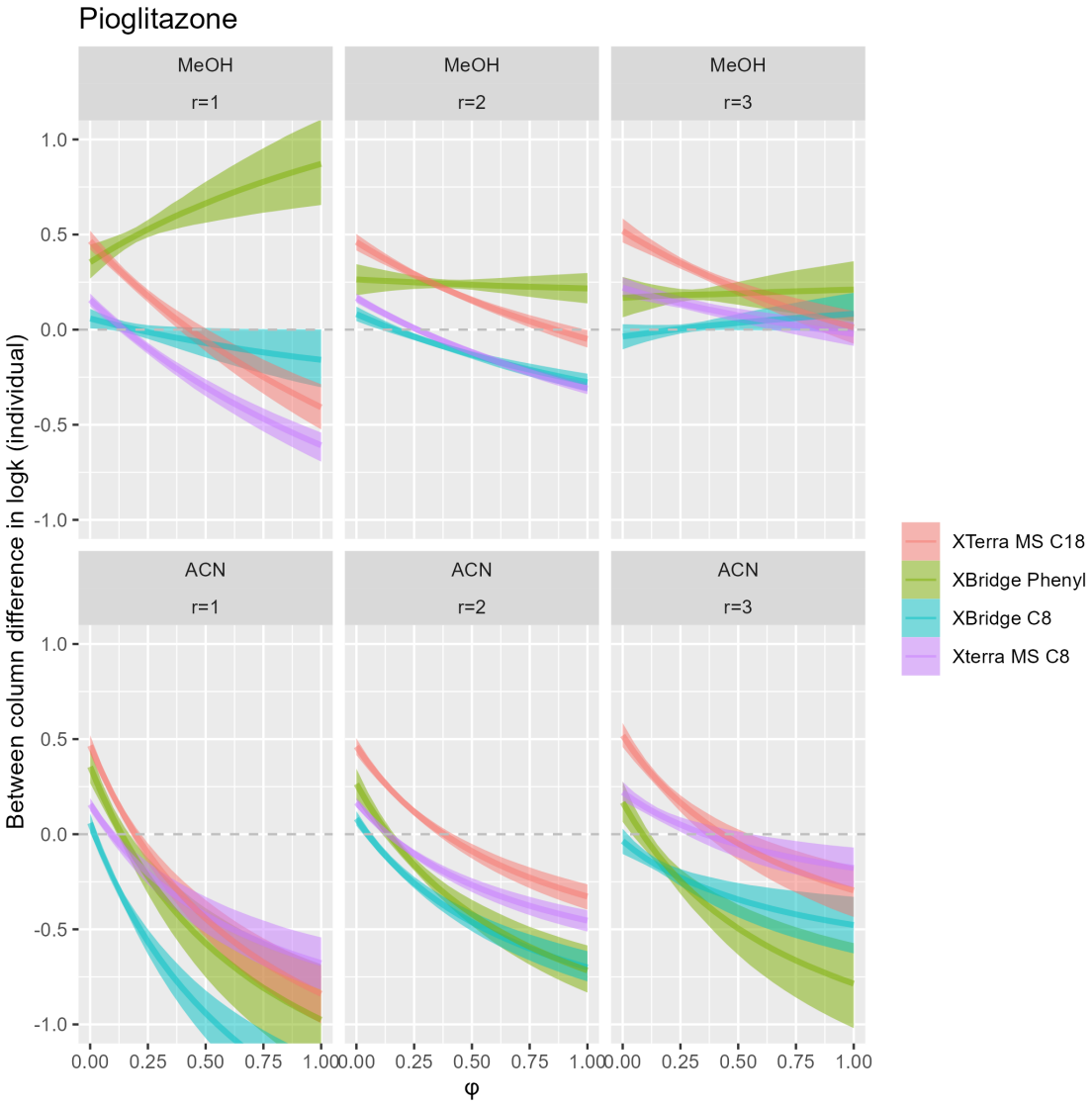


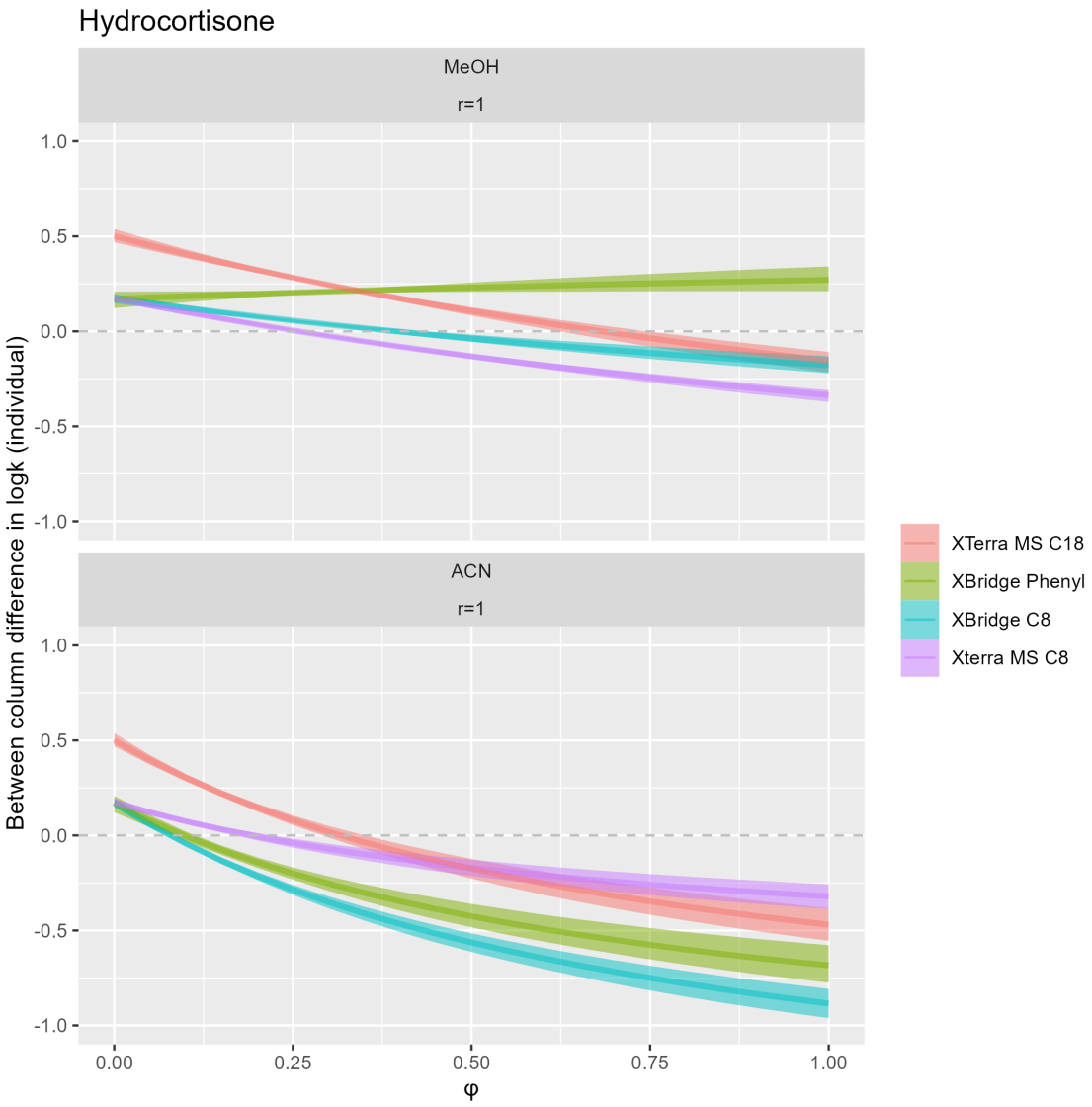
Similarly we can quantify the column effects (between column differences in logk using XBridge Shield RP18 as a reference column):

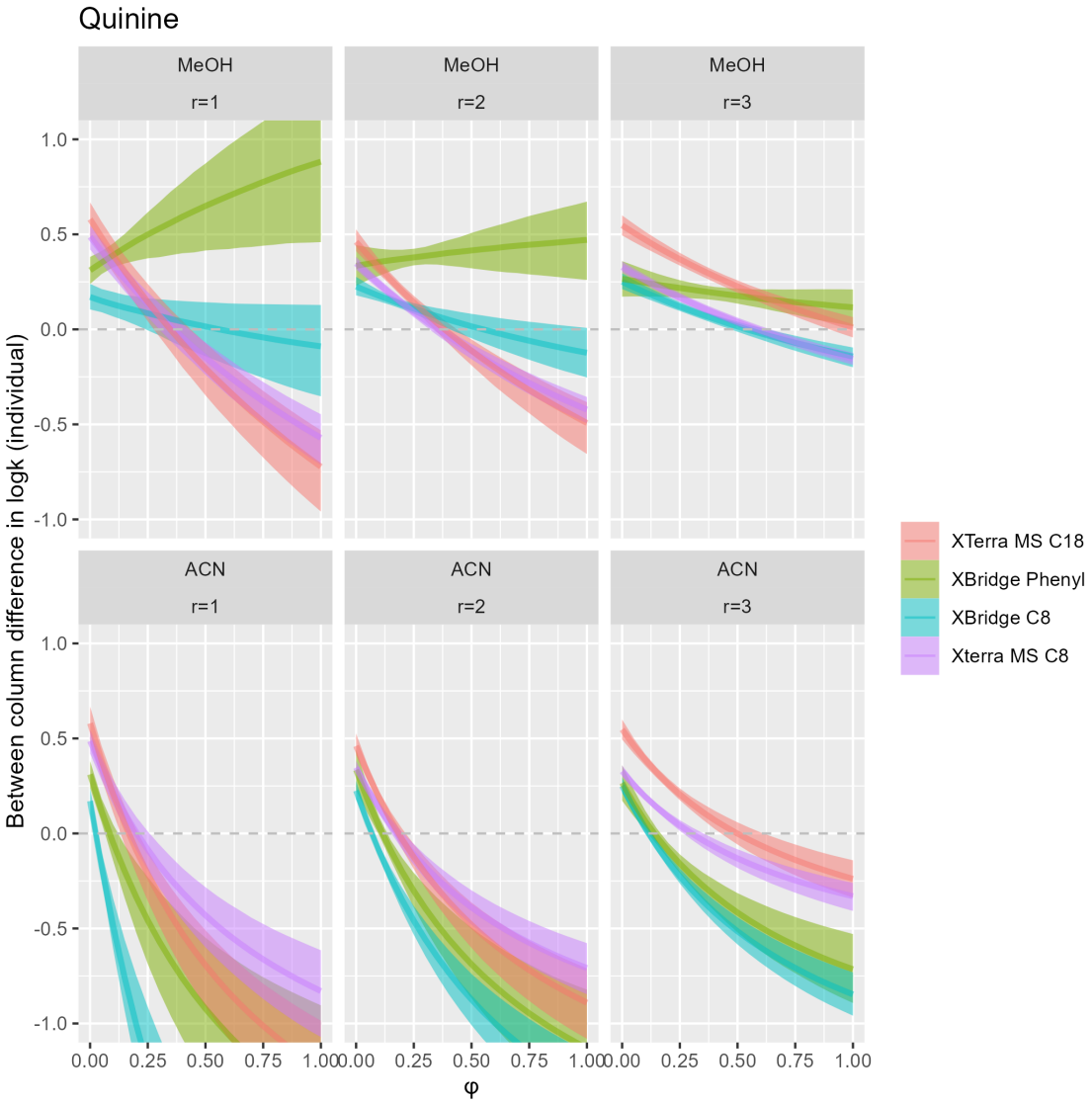

Baclofen
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
Between column difference in logk (individual)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Acridine
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
Between column difference in logk (individual)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

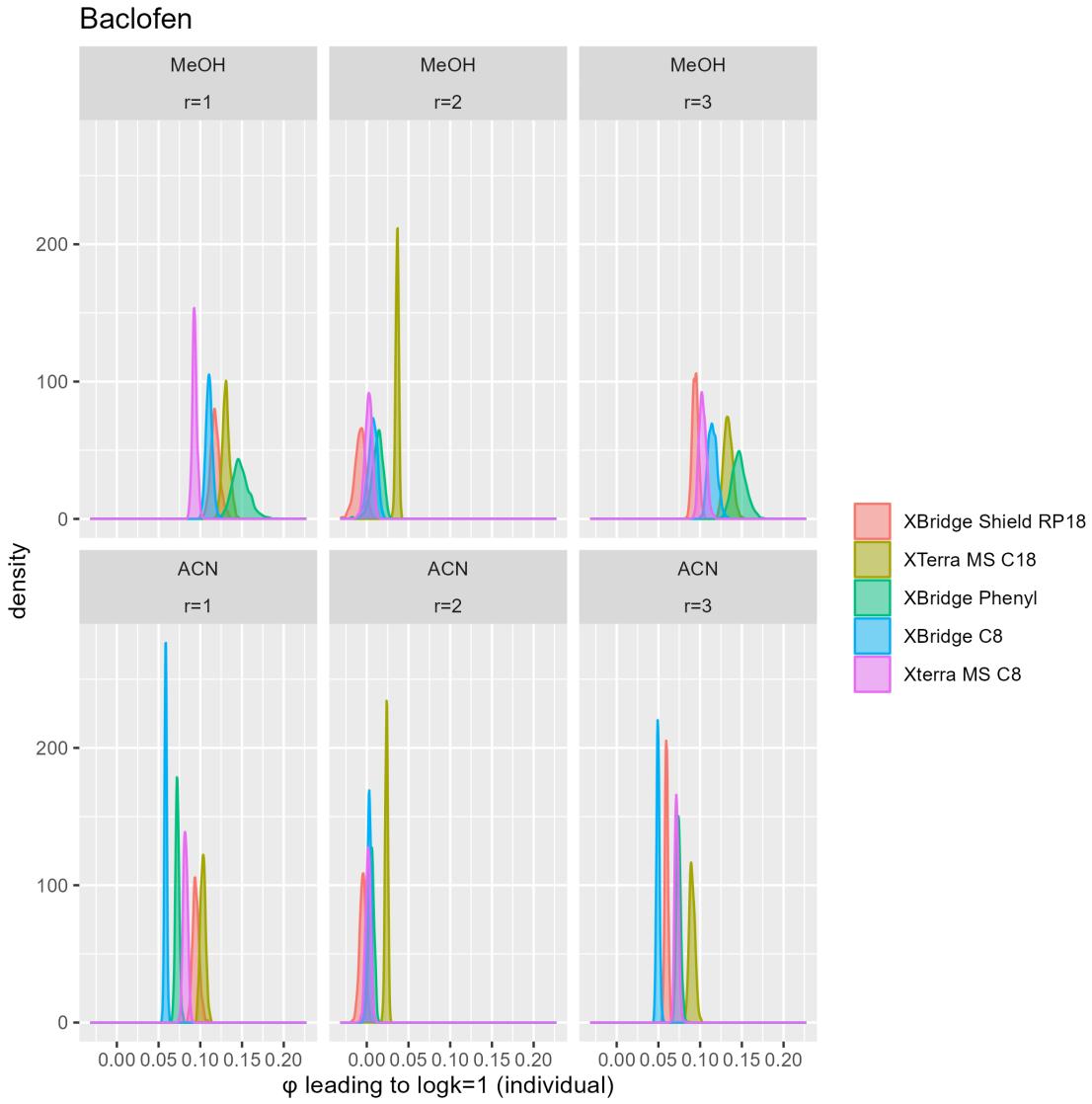

Tolbutamide
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
Between column difference in logk (individual)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

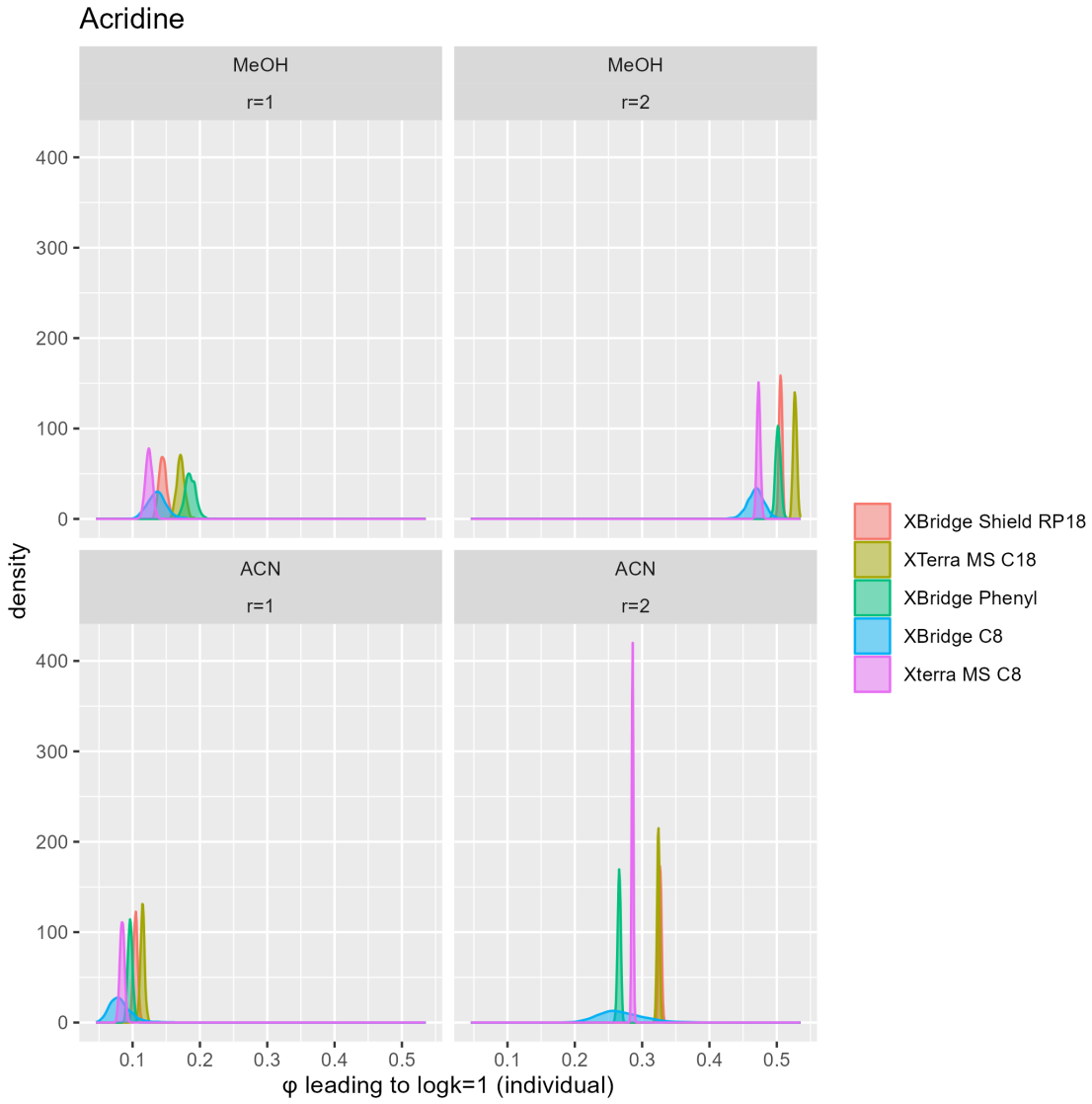

Pioglitazone
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
Between column difference in logk (individual)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

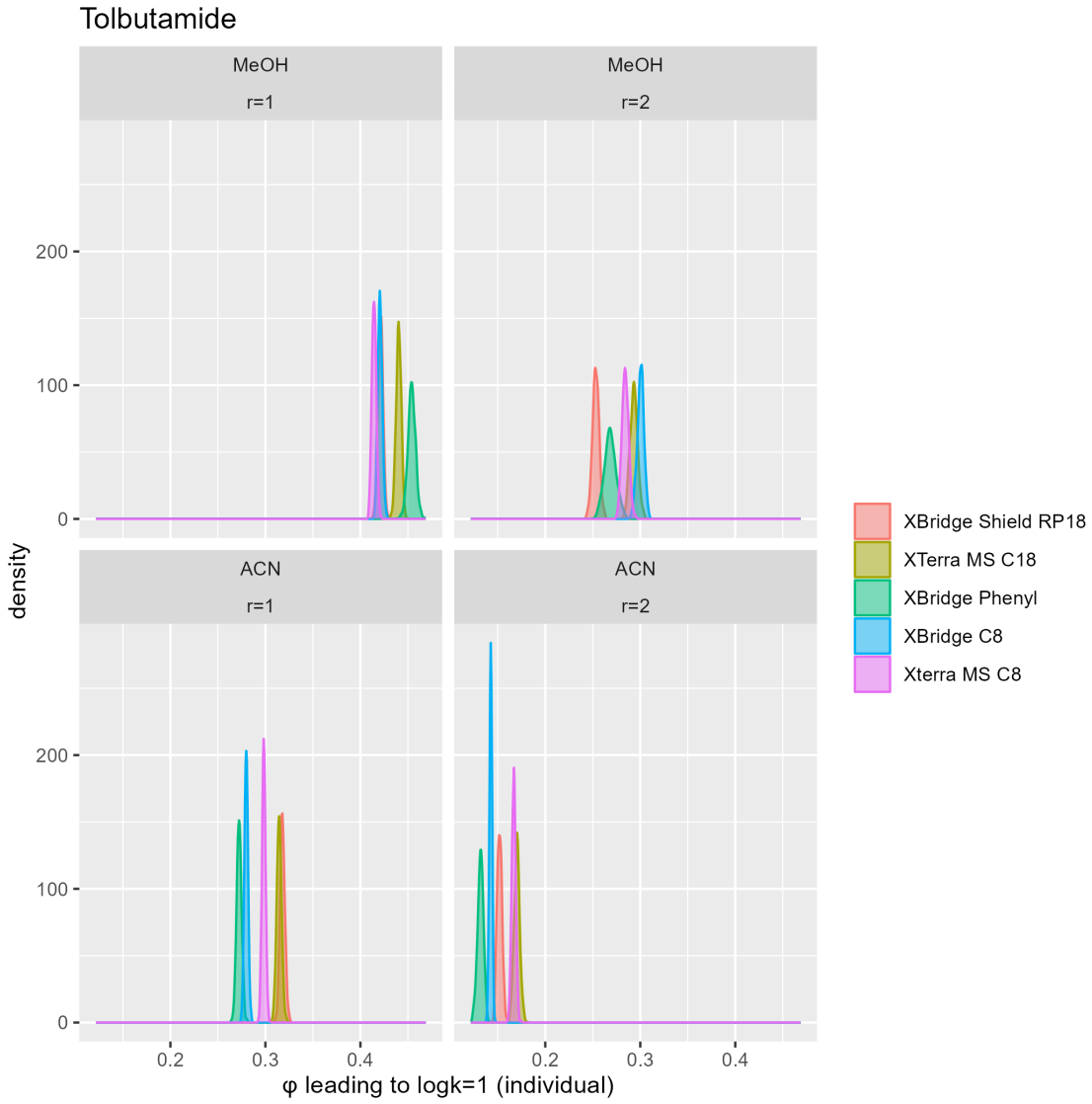

Hydrocortisone
MeOH
r=1
ACN
r=1
Between column difference in logk (individual)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

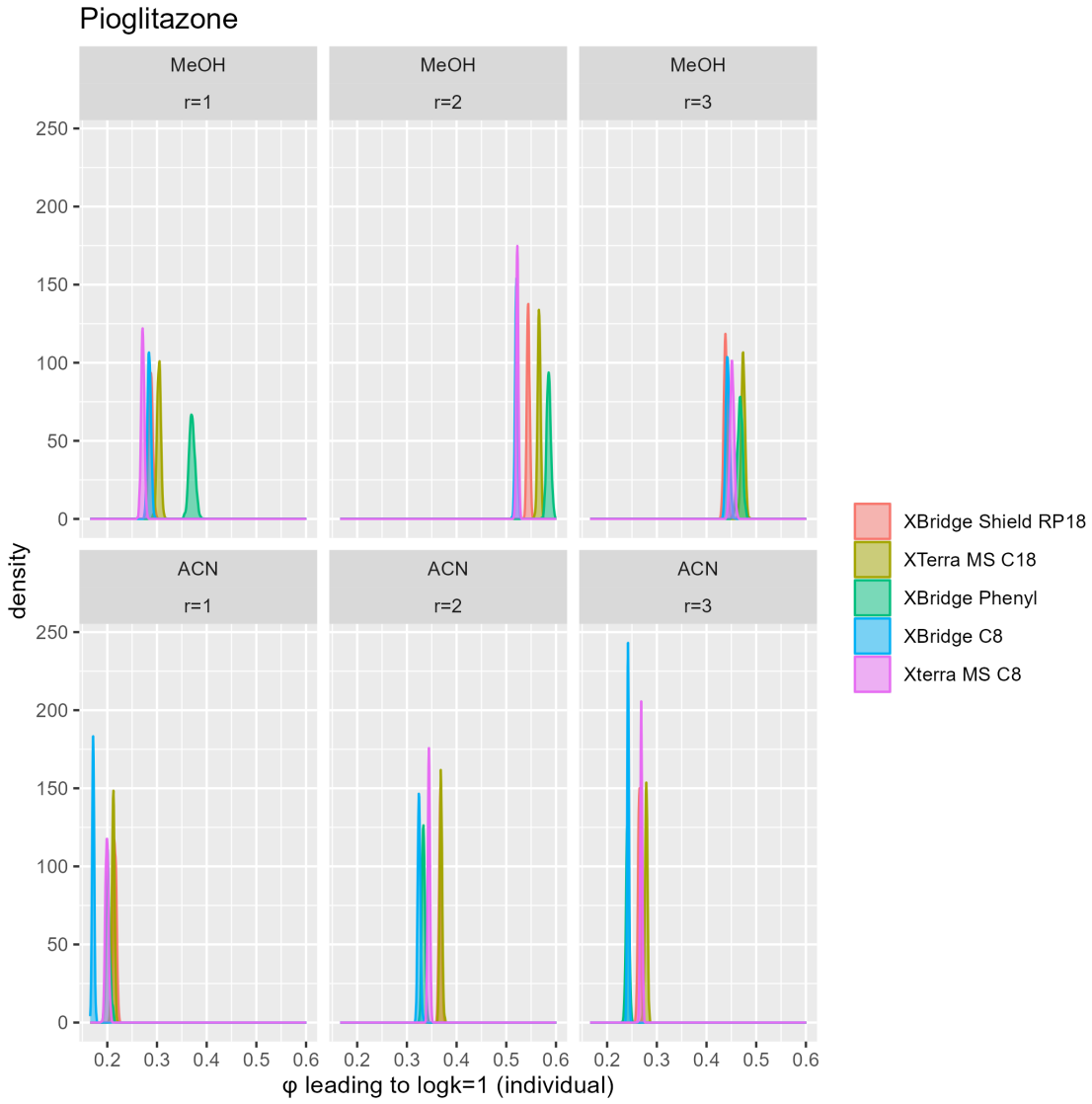


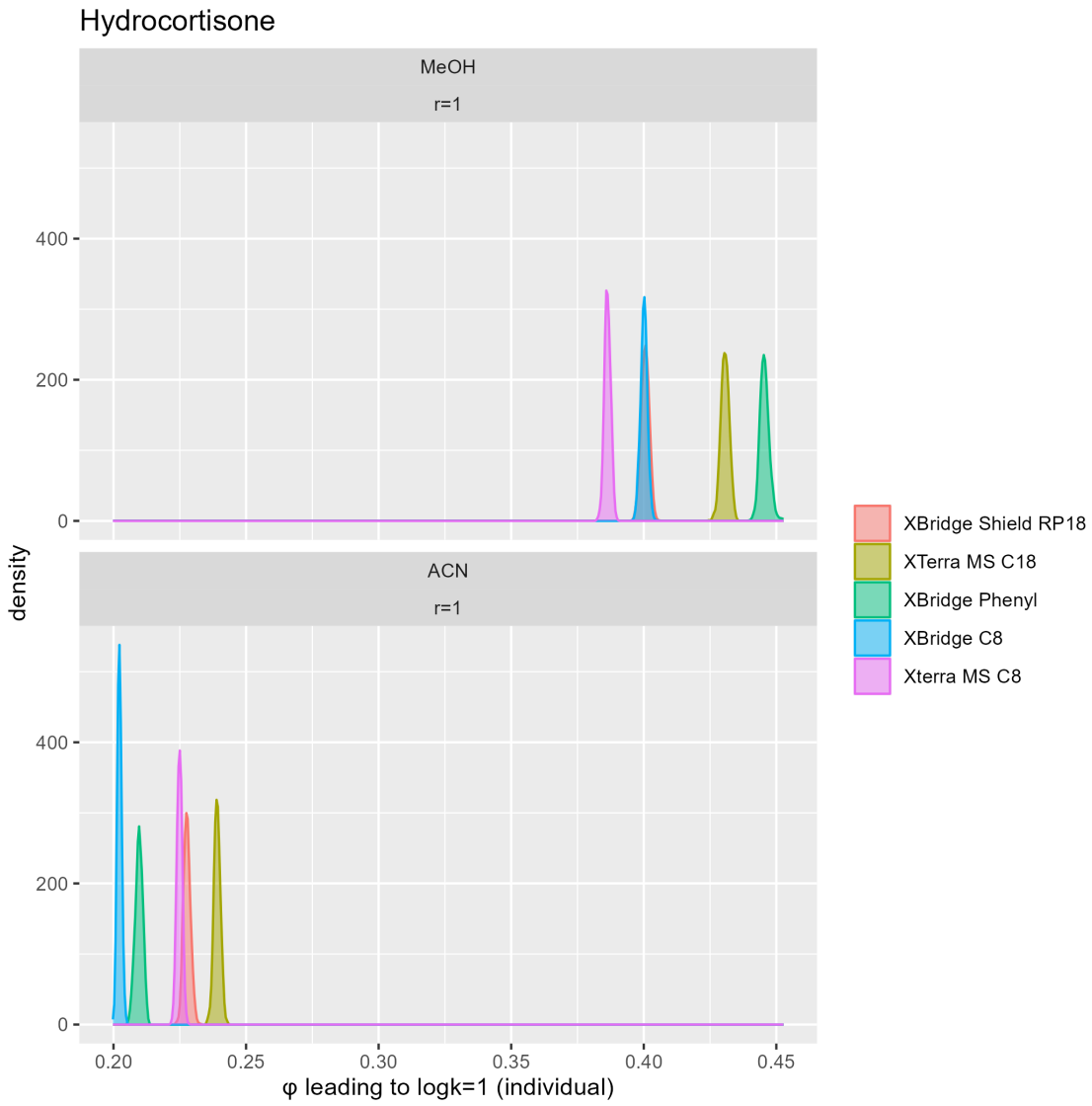
or predict the organic modifier content leading to logk of 1:

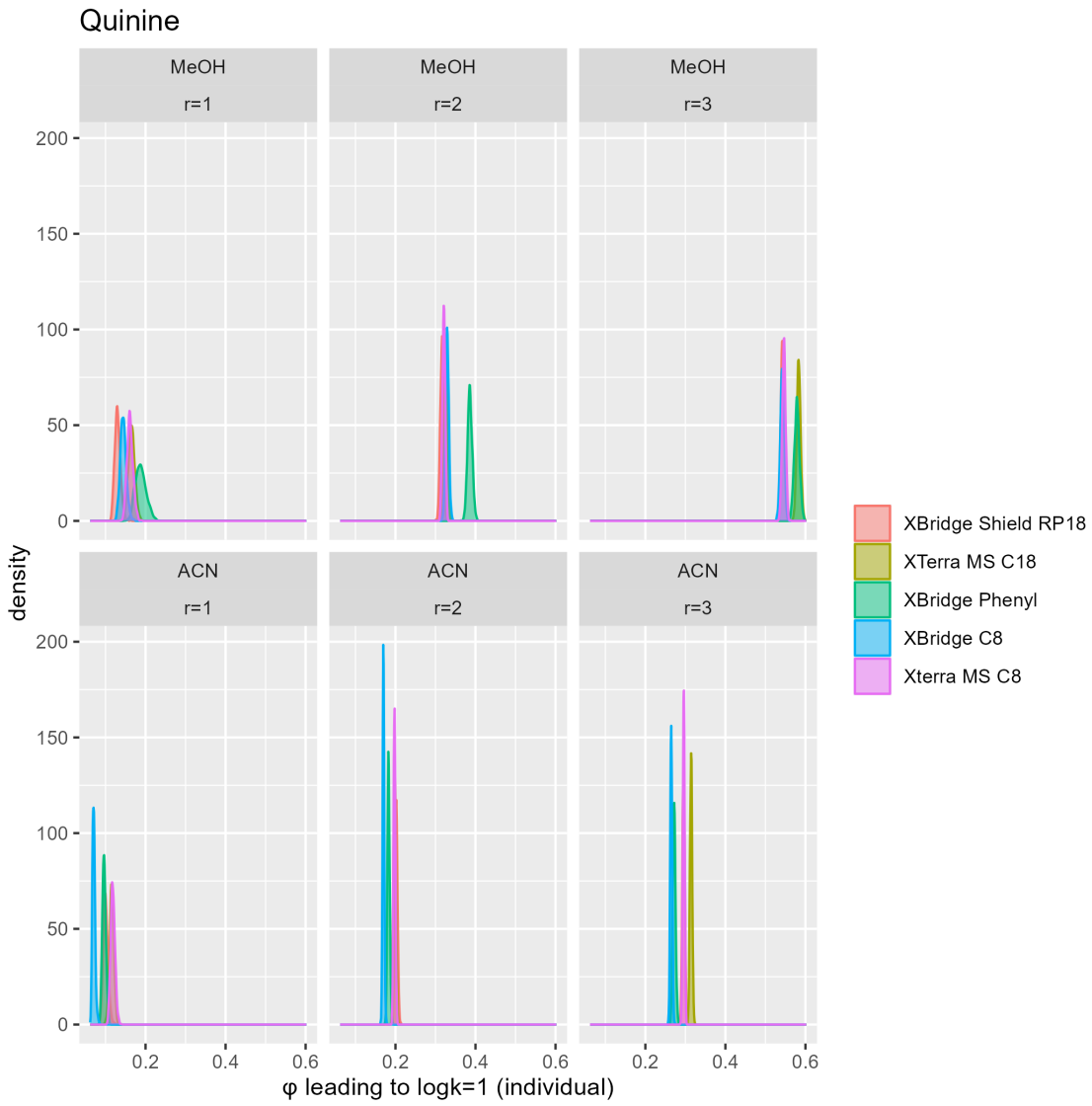

Baclofen
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
density
200
100
0
0.00 0.05 0.10 0.15 0.20
φ leading to logk=1 (individual)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Acridine
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
density
400
300
200
100
0
0.1
0.2
0.3
0.4
0.5
φ leading to logk=1 (individual)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Tolbutamide
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
density
200
100
0
0.2
0.3
0.4
φ leading to logk=1 (individual)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Pioglitazone
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
density
250
200
150
100
50
0
0.2
0.3
0.4
0.5
0.6
φ leading to logk=1 (individual)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Hydrocortisone
MeOH
r=1
ACN
r=1
density
400
200
0
0.20
0.25
0.30
0.35
0.40
0.45
φ leading to logk=1 (individual)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

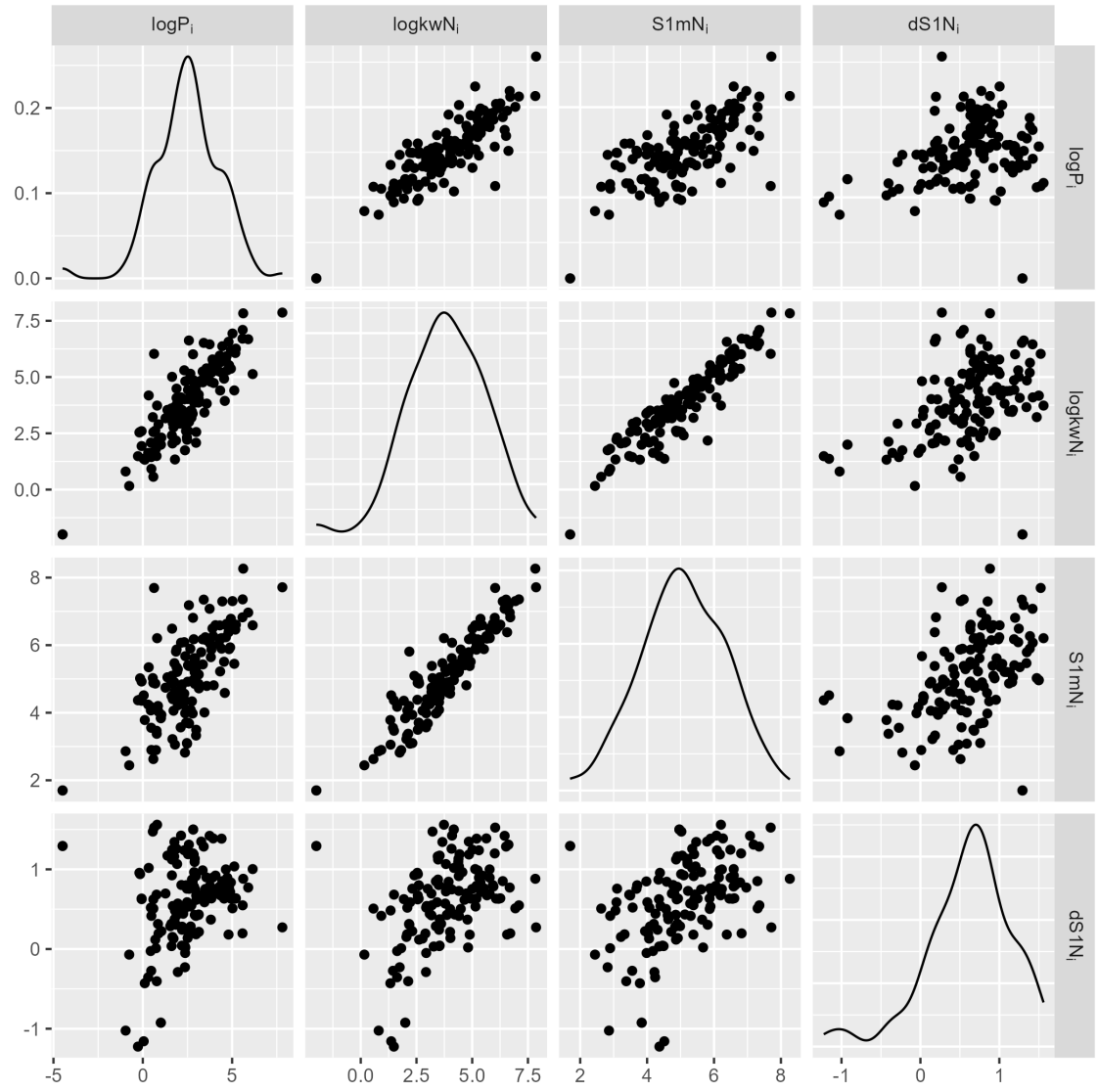


6.5 Individual Parameters

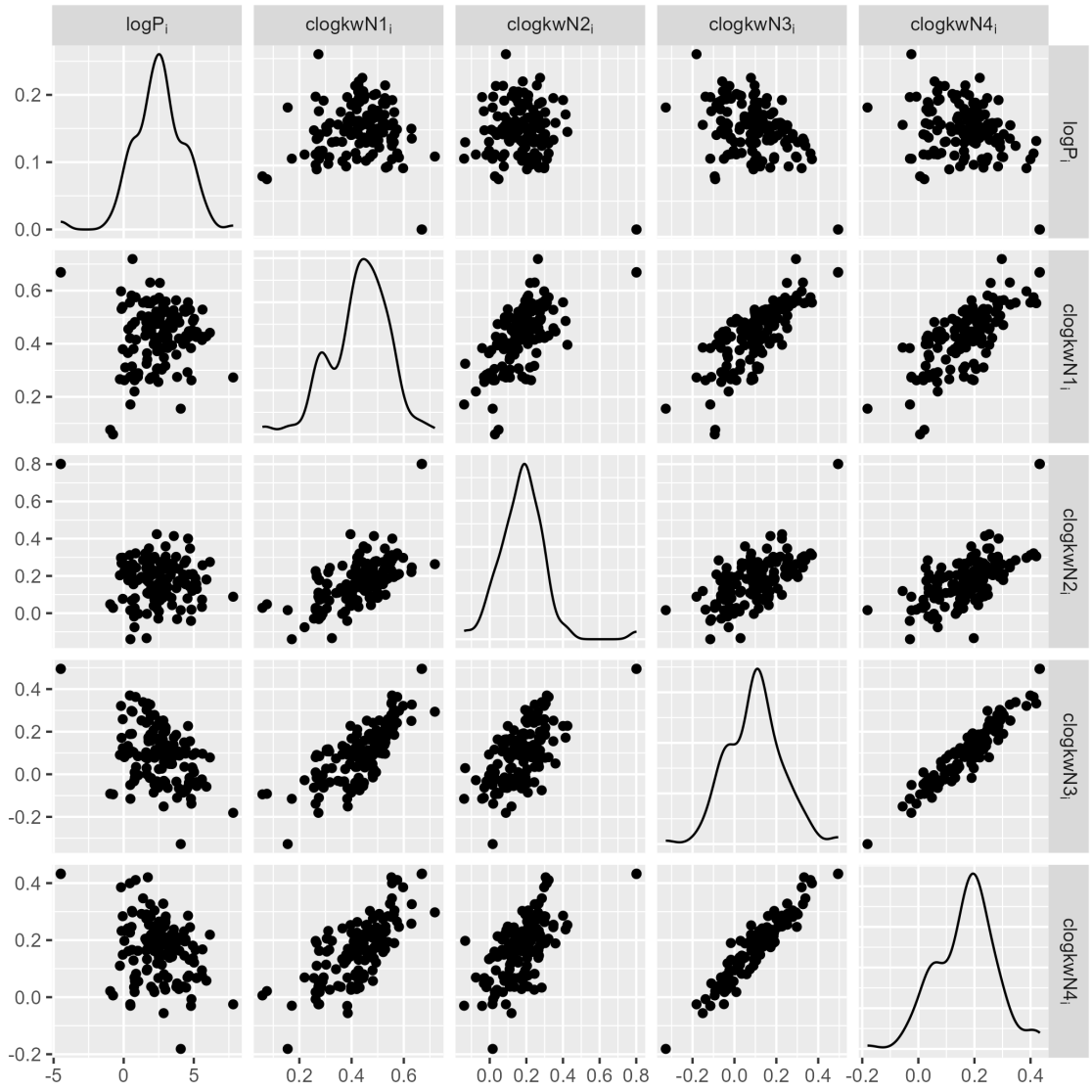
Individual parameter are the analyte-specific parameters estimated by the model. The following plots allow to assess the correlations between these parameters.

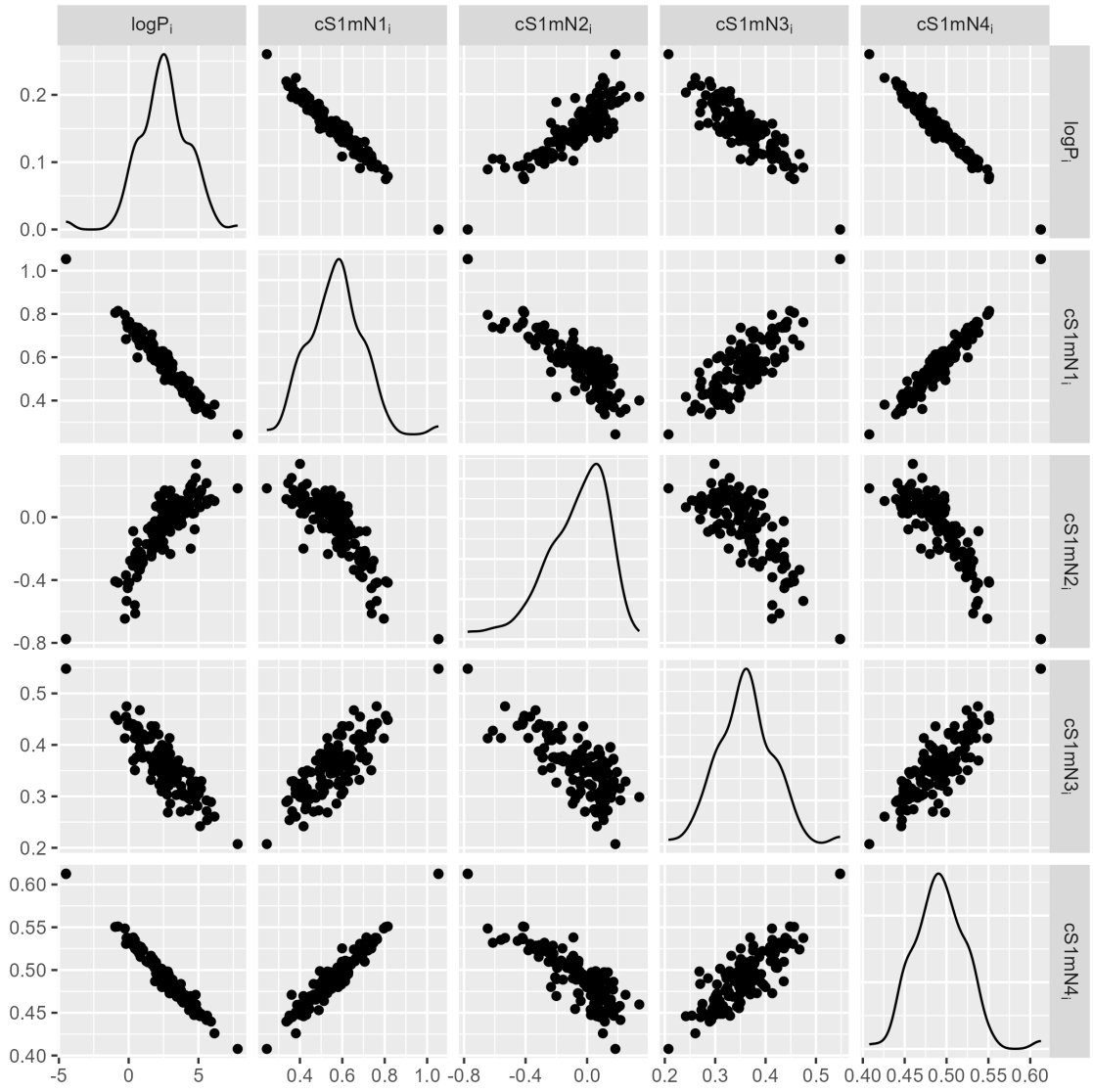
6.5.1 Neutral Form

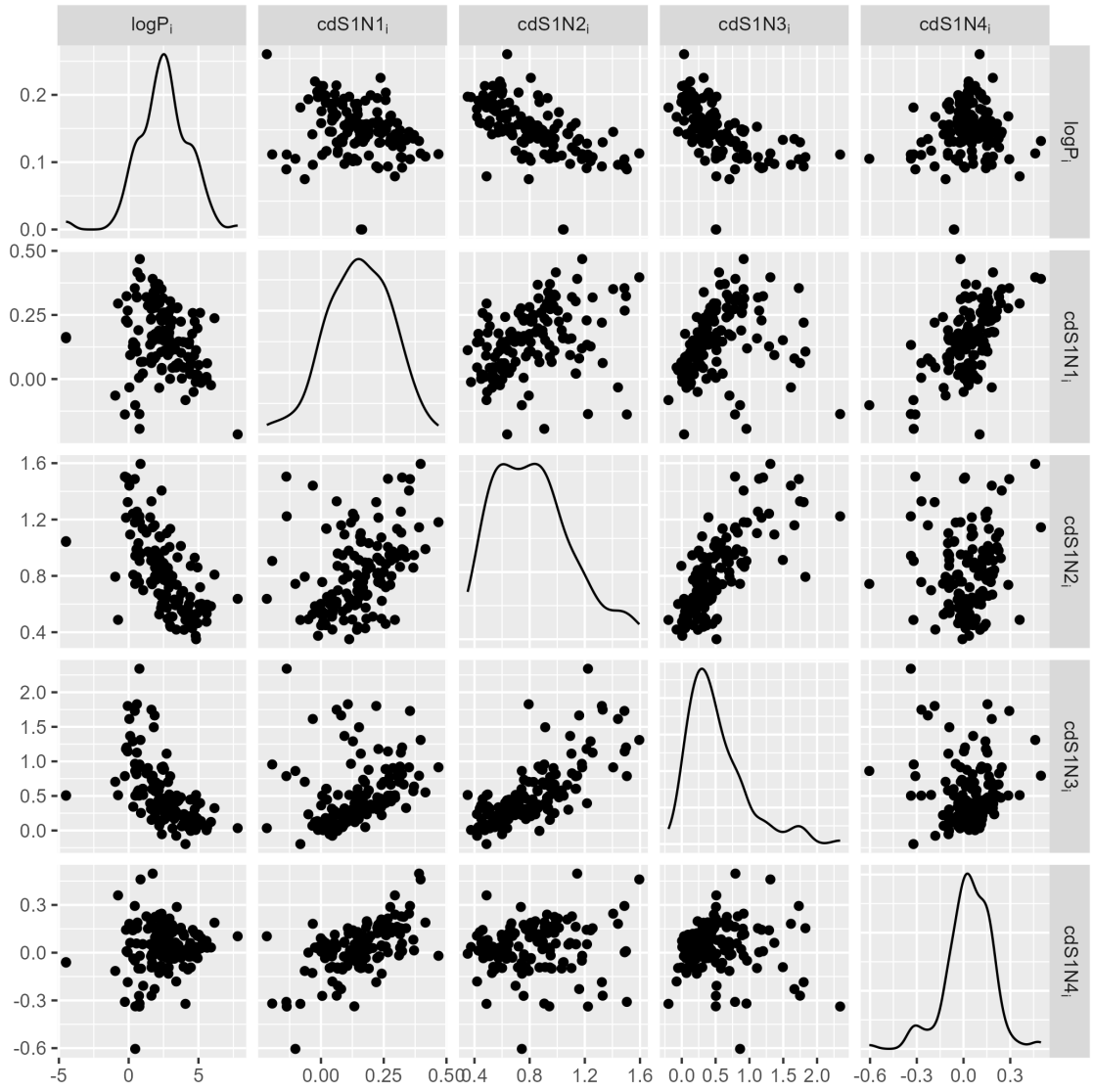
Individual parameters for the reference column (XBridge Shield RP18):



Individual parameters for column effects:

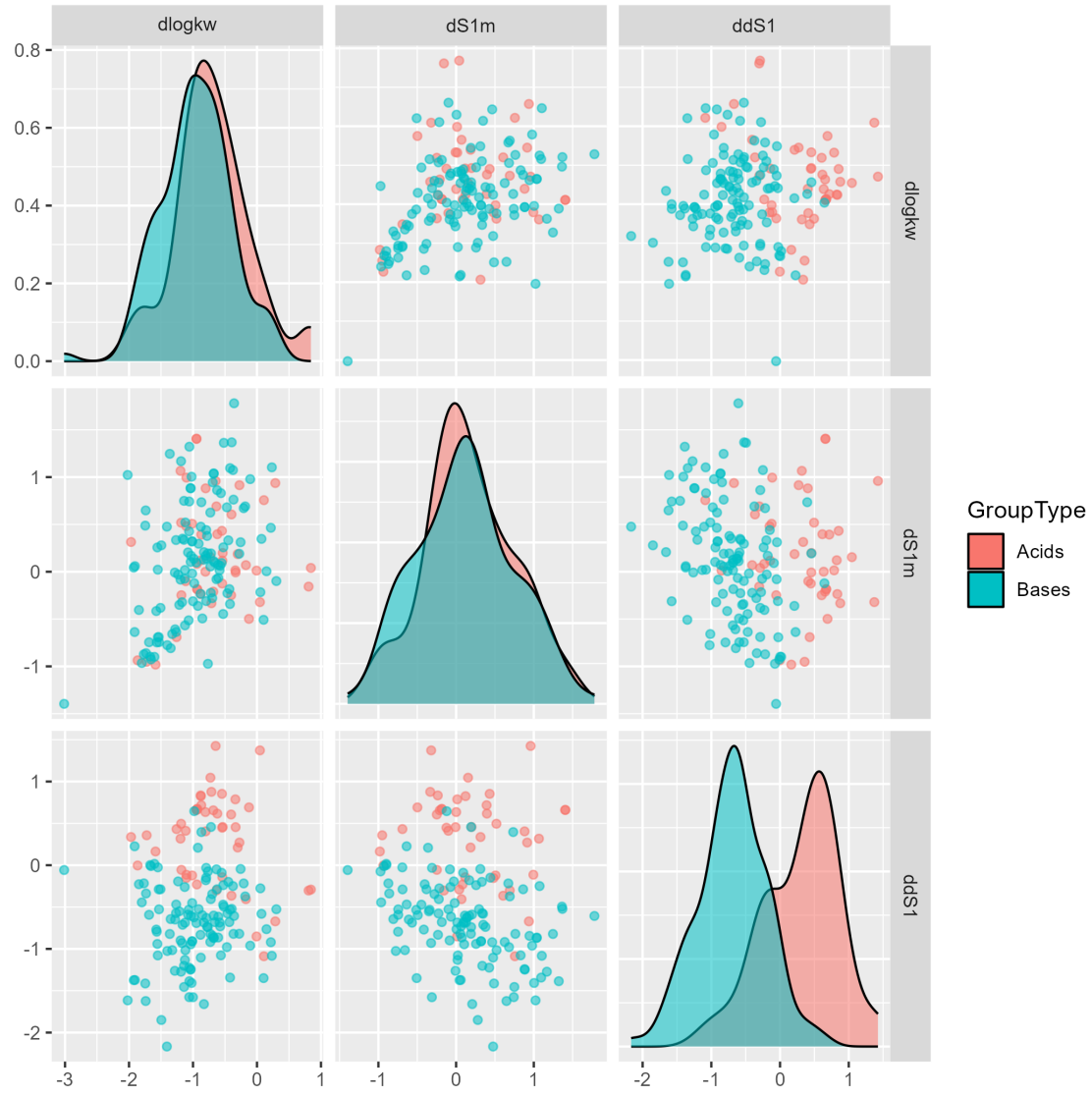

logP$_i$
clogkwN1$_i$
clogkwN2$_i$
clogkwN3$_i$
clogkwN4$_i$
0.2
0.1
0.0
0.6
0.4
0.2
0.8
0.6
0.4
0.2
0.0
0.4
0.2
0.0
-0.2
0.4
0.2
0.0
-0.2
-5
0
5
0.2
0.4
0.6
0.0
0.2
0.4
0.6
0.8
-0.2
0.0
0.2
0.4
-0.2
0.0
0.2
0.4


logP_i
cS1mN1_i
cS1mN2_i
cS1mN3_i
cS1mN4_i
0.2
0.1
0.0
1.0
0.8
0.6
0.4
0.0
-0.4
-0.8
0.5
0.4
0.3
0.2
0.60
0.55
0.50
0.45
0.40
-5
0
5
0.4
0.6
0.8
1.0
-0.8
-0.4
0.0
0.2
0.3
0.4
0.5
0.40
0.45
0.50
0.55
0.60

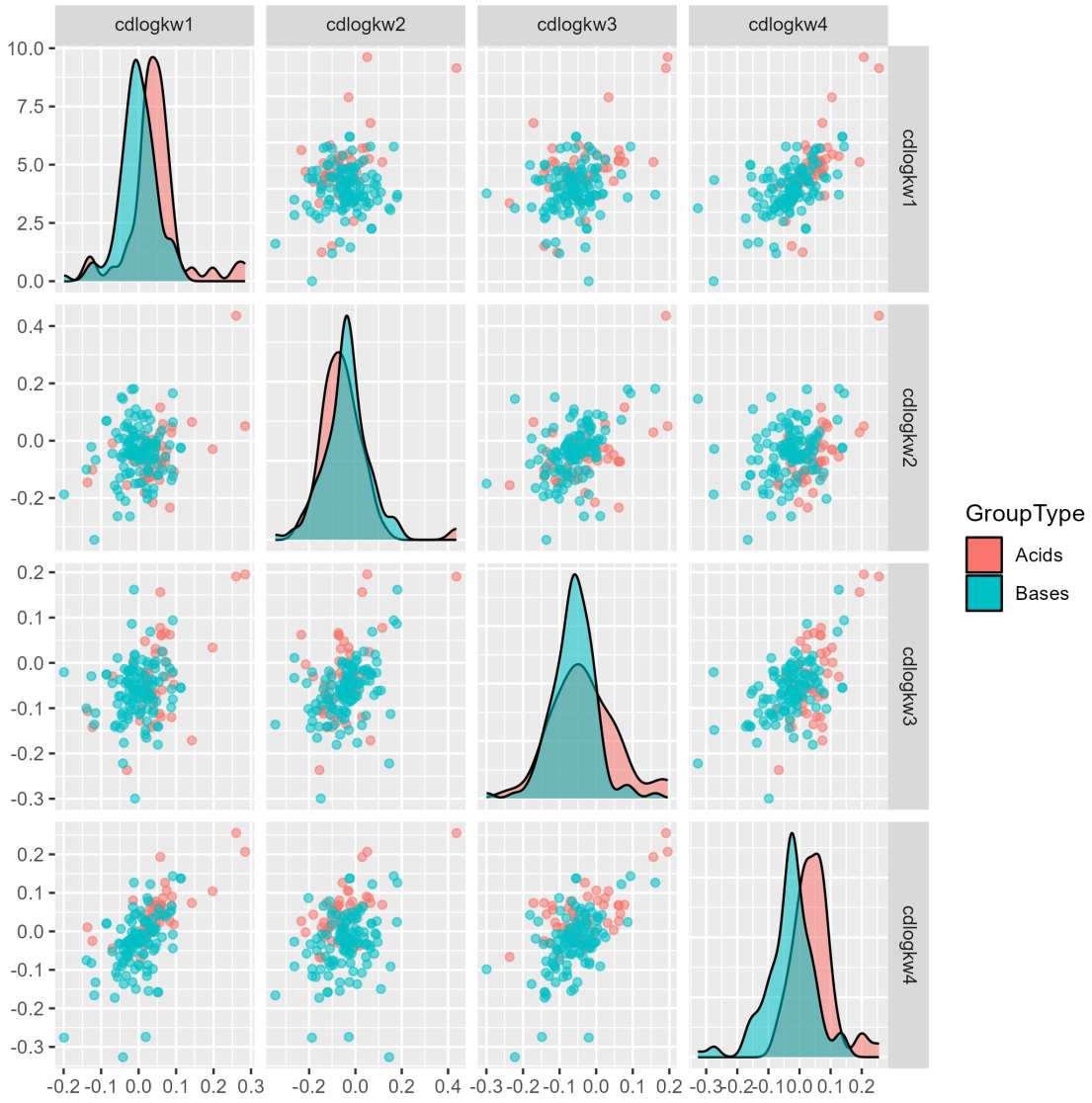


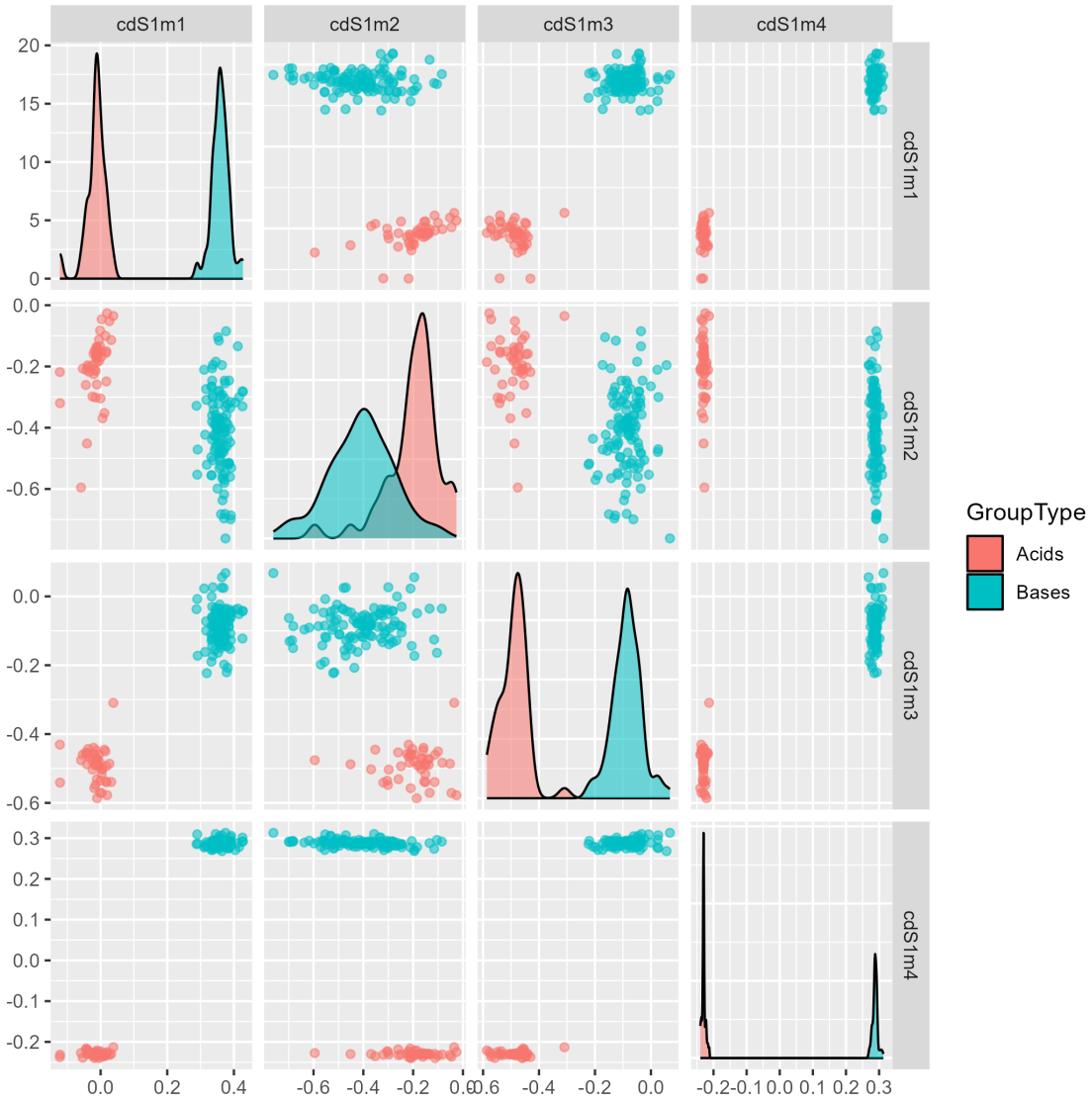
6.5.2 Effects of dissociation

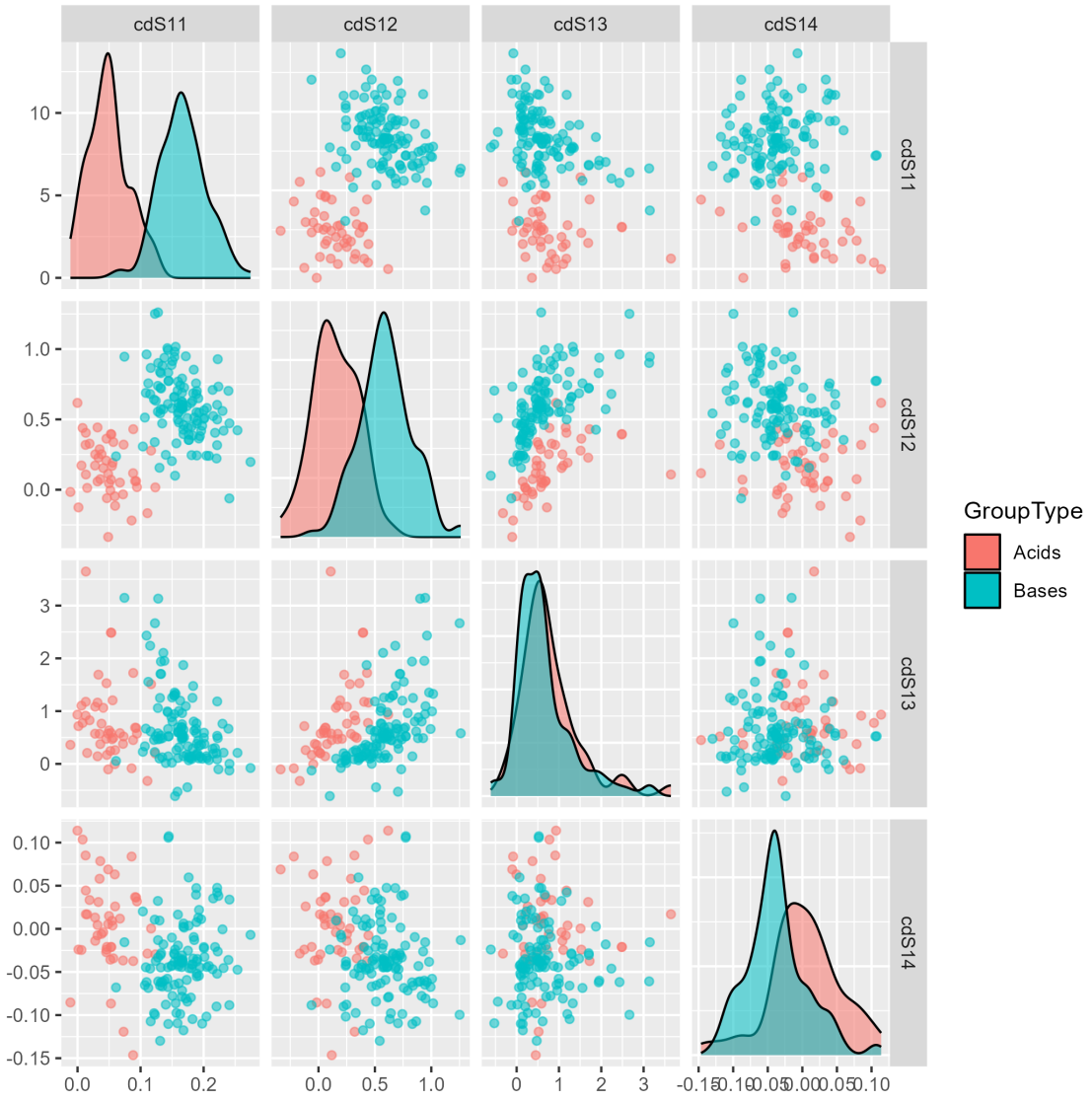
Individual parameters for the reference column (XBridge Shield RP18)



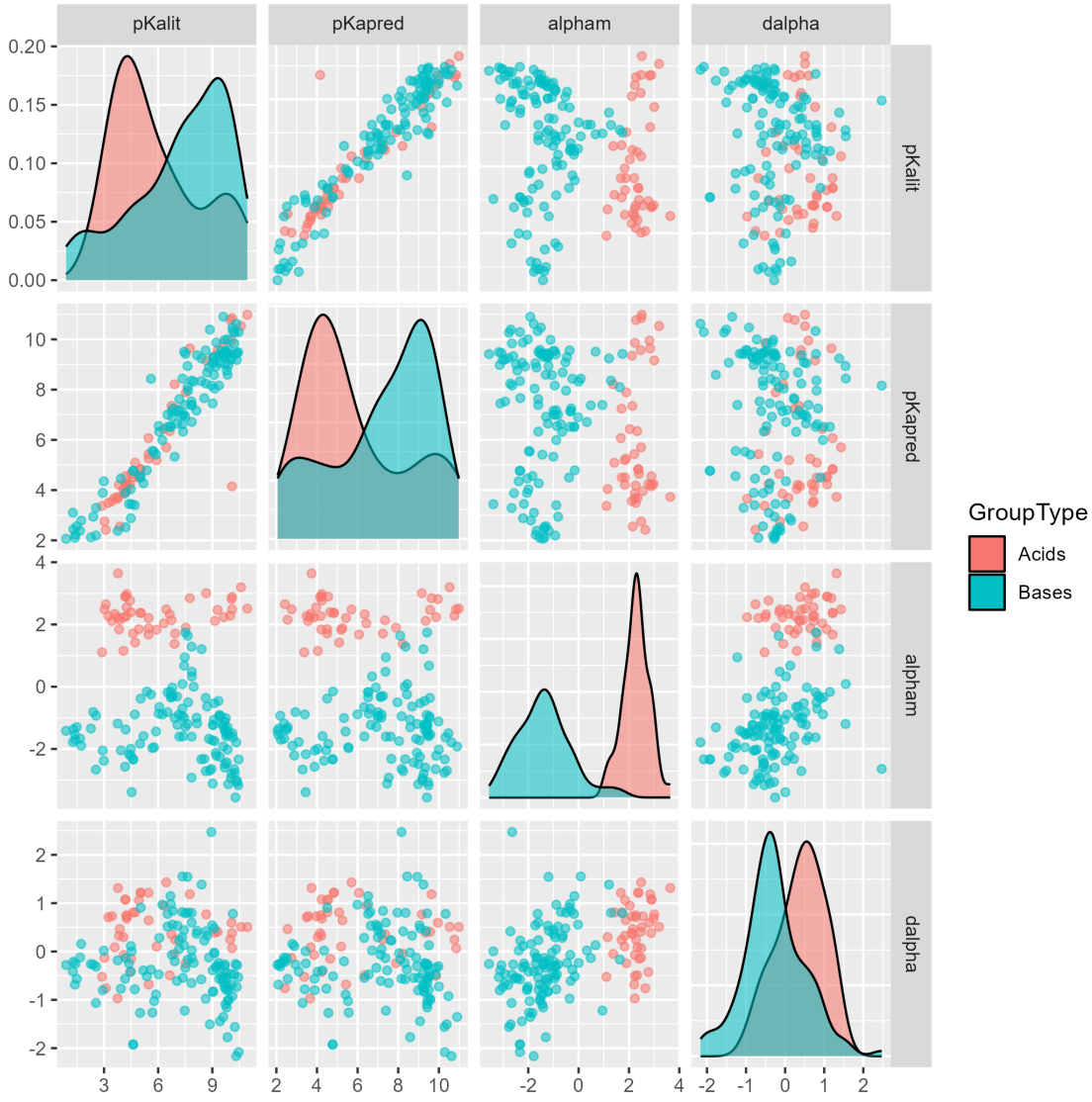
Individual parameters for the particular column (1-4) relative to XBridge Shield RP18


cdlogkw1
cdlogkw2
cdlogkw3
cdlogkw4
GroupType
Acids
Bases


cdS1m1
cdS1m2
cdS1m3
cdS1m4
GroupType
Acids
Bases


cdS11
cdS12
cdS13
cdS14
GroupType
Acids
Bases

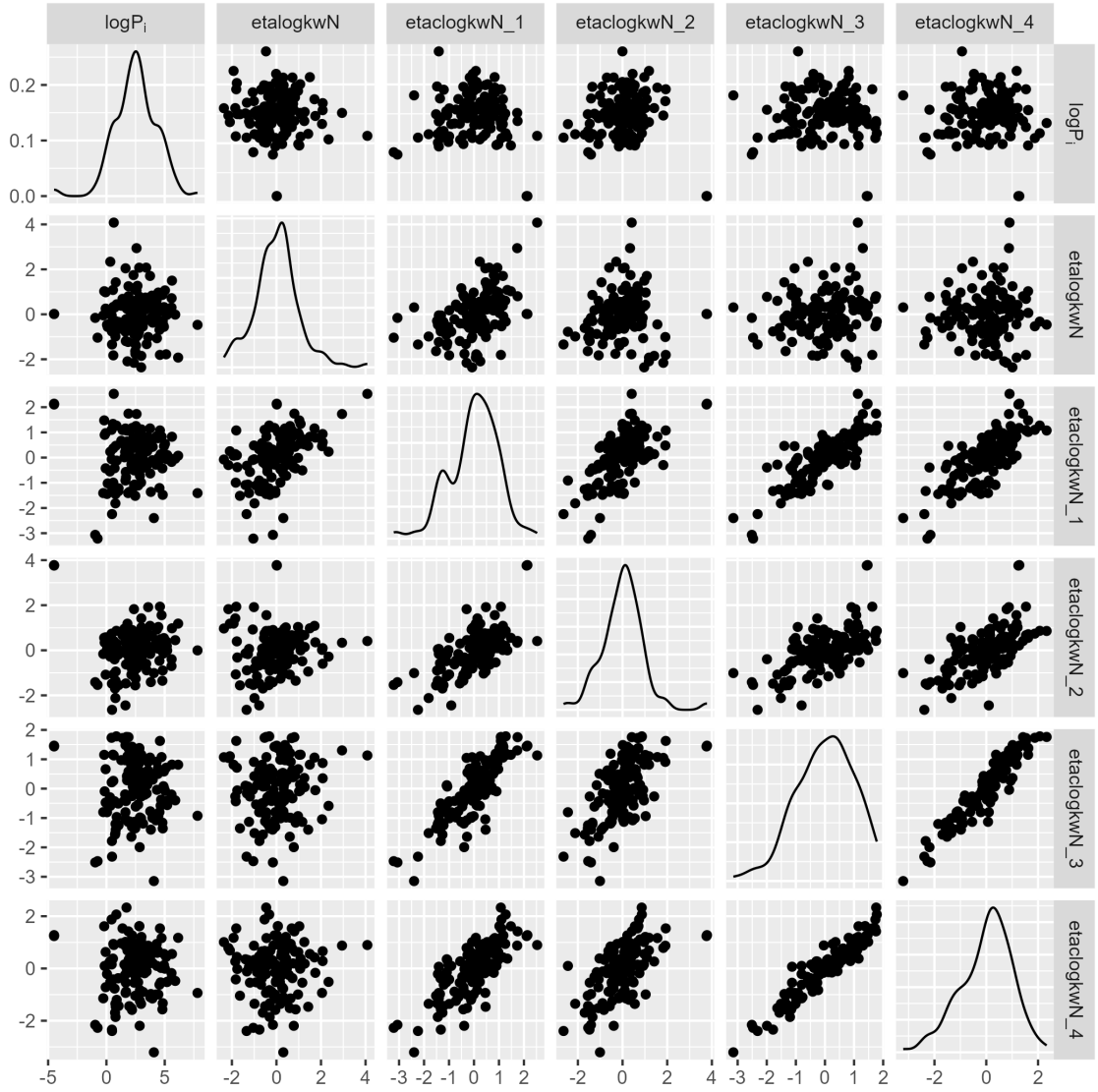
6.5.3 pKa-related paraemters

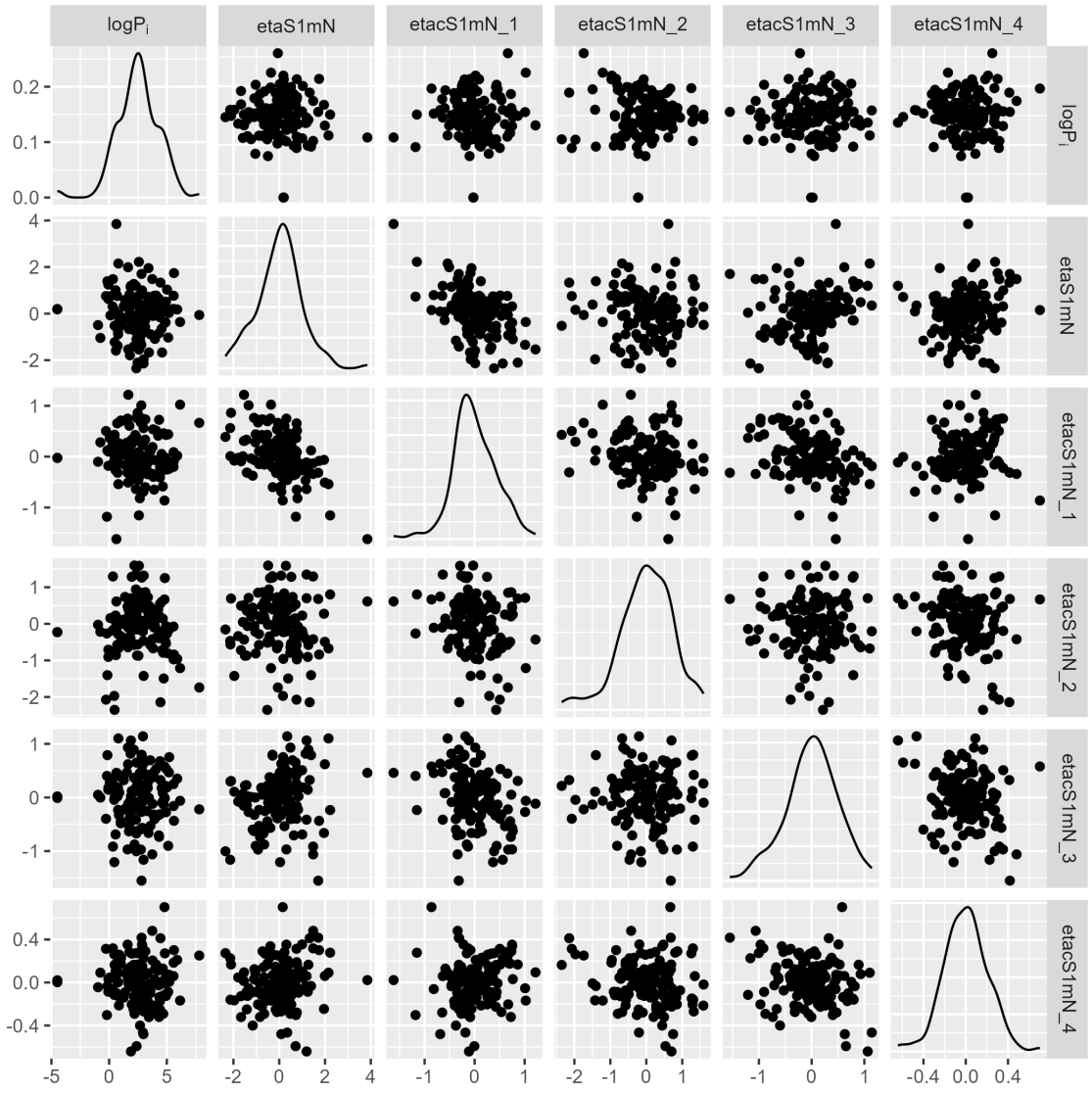


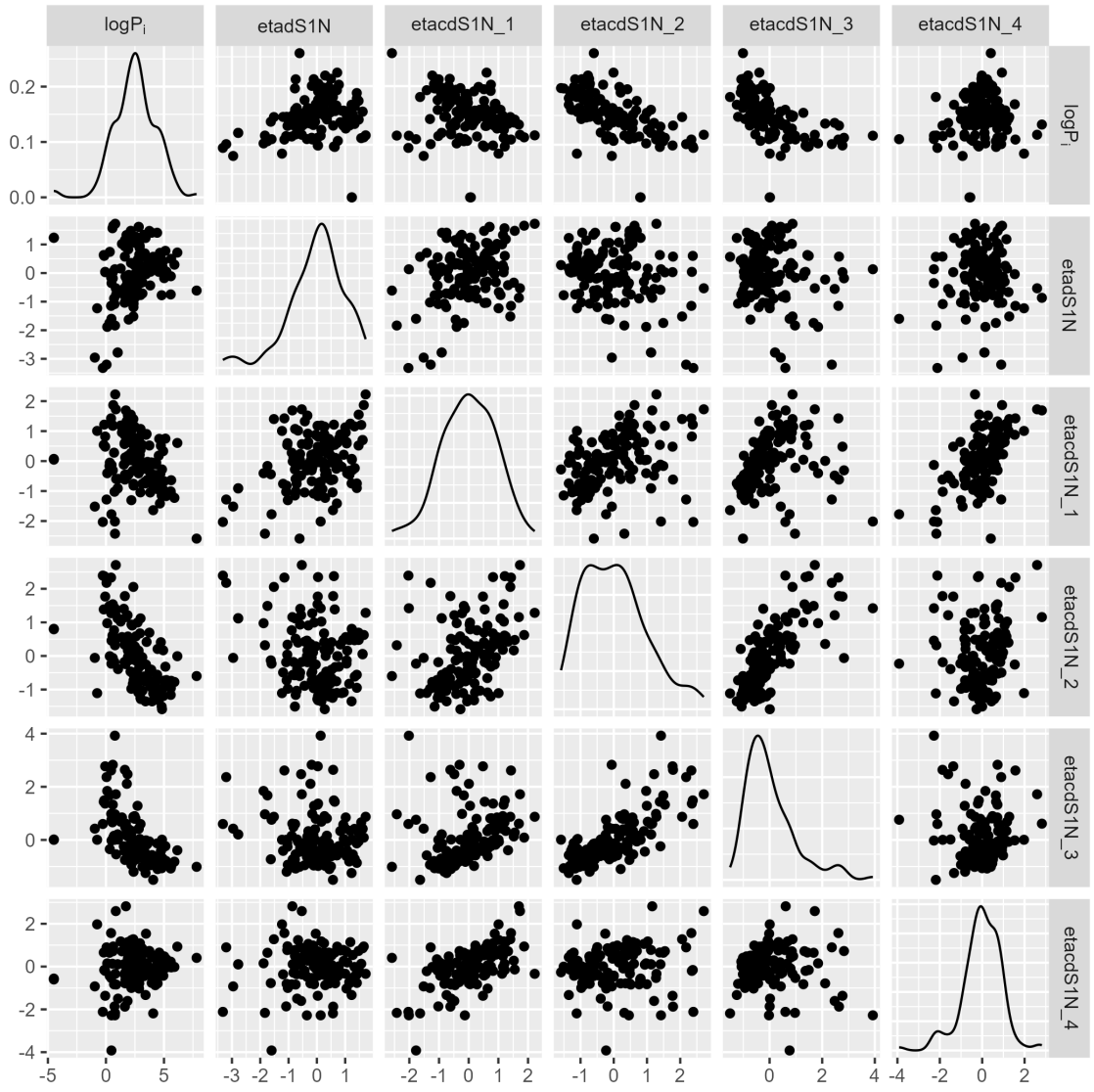
6.6 Eta plots

Eta plots shows the centered and standardized individual parameters (e.g. $\eta_{logkwN,i} = (logkwN_i - (\widehat{logkwN} + \beta_1 \cdot \log P_i))/\omega_1$). They allow to visualize the unexplained between-analyte variability of chromatographic parameters.

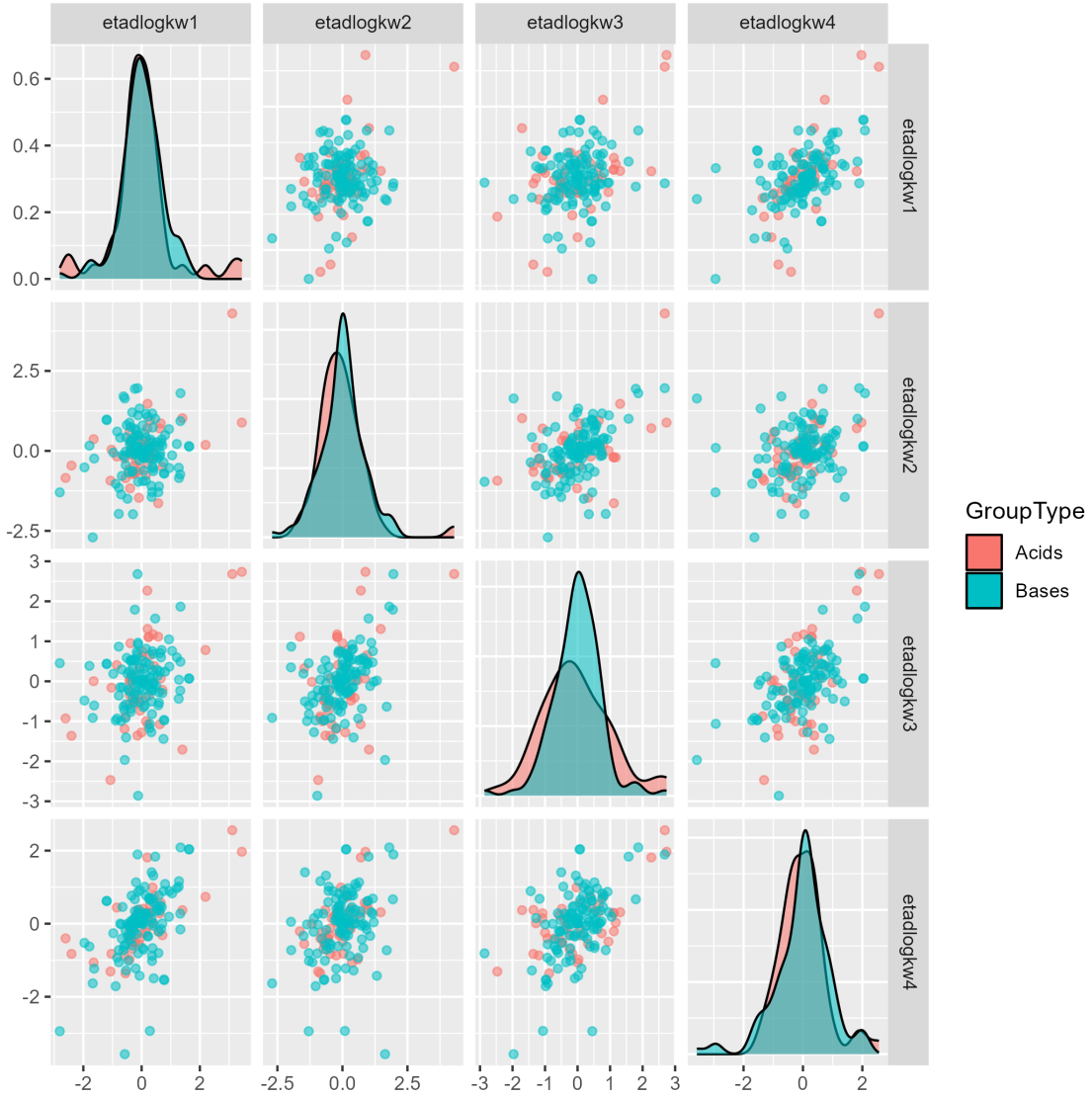
6.6.1 Neutral Forms

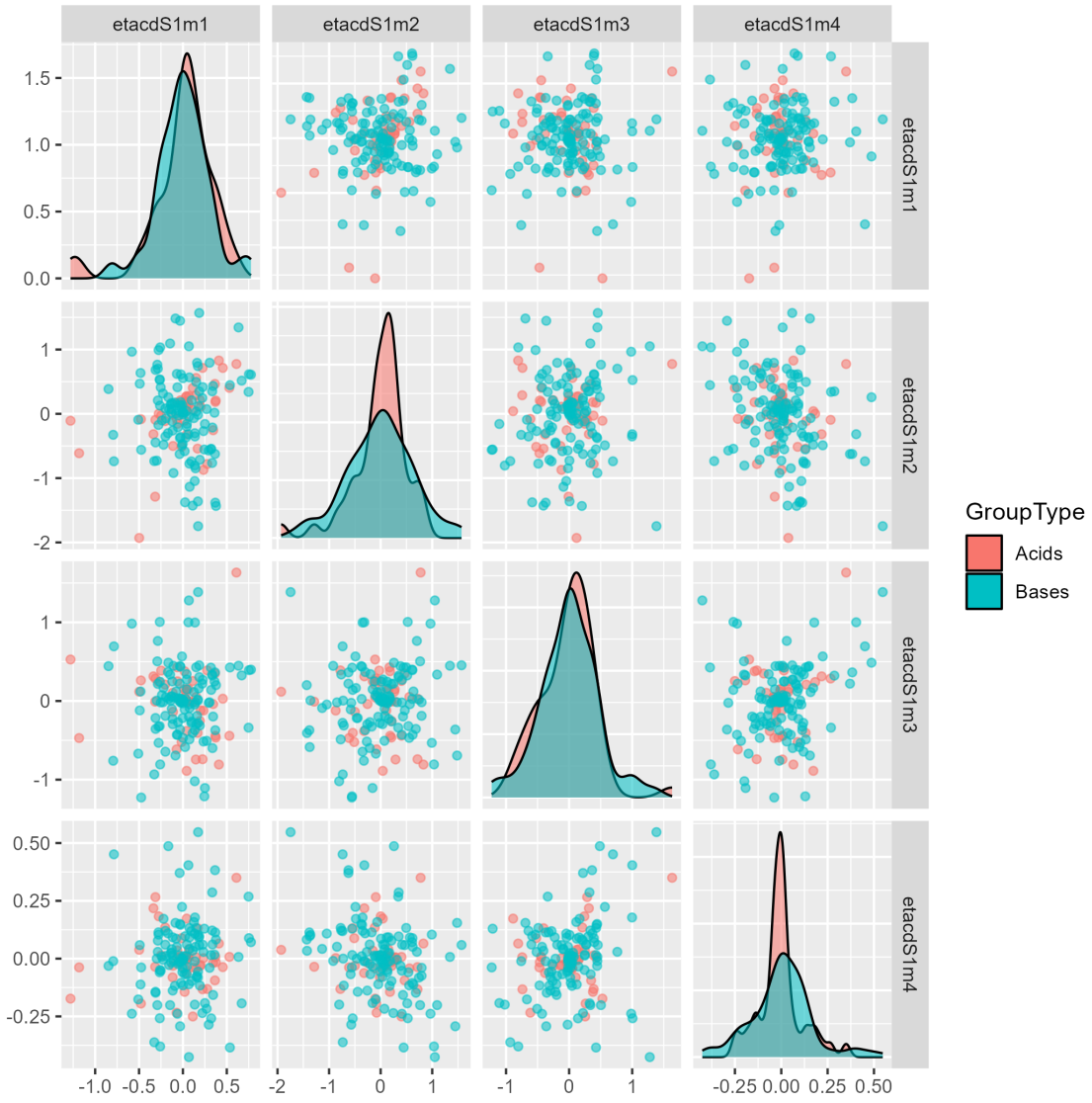


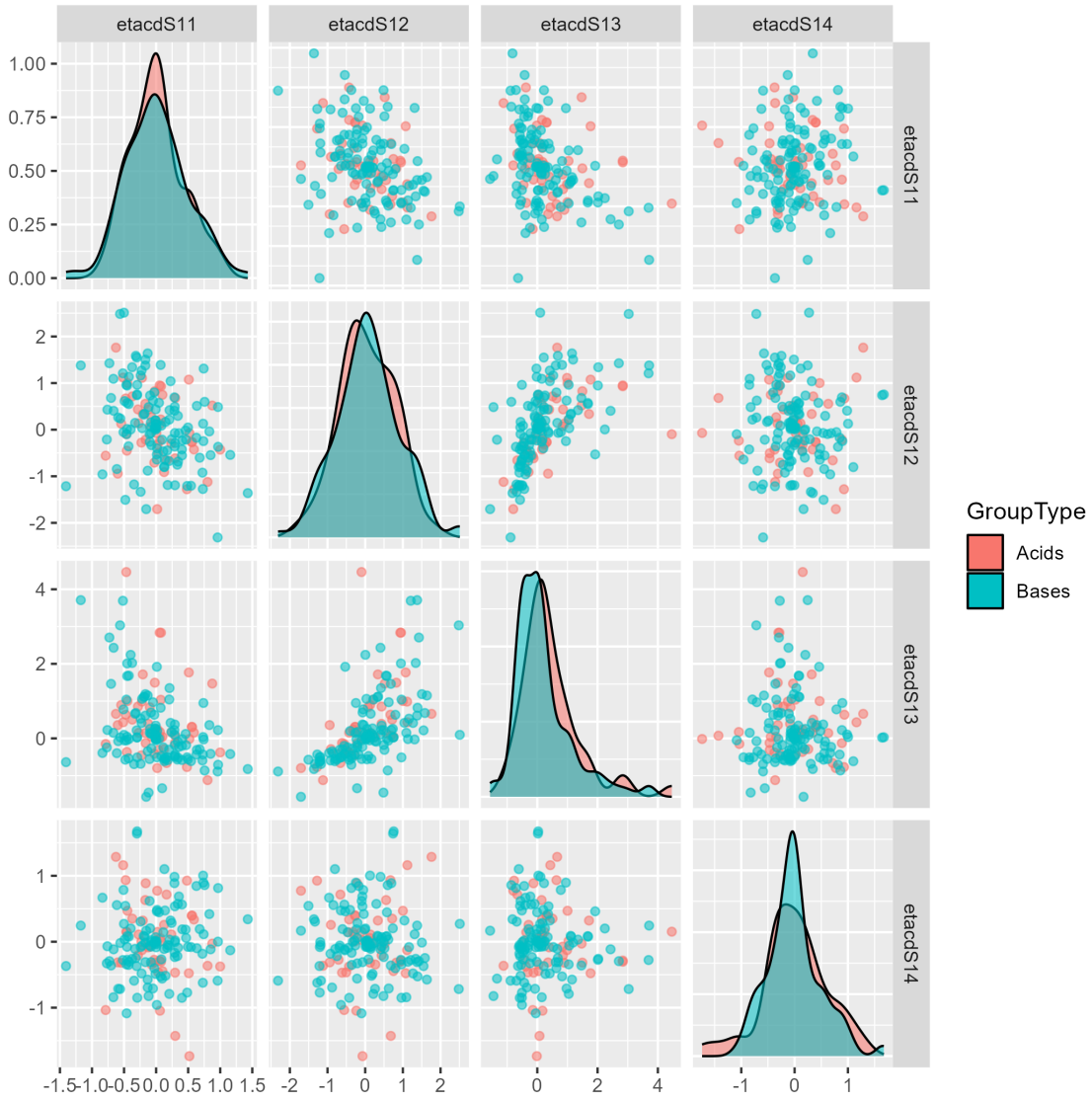

logP_i
etaS1mN
etacS1mN_1
etacS1mN_2
etacS1mN_3
etacS1mN_4


logP_i
etadS1N
etacdS1N_1
etacdS1N_2
etacdS1N_3
etacdS1N_4

6.6.2 Effect of dissociation

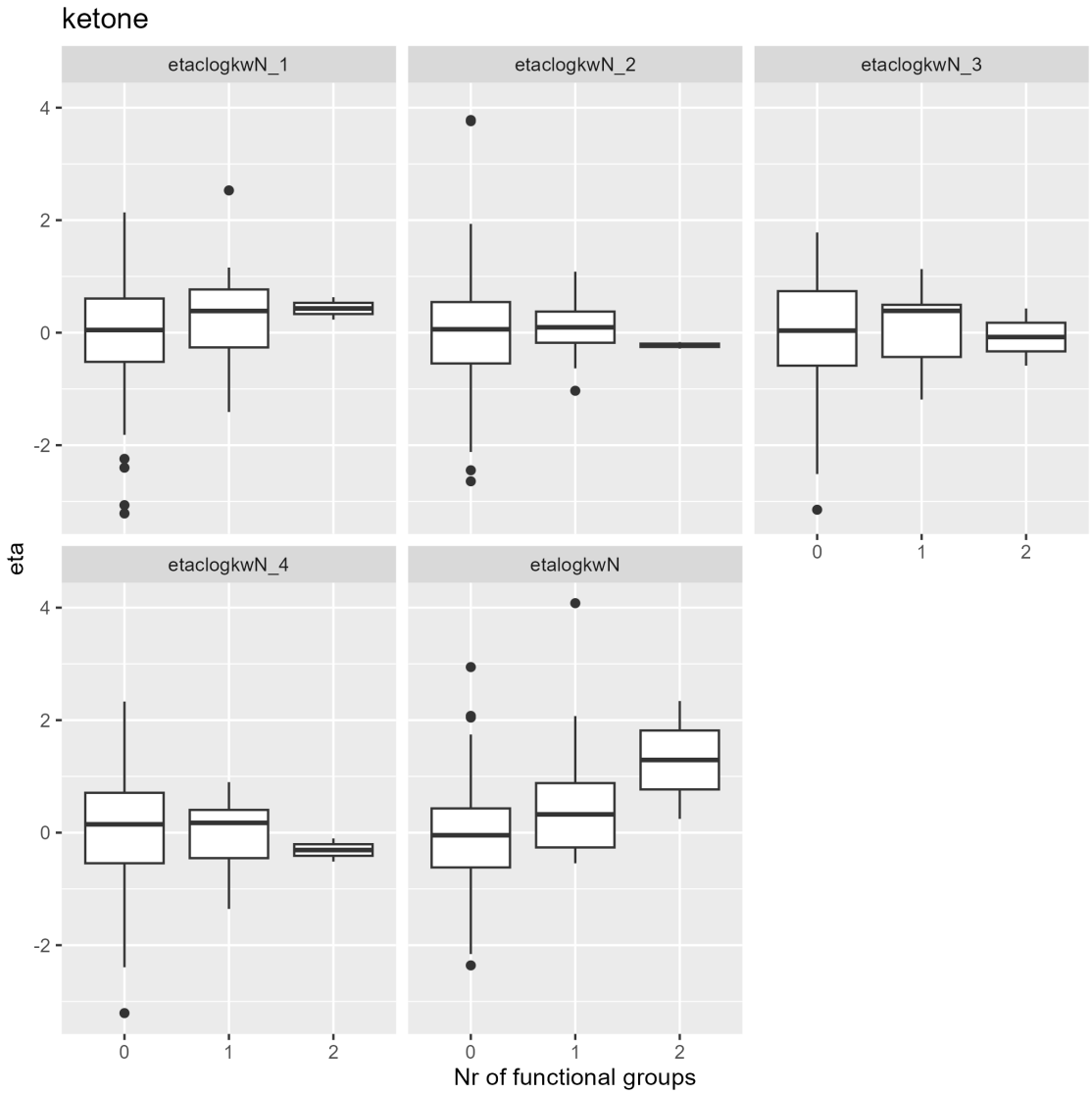


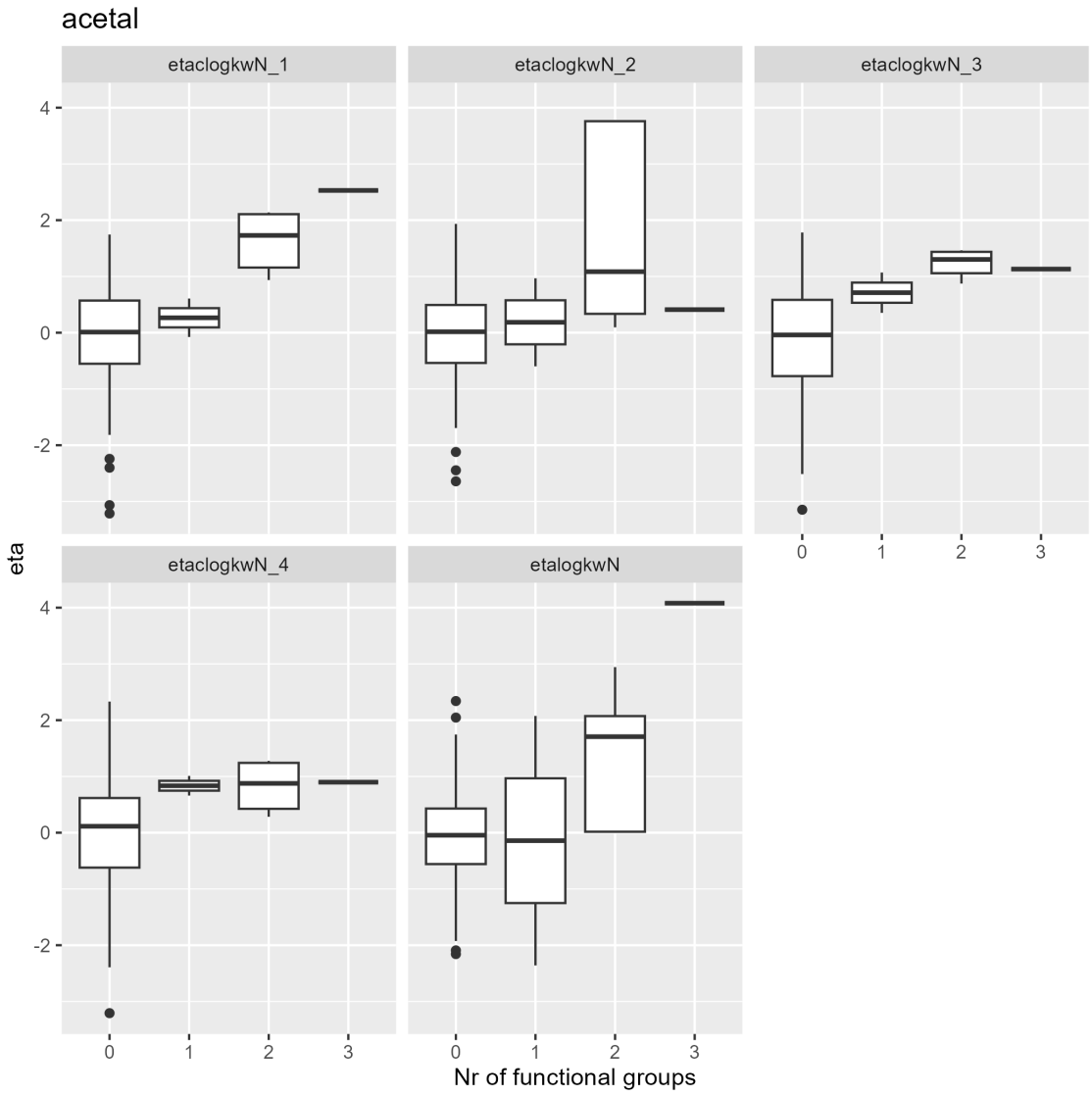

etacdS1m1
etacdS1m2
etacdS1m3
etacdS1m4
GroupType
Acids
Bases

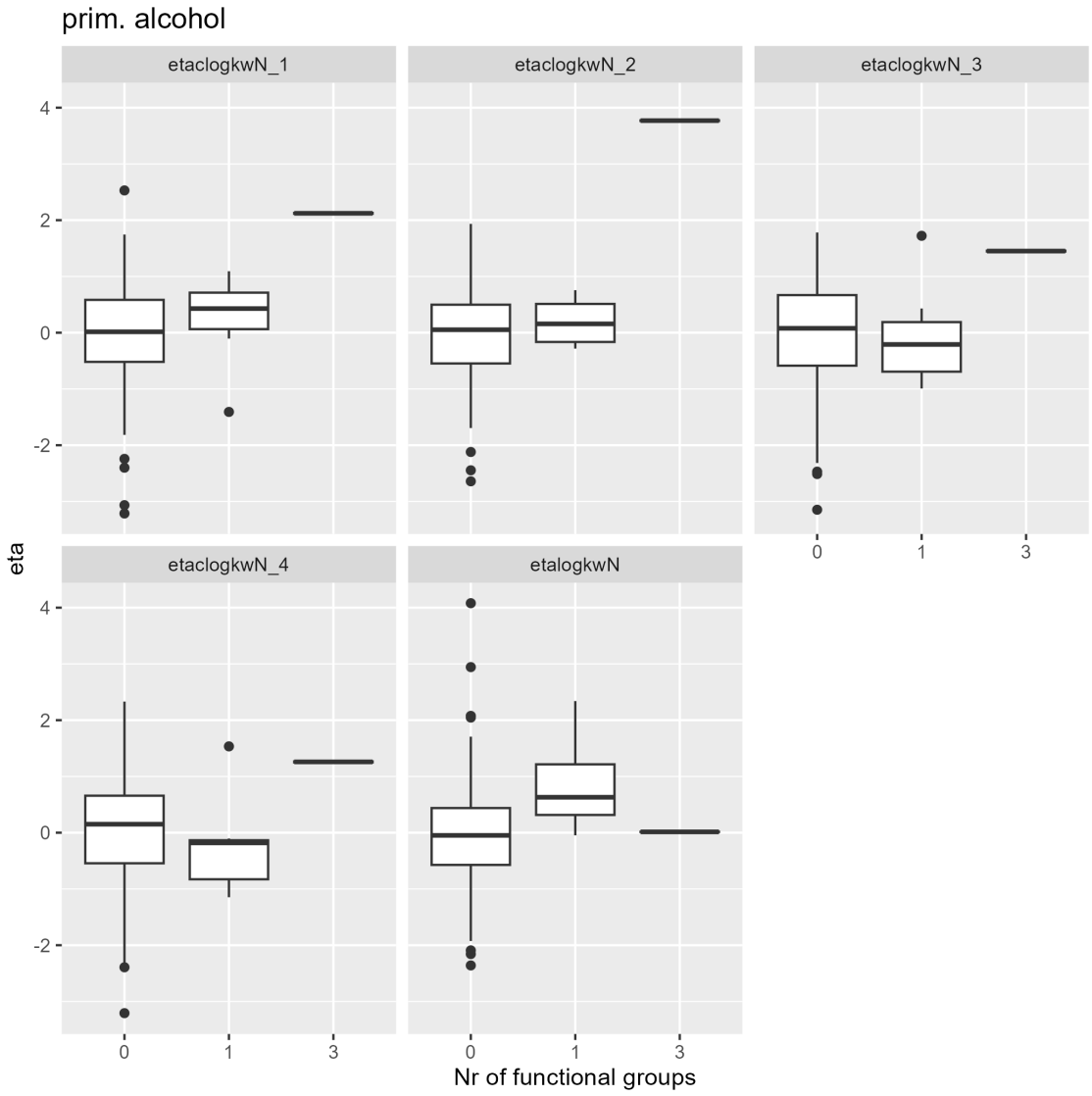


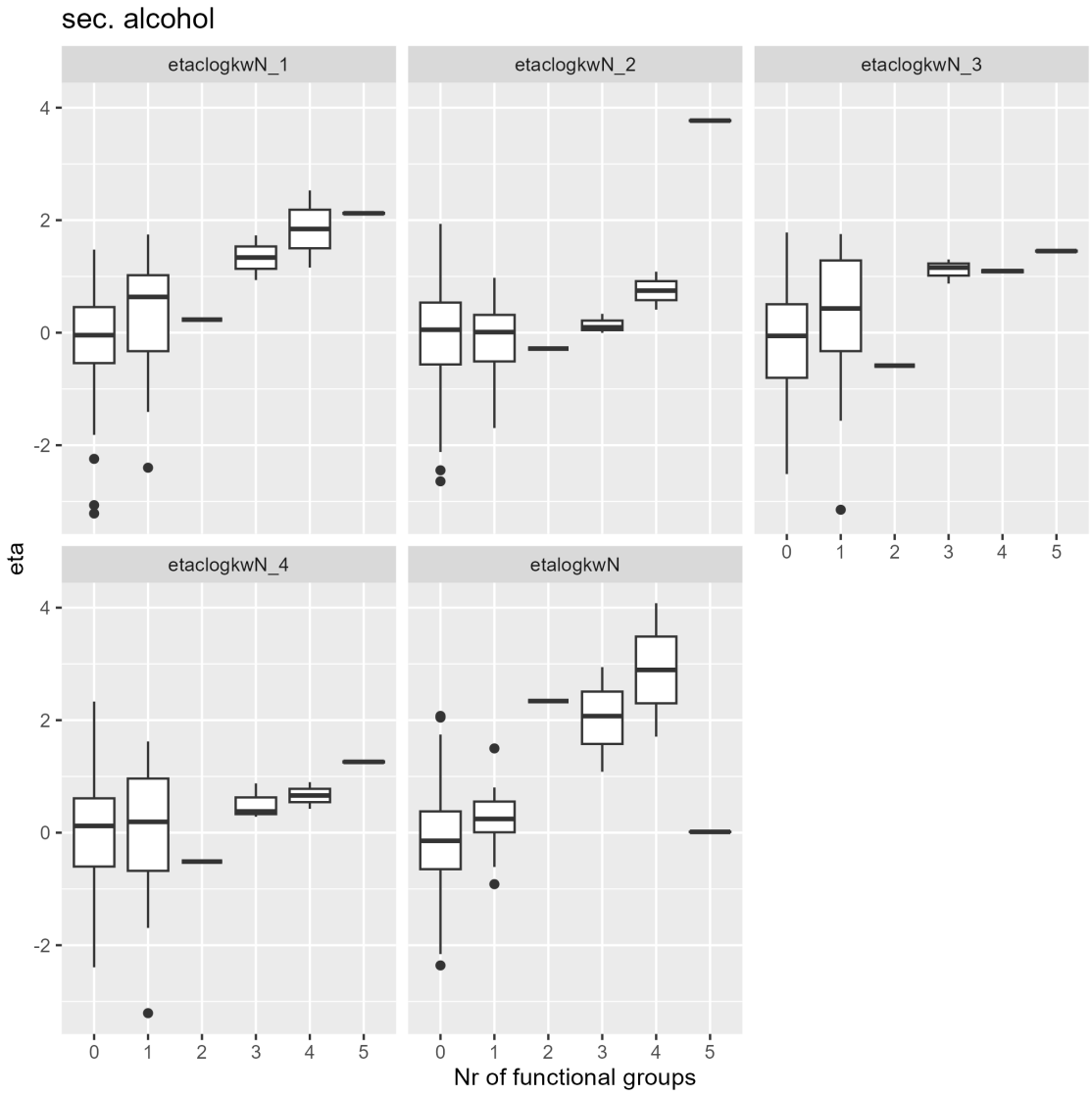
6.6.3 Effect of functional groups (exploratory)

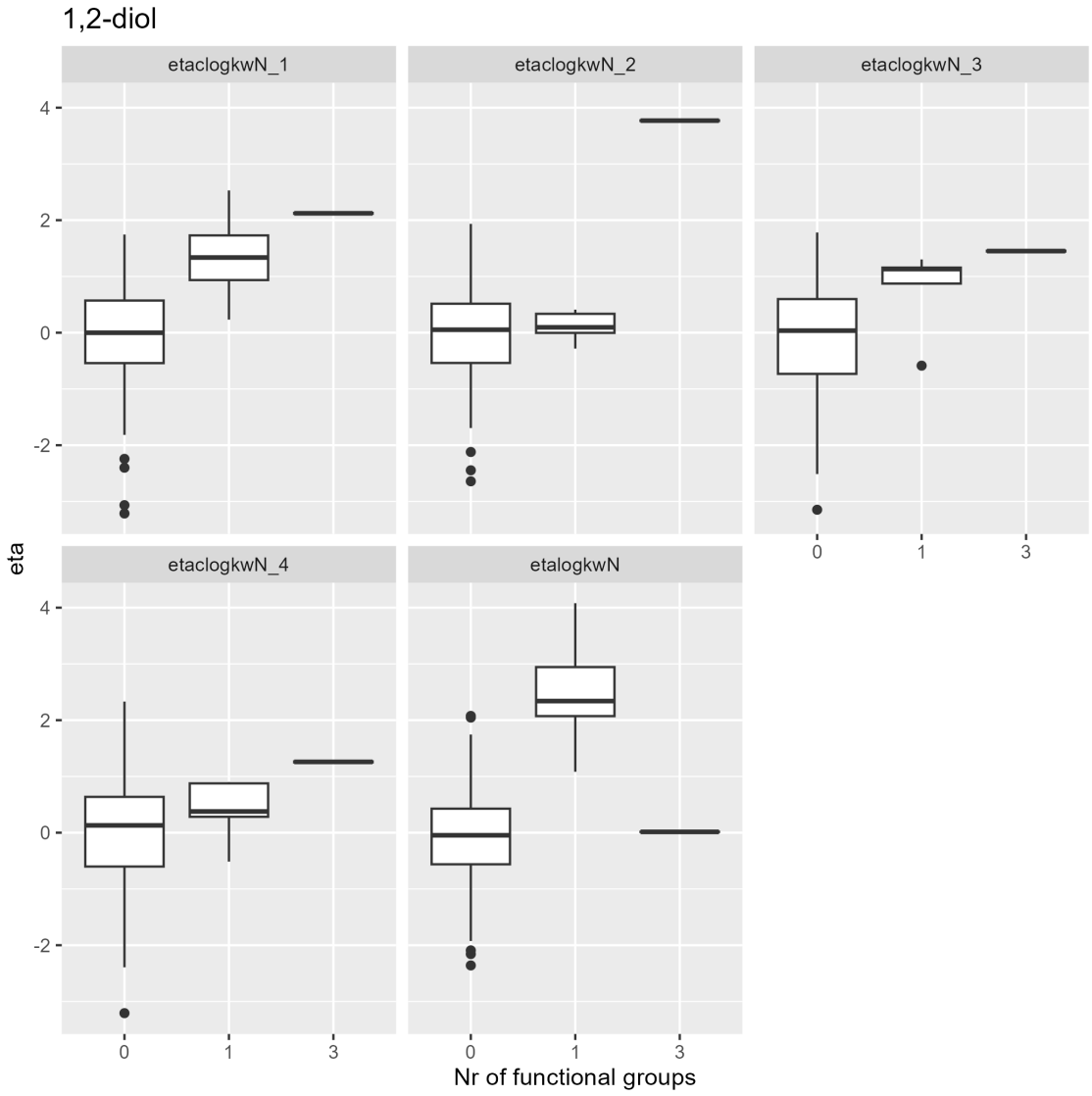
The following ETA plots present the relationship between individual eta values for logkwN and number of functional groups. This part is exploratory. Graphs are shown if there are at least 10 functional groups present in the dataset.

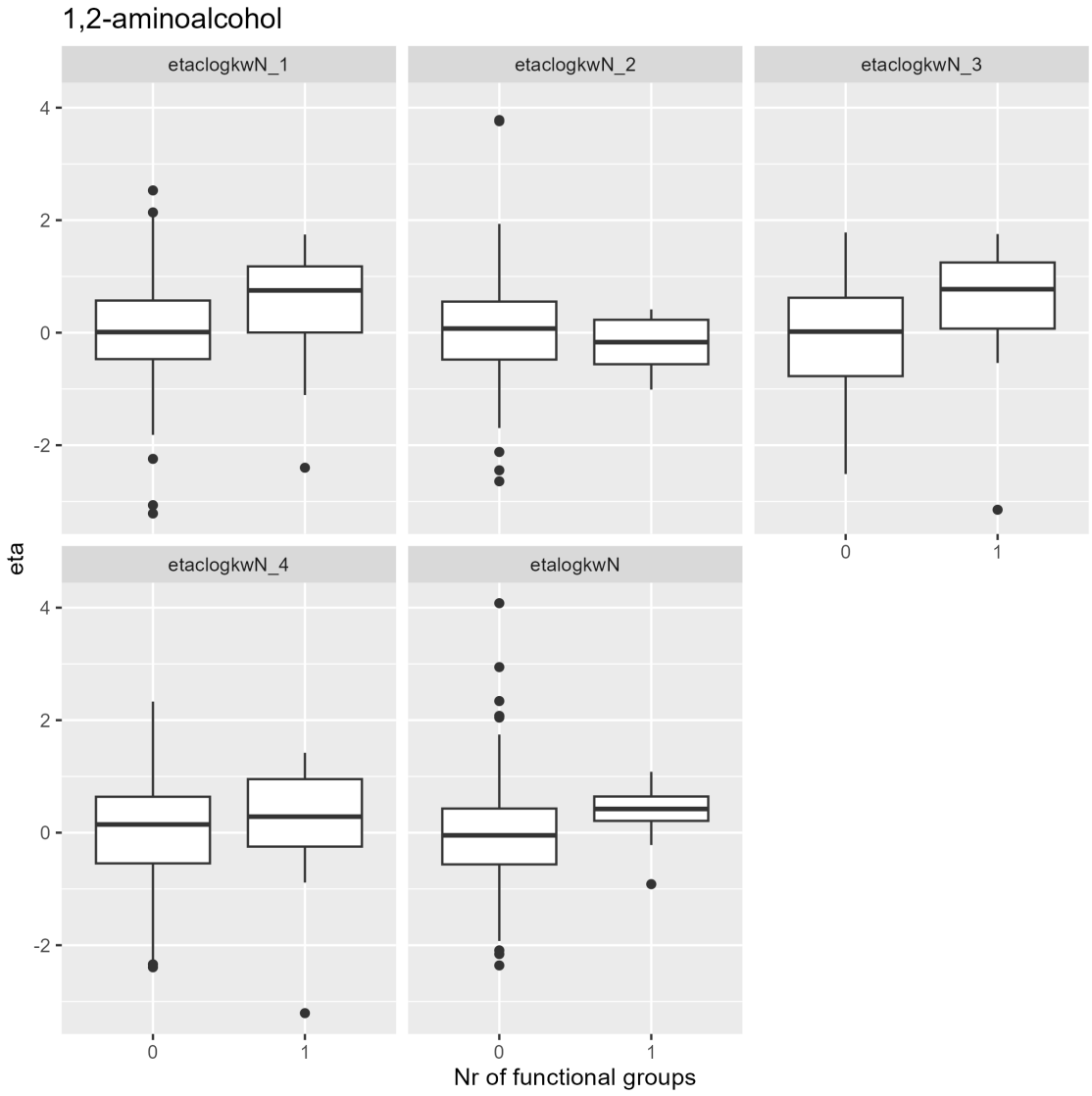

ketone
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups

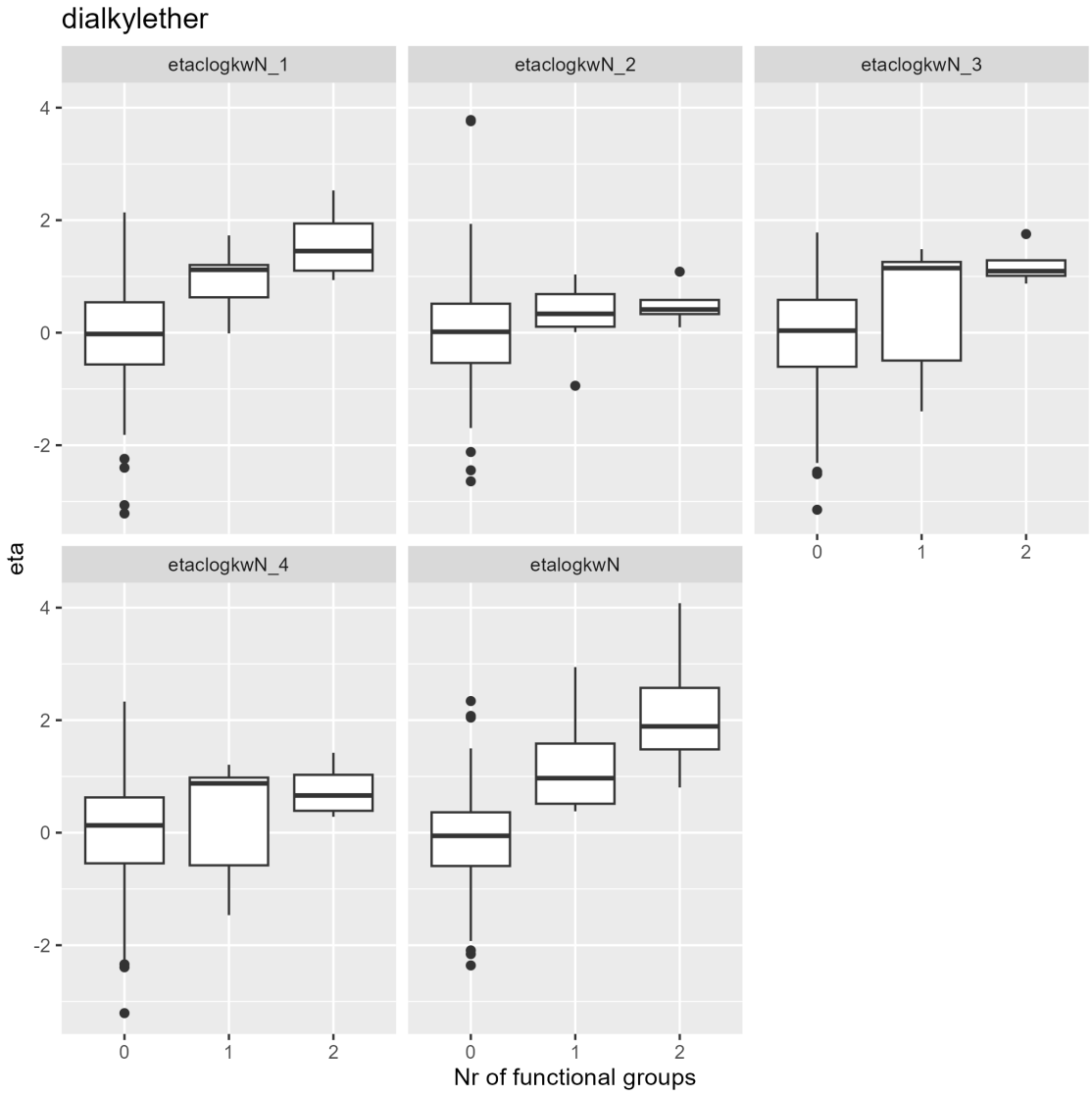

acetal
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
Nr of functional groups
4
2
0
-2
0
1
2
3

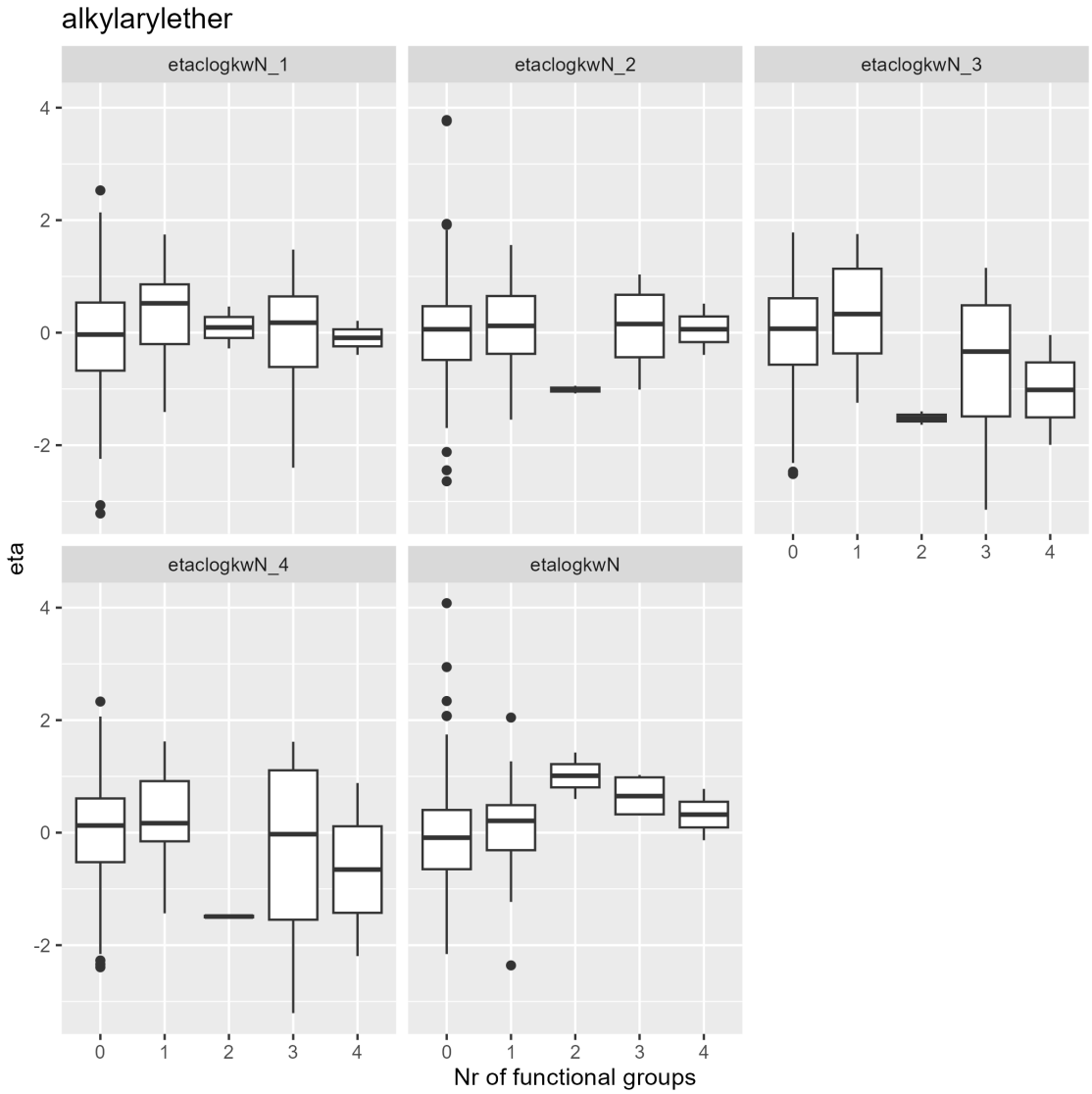

prim. alcohol
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
3
Nr of functional groups

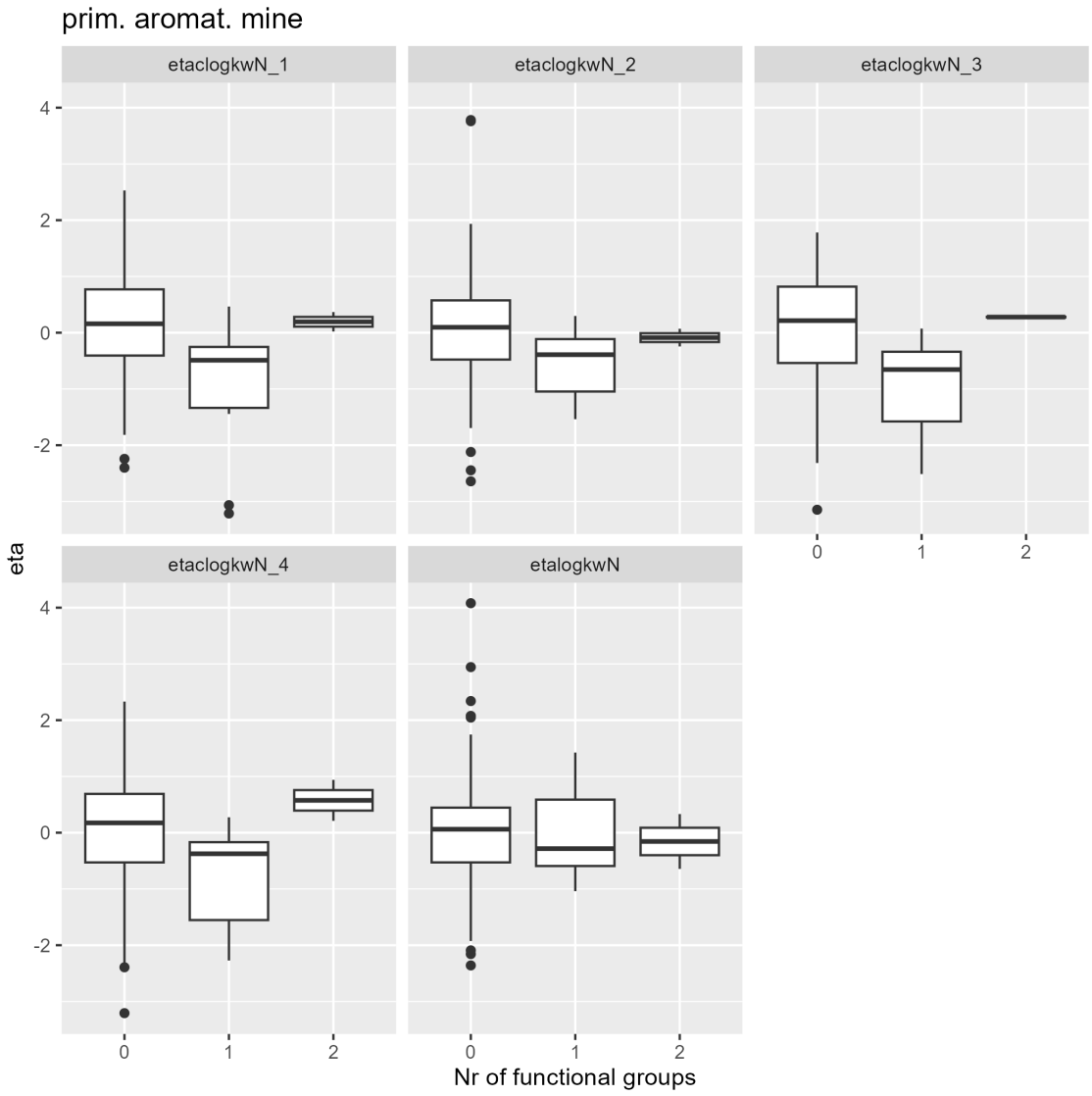

sec. alcohol
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
Nr of functional groups

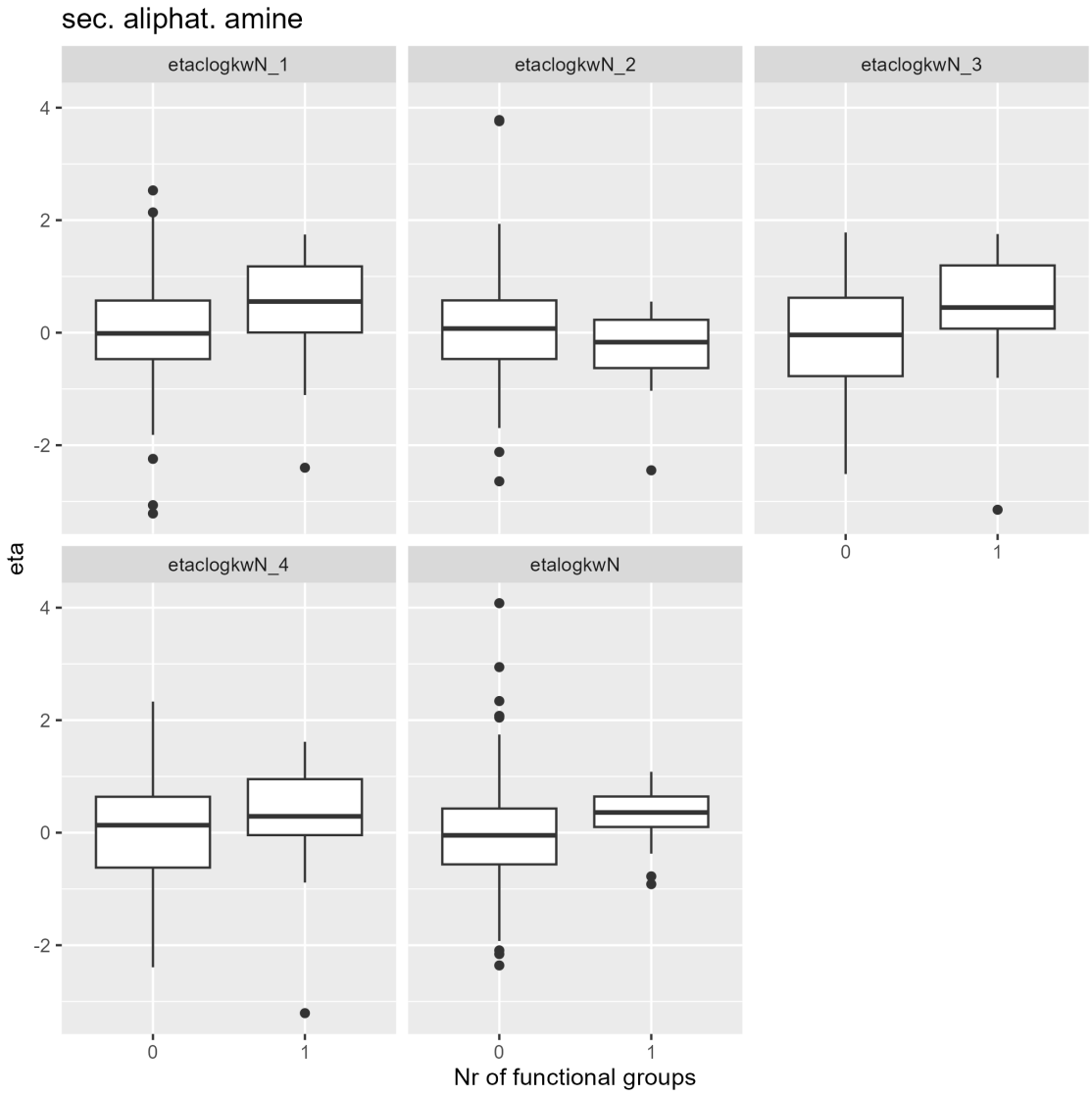

1,2-diol
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
3
Nr of functional groups

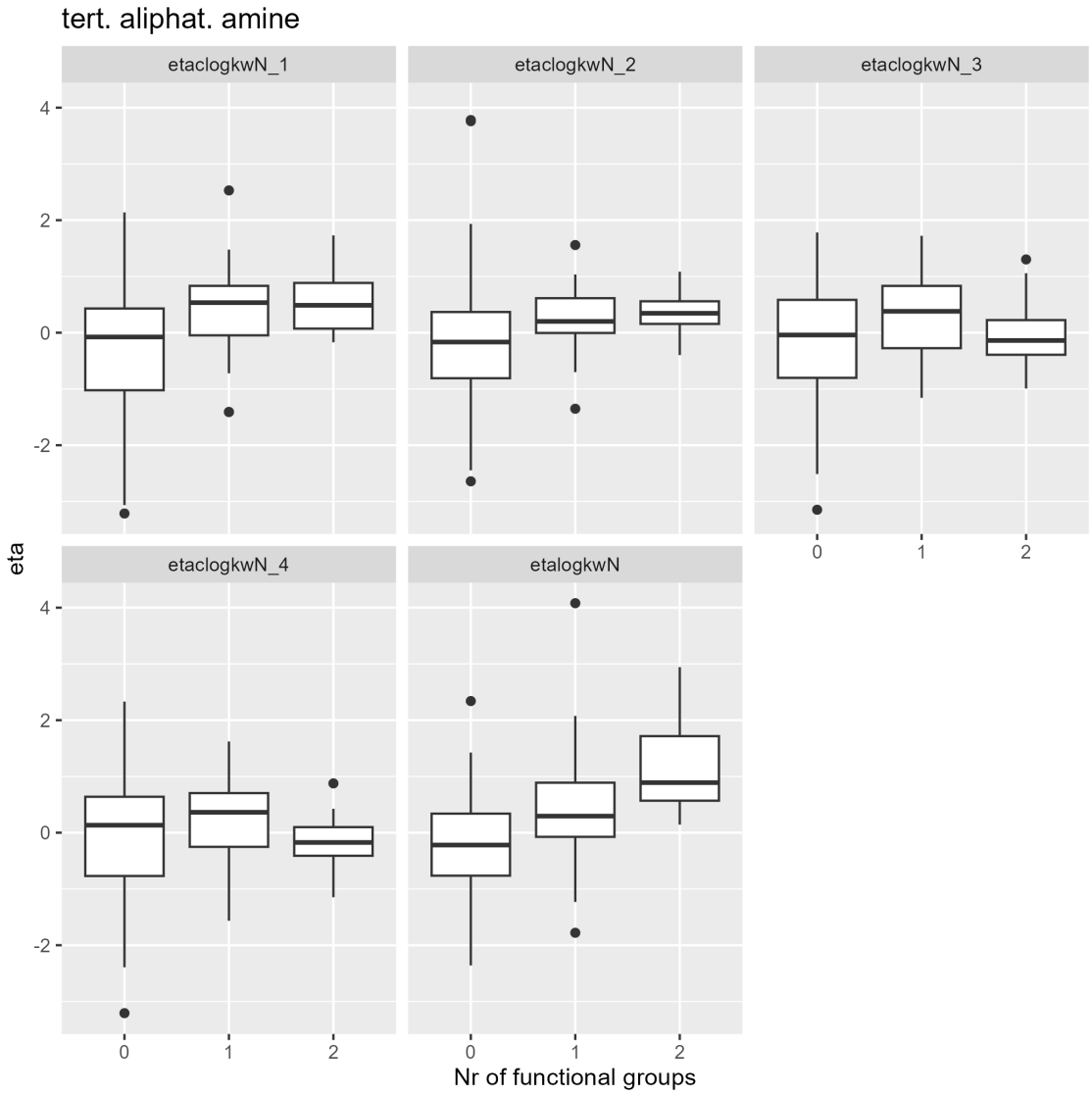

1,2-aminoalcohol
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
Nr of functional groups

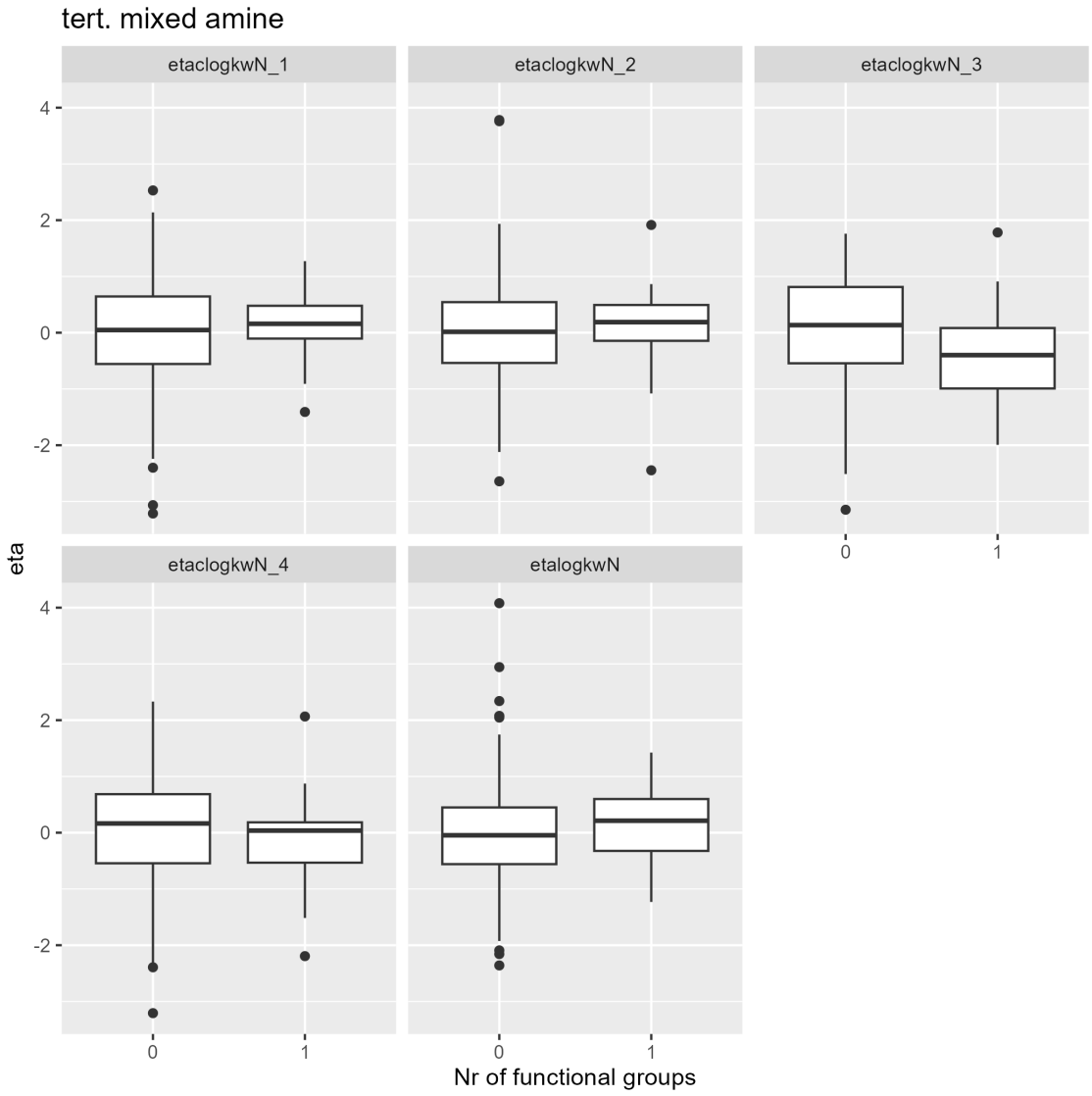

dialkylether
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups

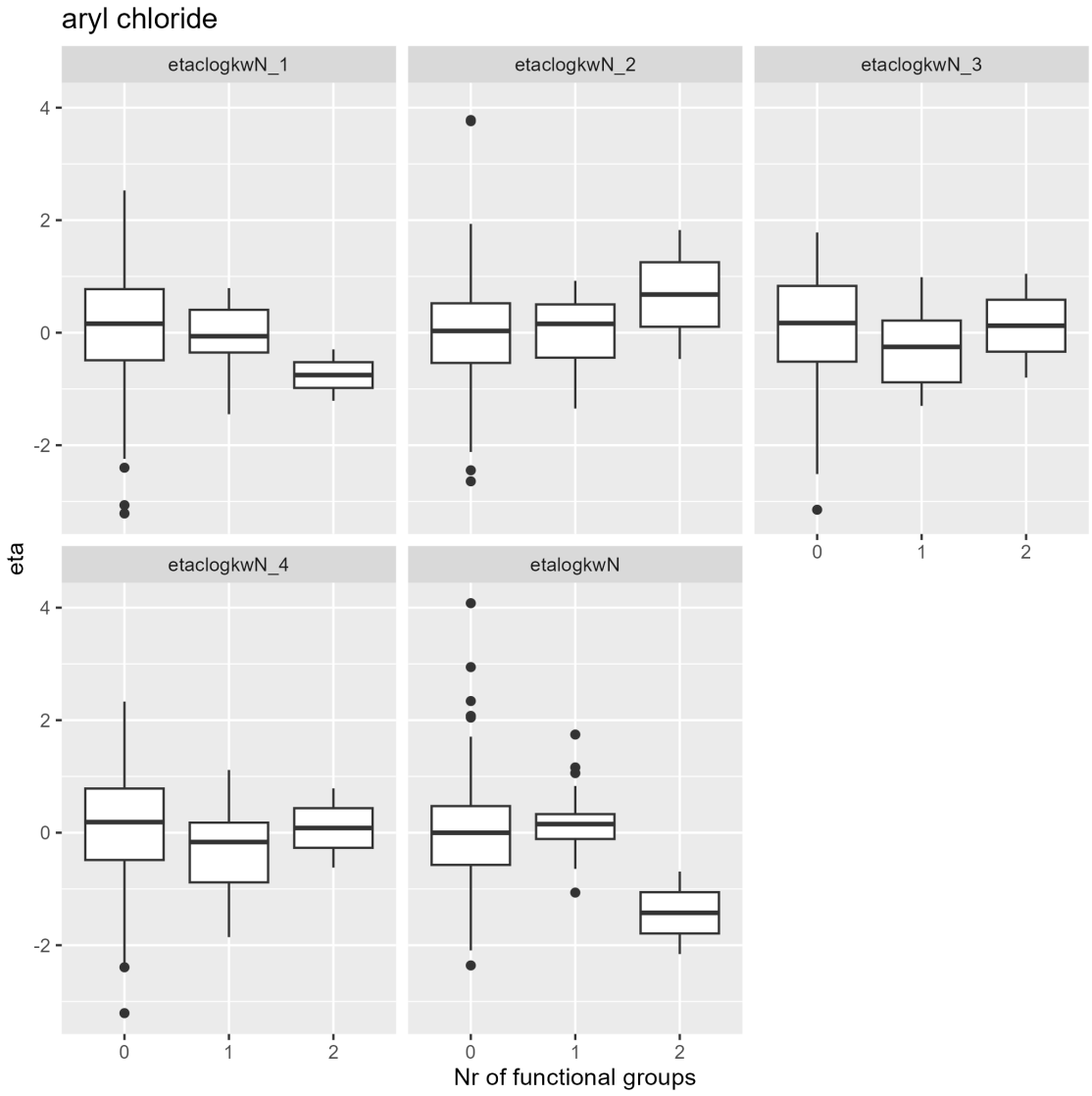

alkylarylether
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
Nr of functional groups
4
2
0
-2
0
1
2
3
4

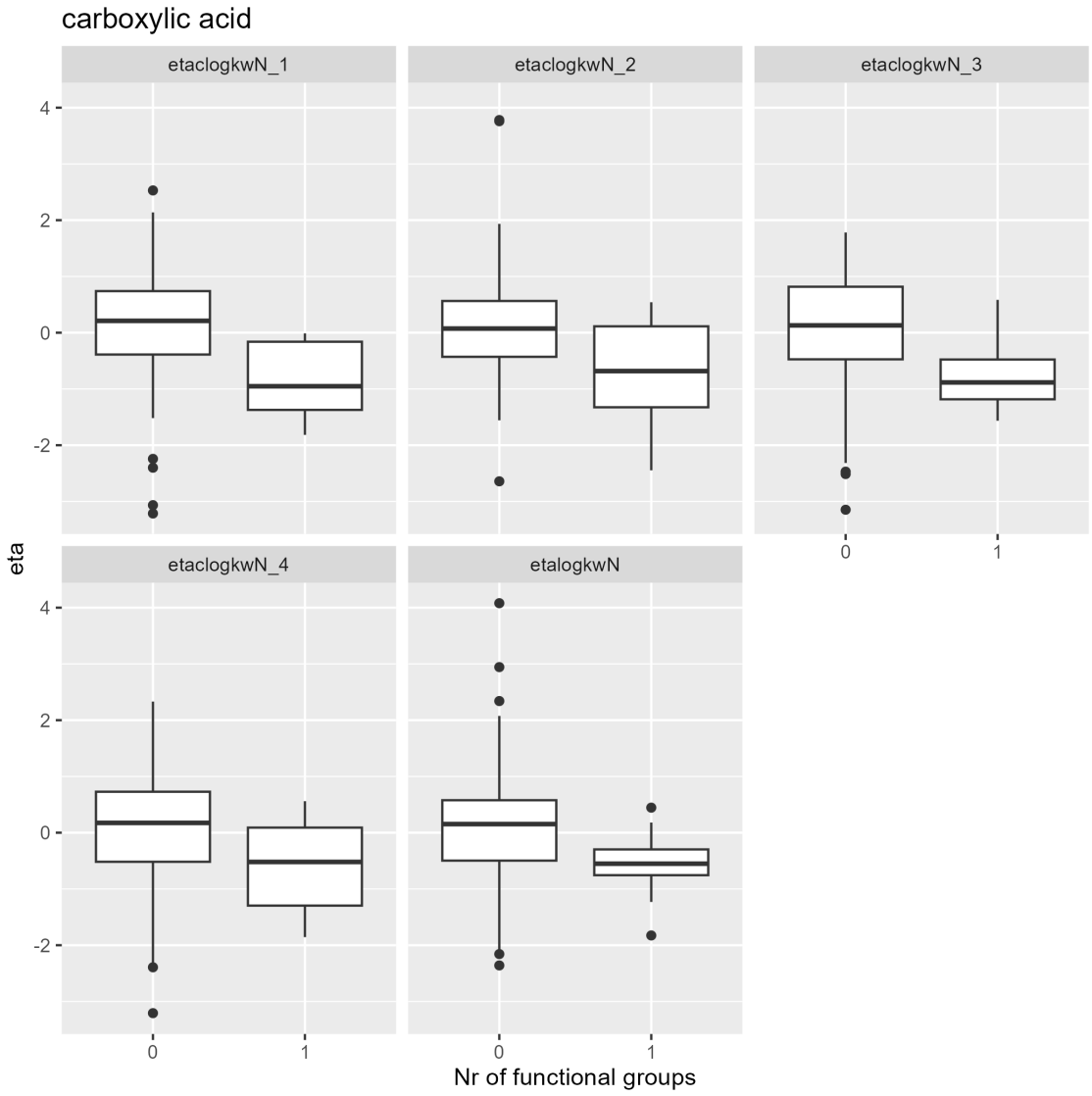

prim. aromat. mine
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups

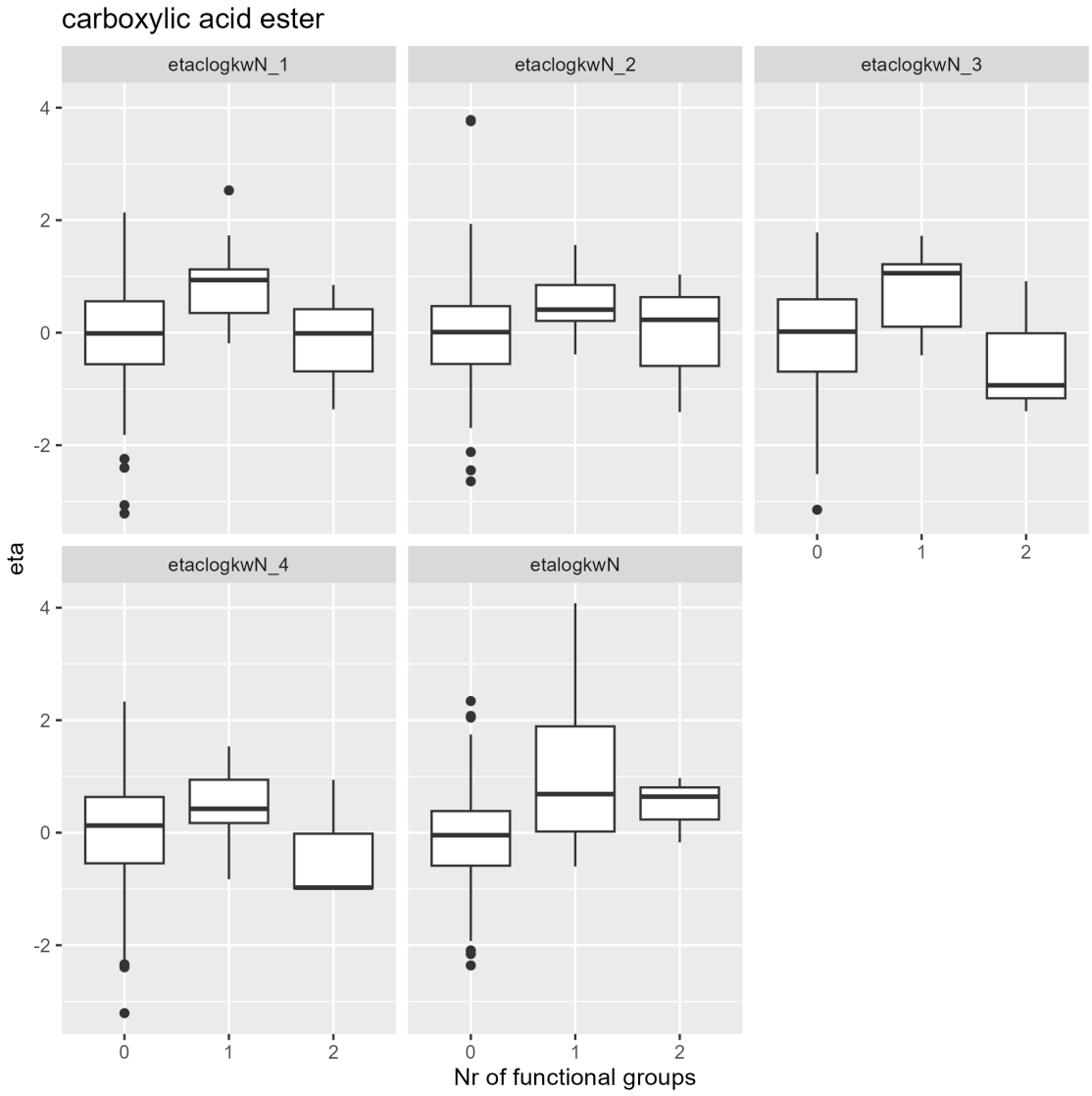

sec. aliphat. amine
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
Nr of functional groups

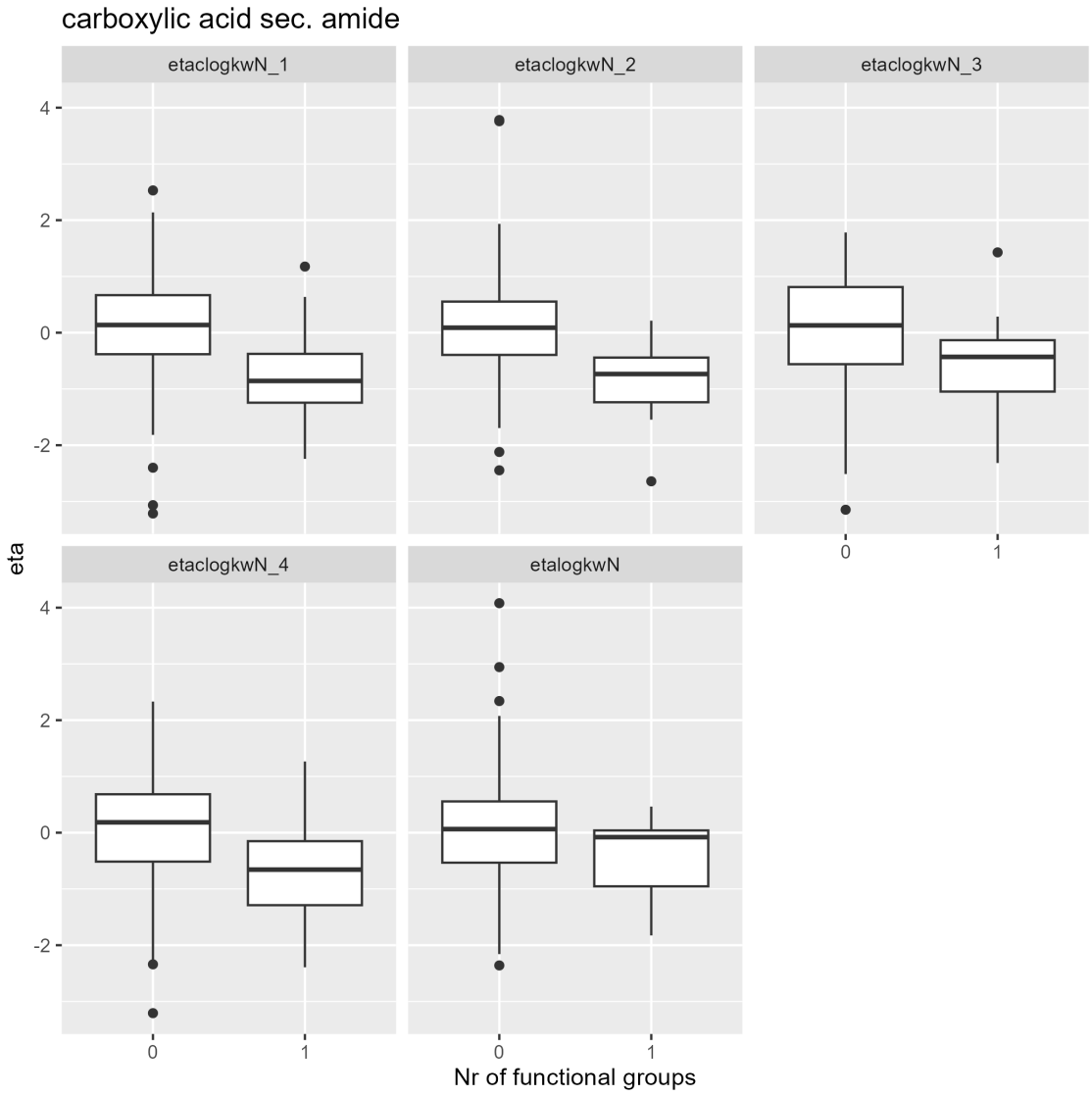

tert. aliphat. amine
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups

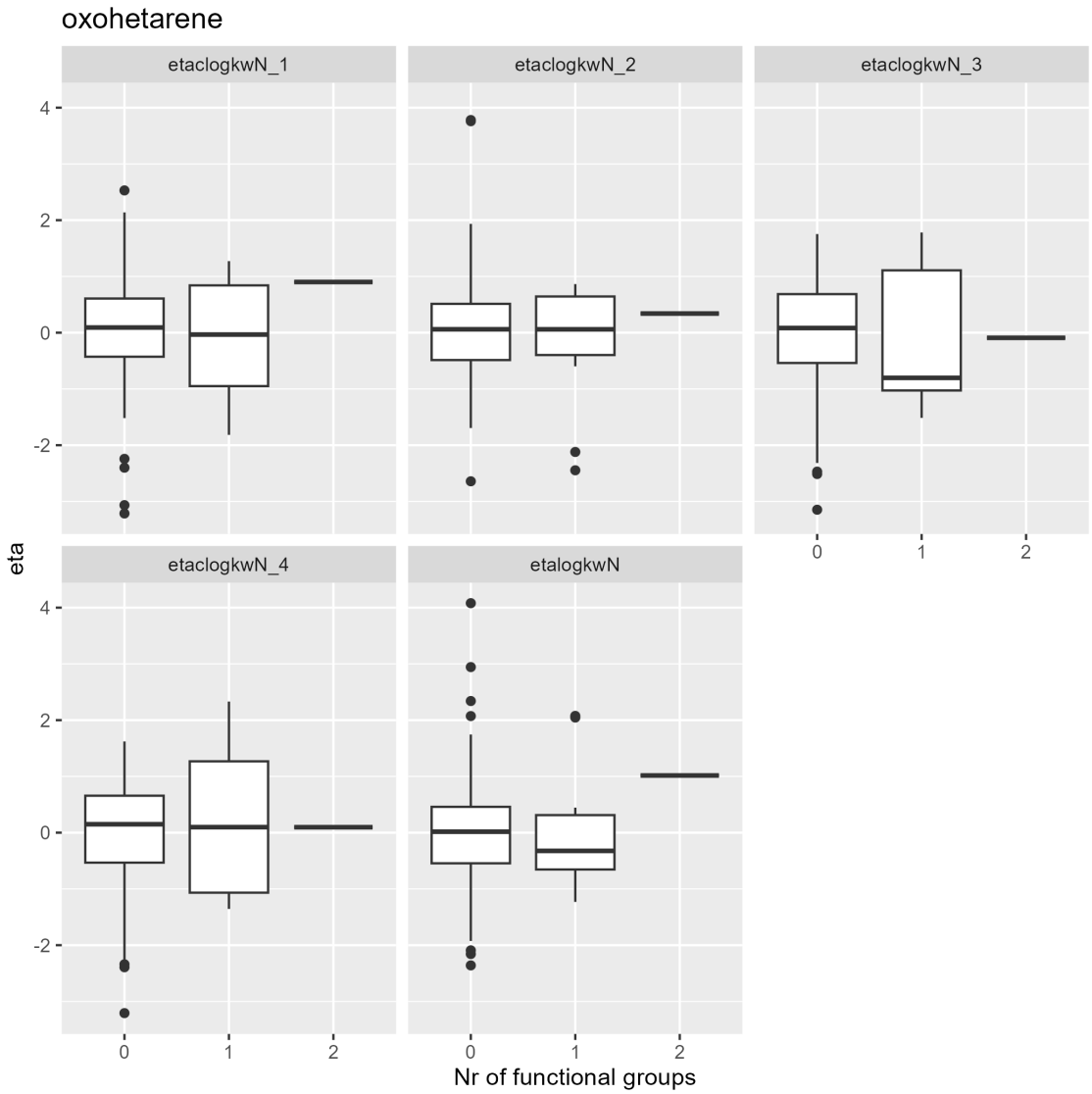

tert. mixed amine
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
Nr of functional groups

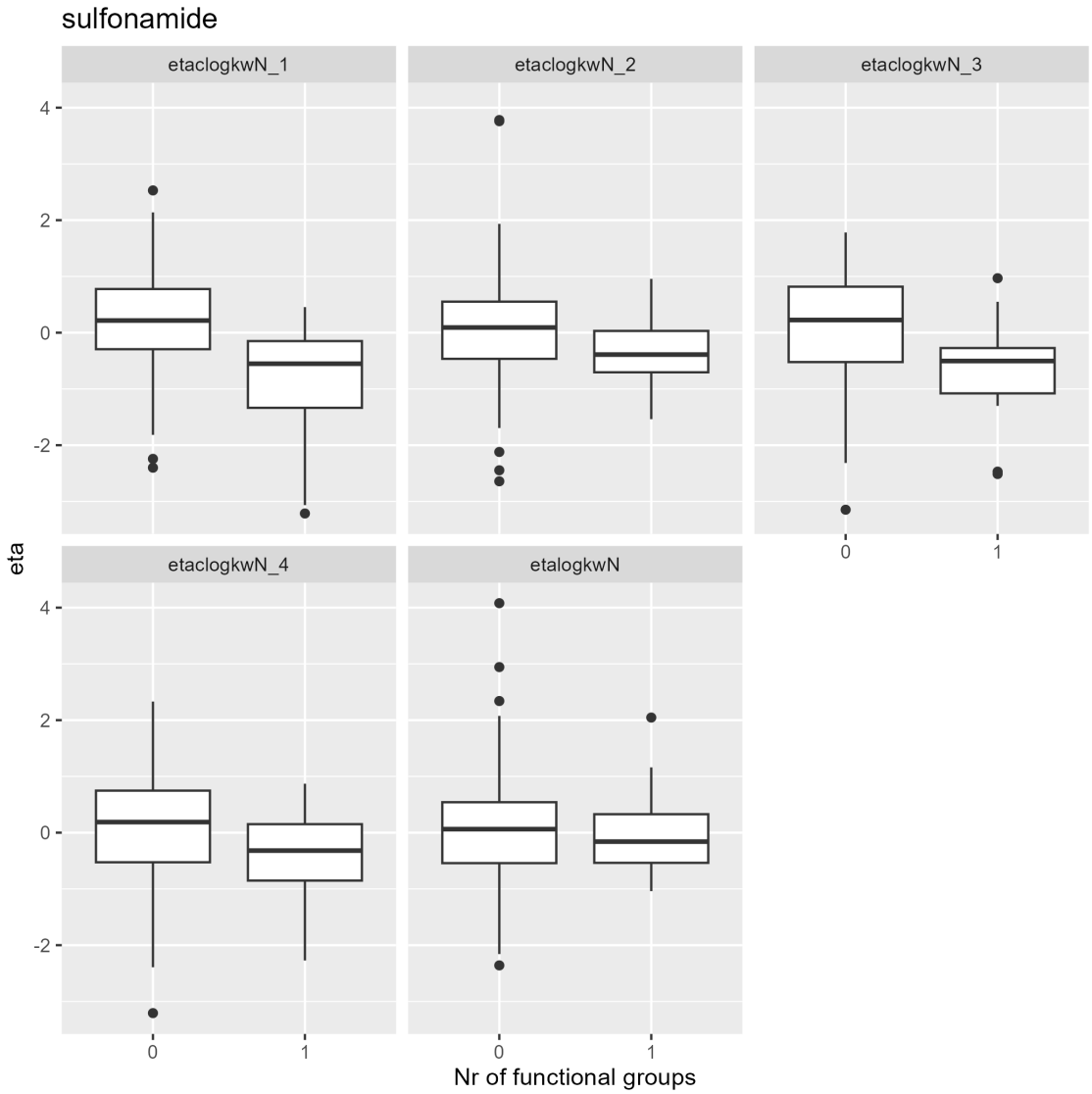

aryl chloride
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups


carboxylic acid
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
Nr of functional groups


carboxylic acid ester
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups


carboxylic acid sec. amide
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
Nr of functional groups

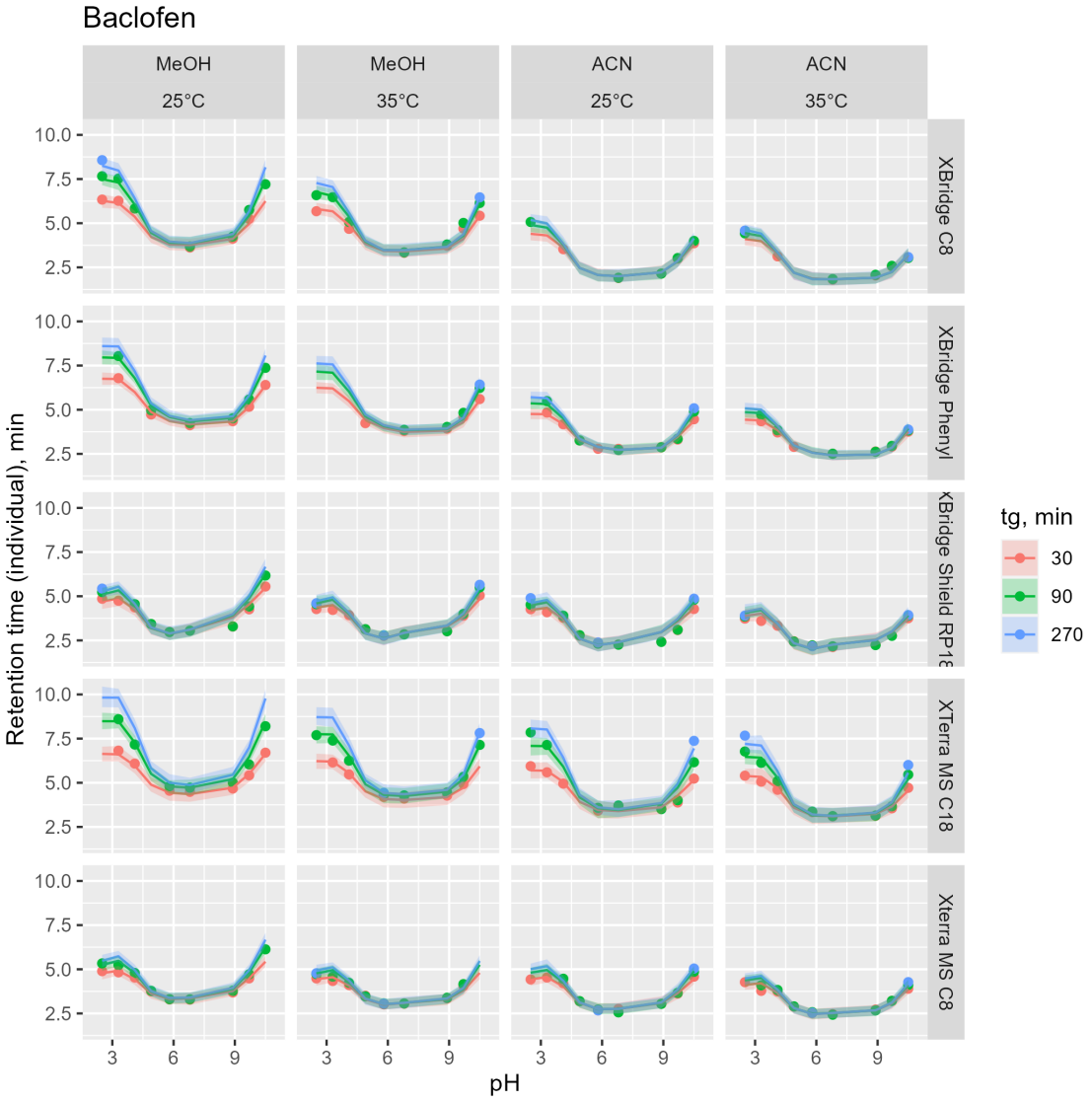

oxohetarene
etaclogkwN_1
etaclogkwN_2
etaclogkwN_3
etaclogkwN_4
etalogkwN
eta
4
2
0
-2
0
1
2
Nr of functional groups

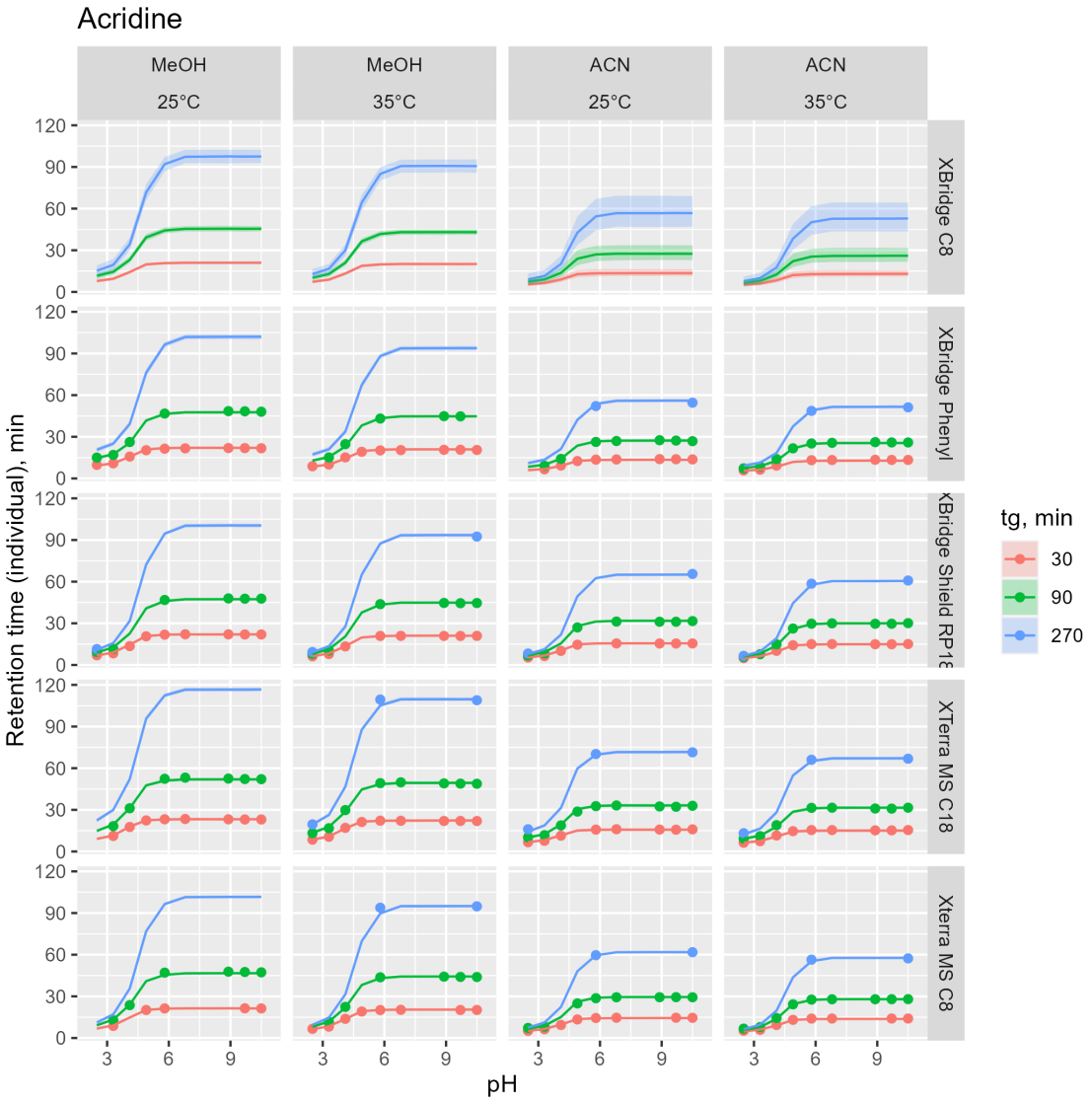


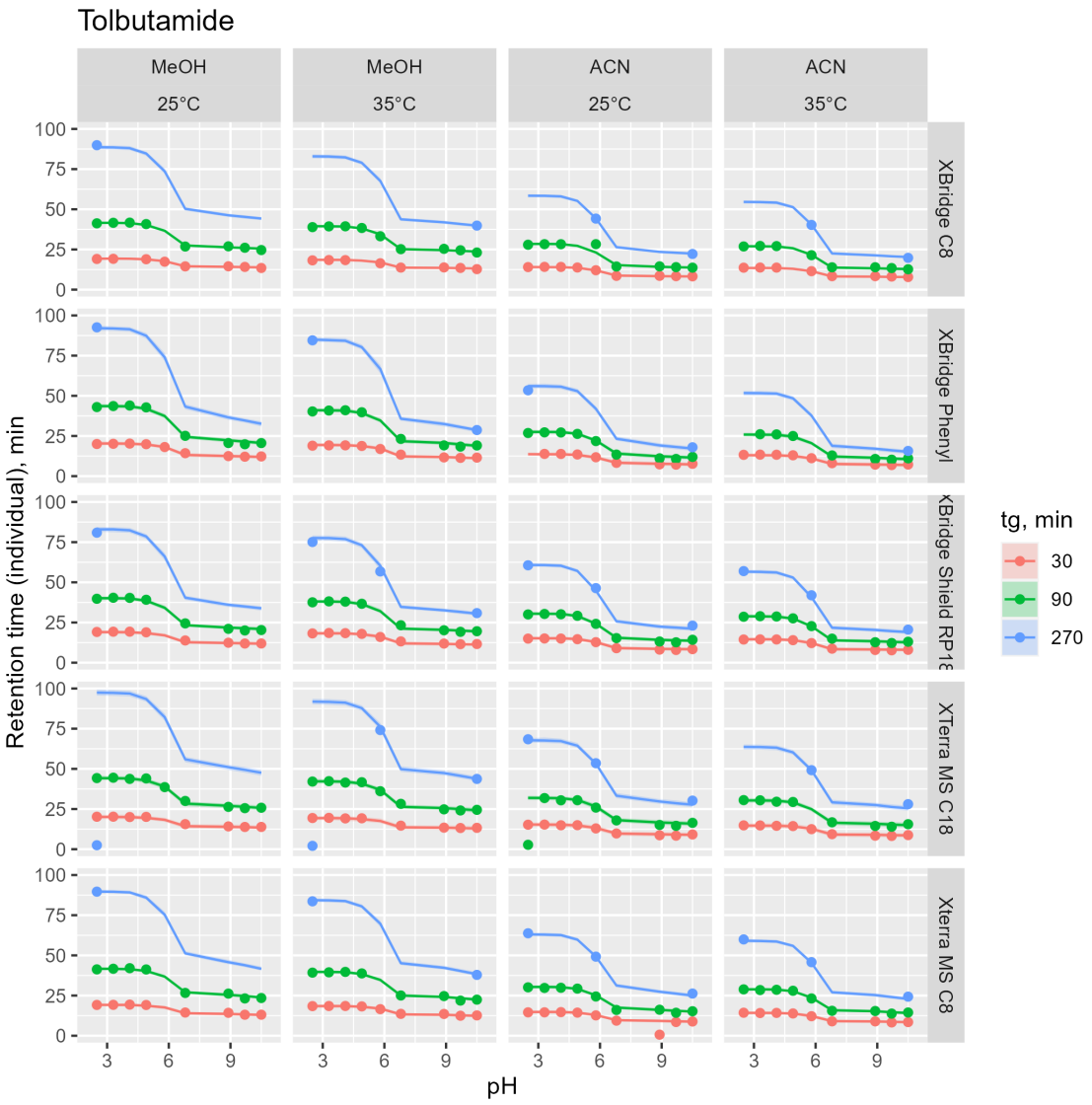
7 Comparison of observed and model predicted retention times

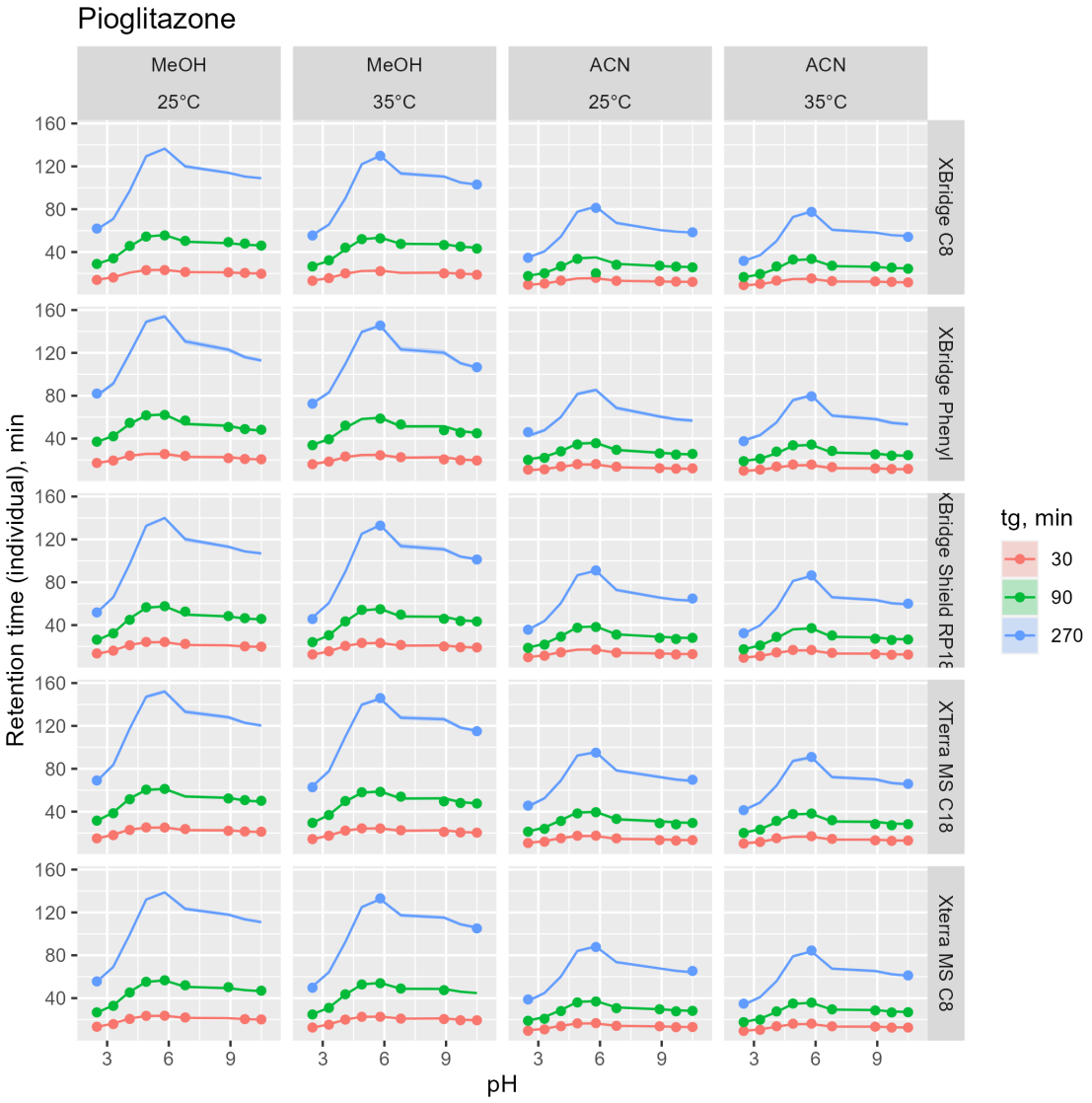
The individual and population model predictions were simulated using a more dense design. The individual predictions are based on population-level parameters, predictors, and all of the observed retention time measurements. The population predictions are based on population-level parameters and predictors.

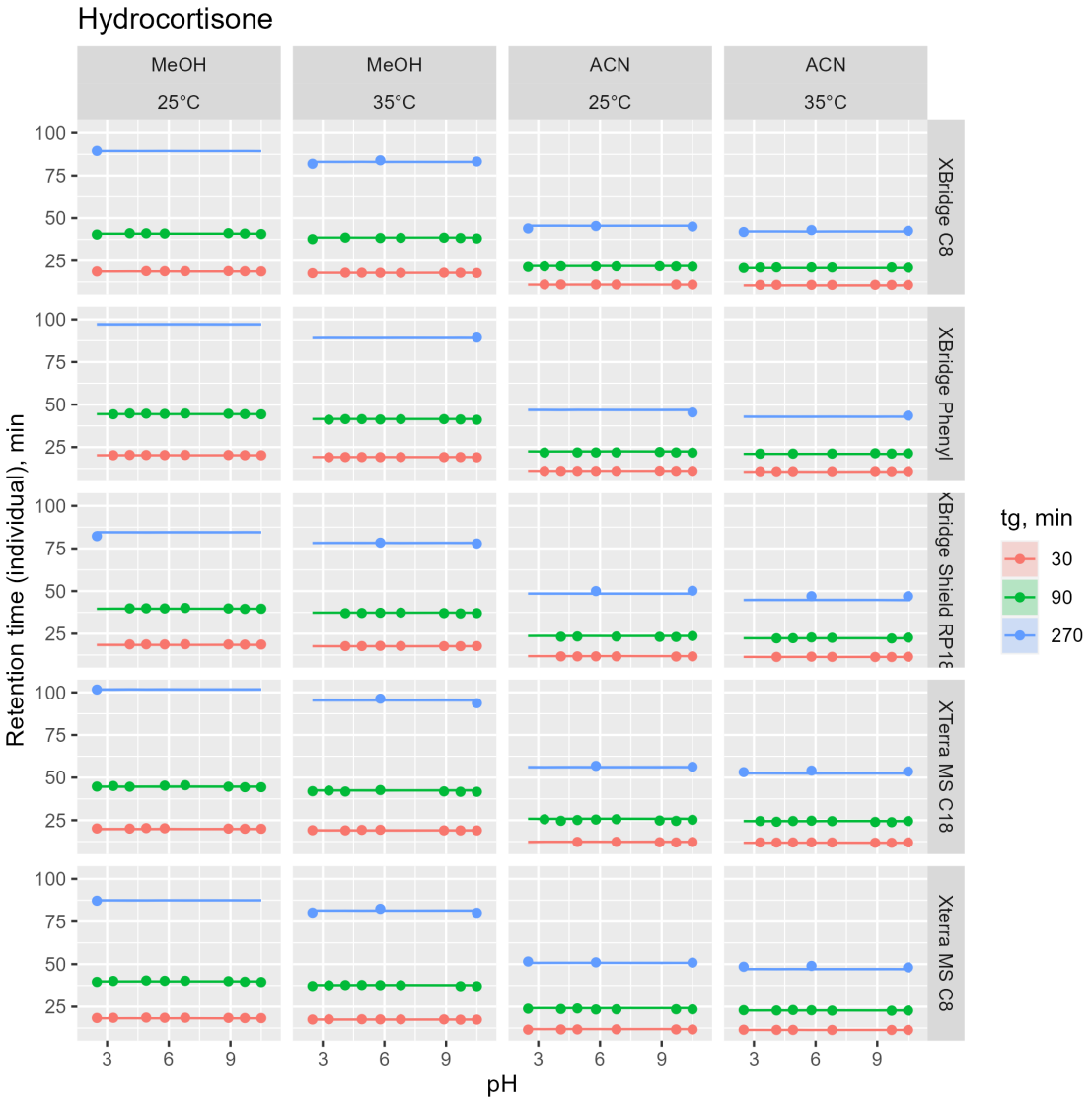
7.1 Individual predictions:

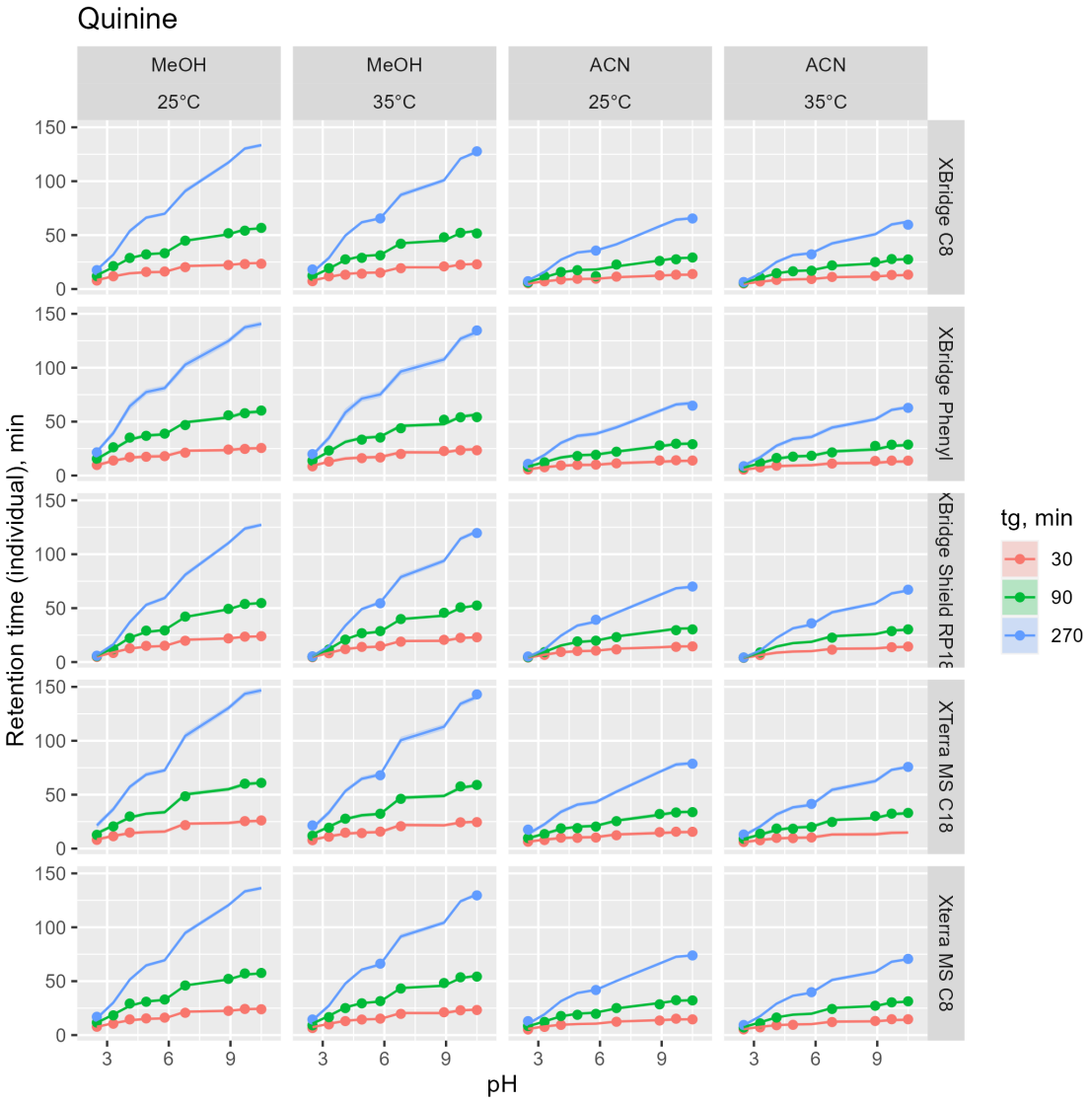



Acridine
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time (individual), min
pH
tg, min
30
90
270

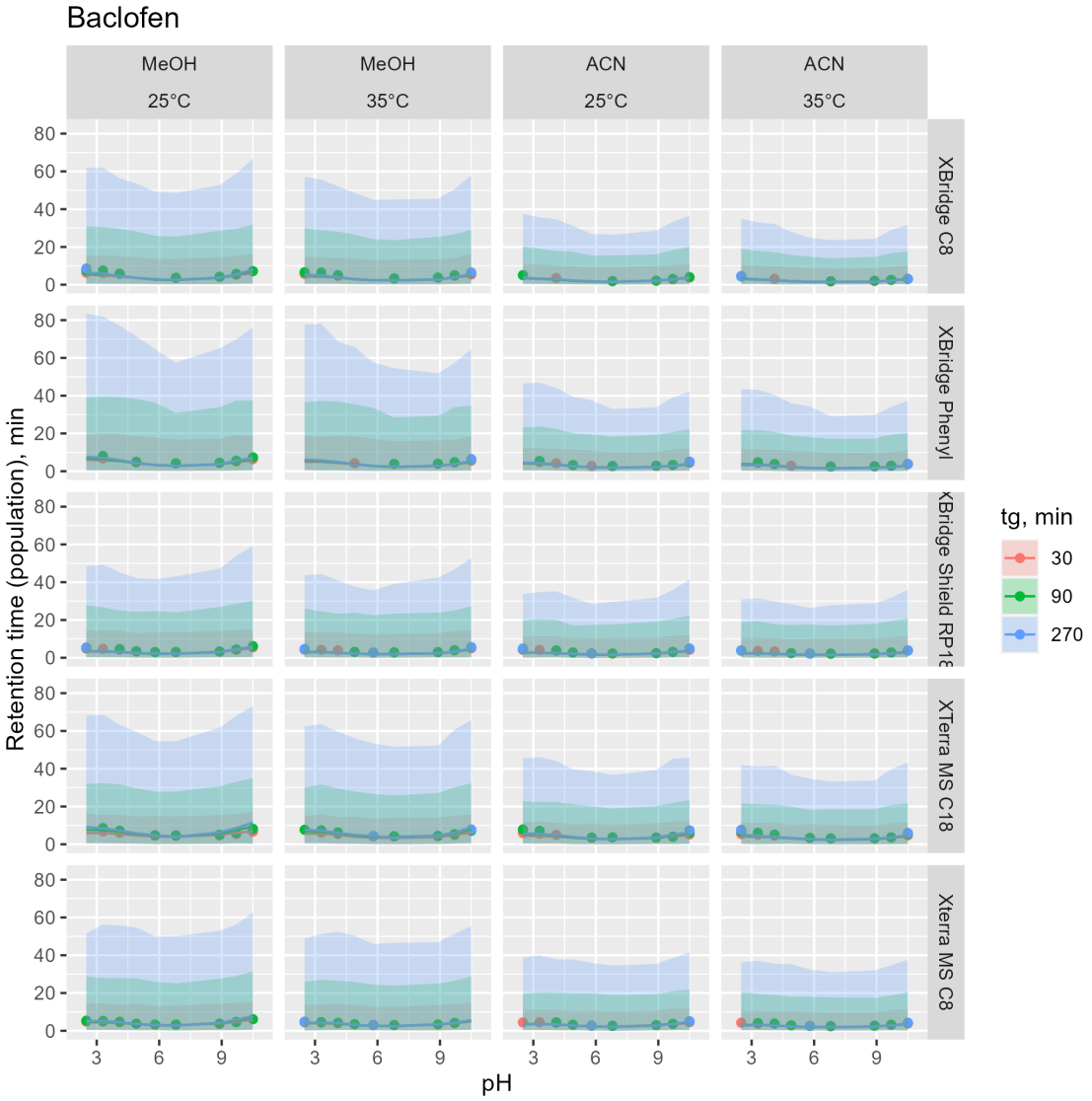

Tolbutamide
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time (individual), min
pH
tg, min
30
90
270

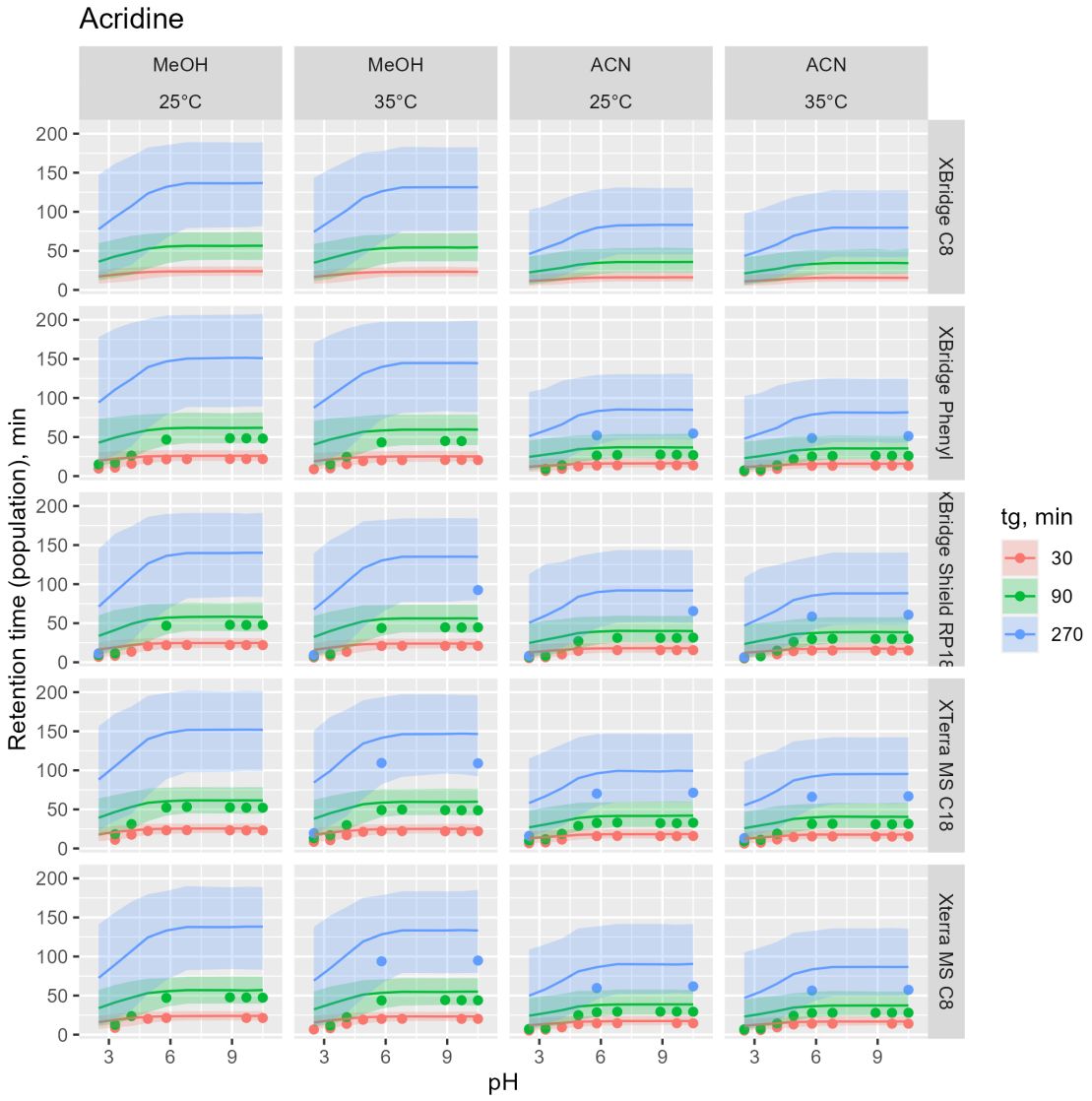

Pioglitazone
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time (individual), min
160
120
80
40
3
6
9
pH
tg, min
30
90
270

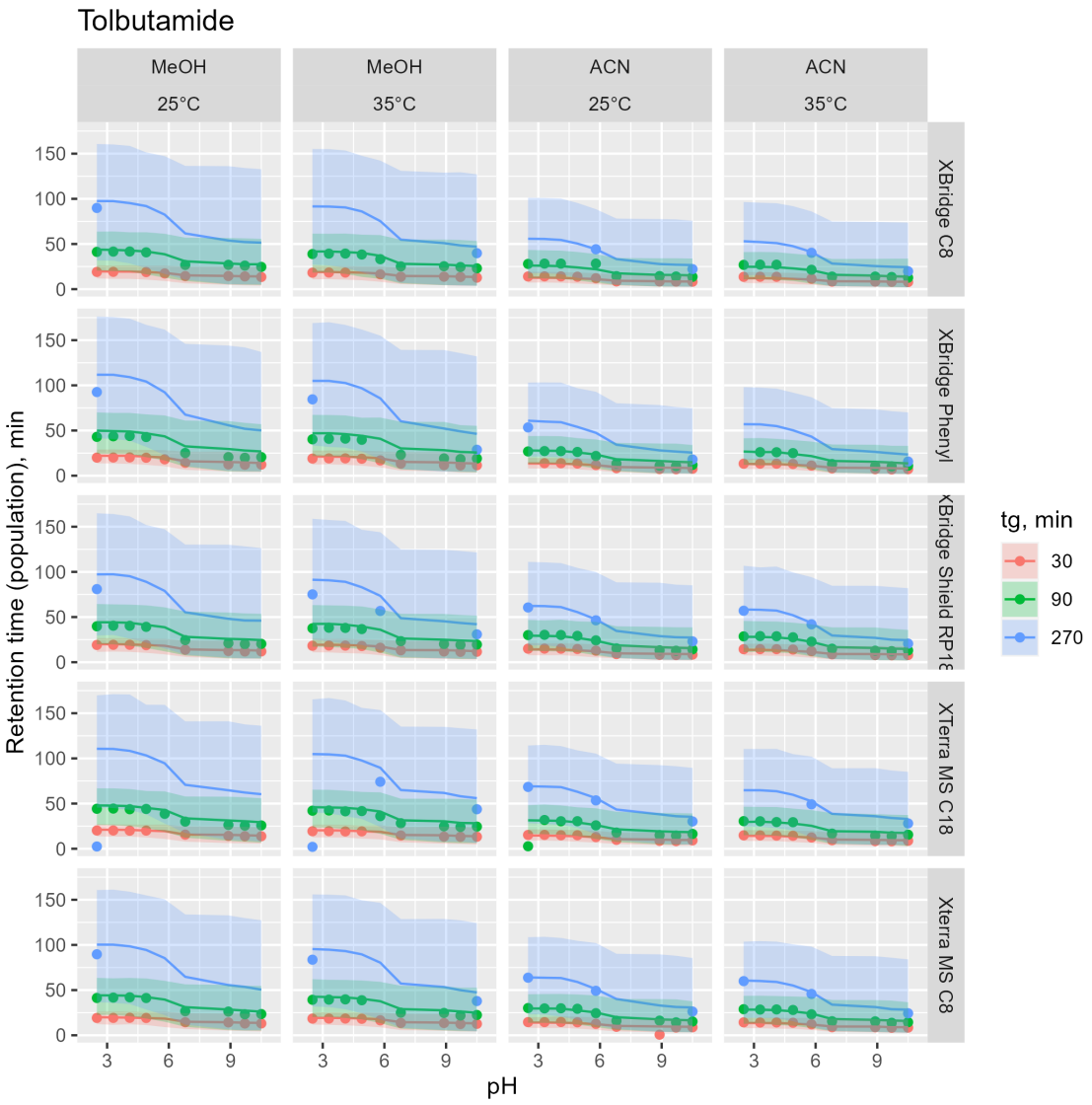

Hydrocortisone
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Retention time (individual), min
100
75
50
25
3
6
9
pH
tg, min
30
90
270

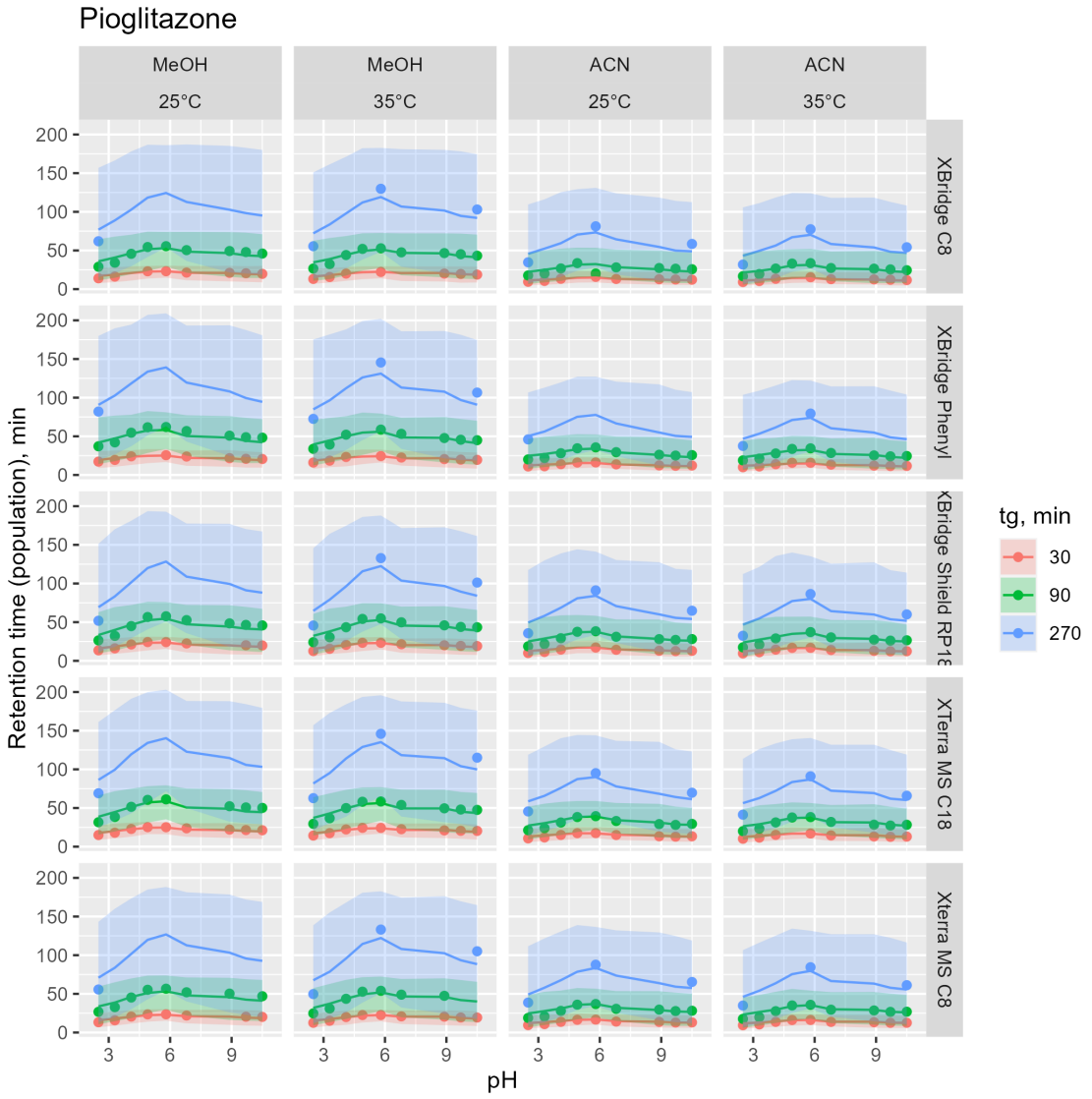

Quinine
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time (individual), min
pH
tg, min
30
90
270

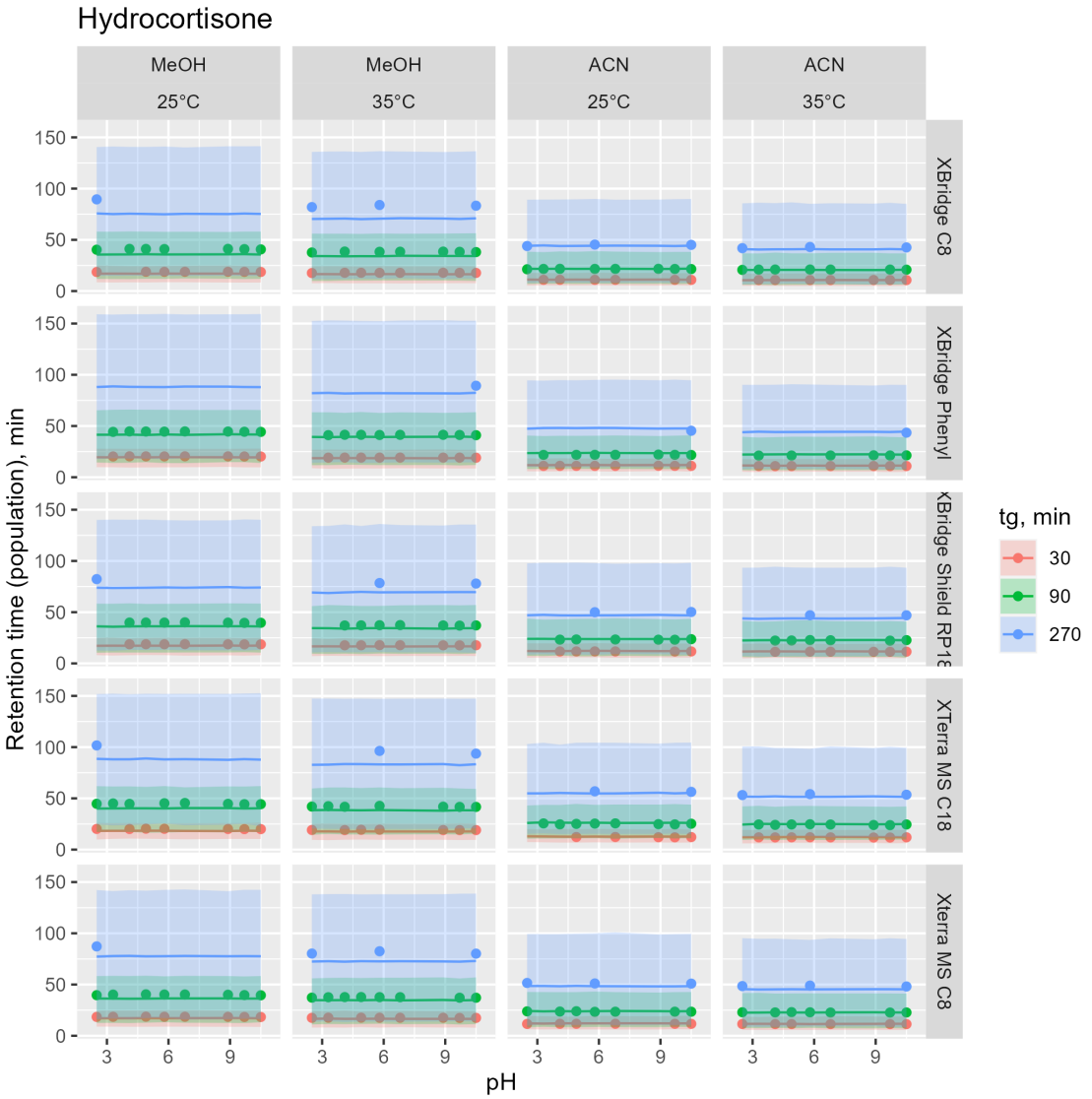
7.2 Population predictions:

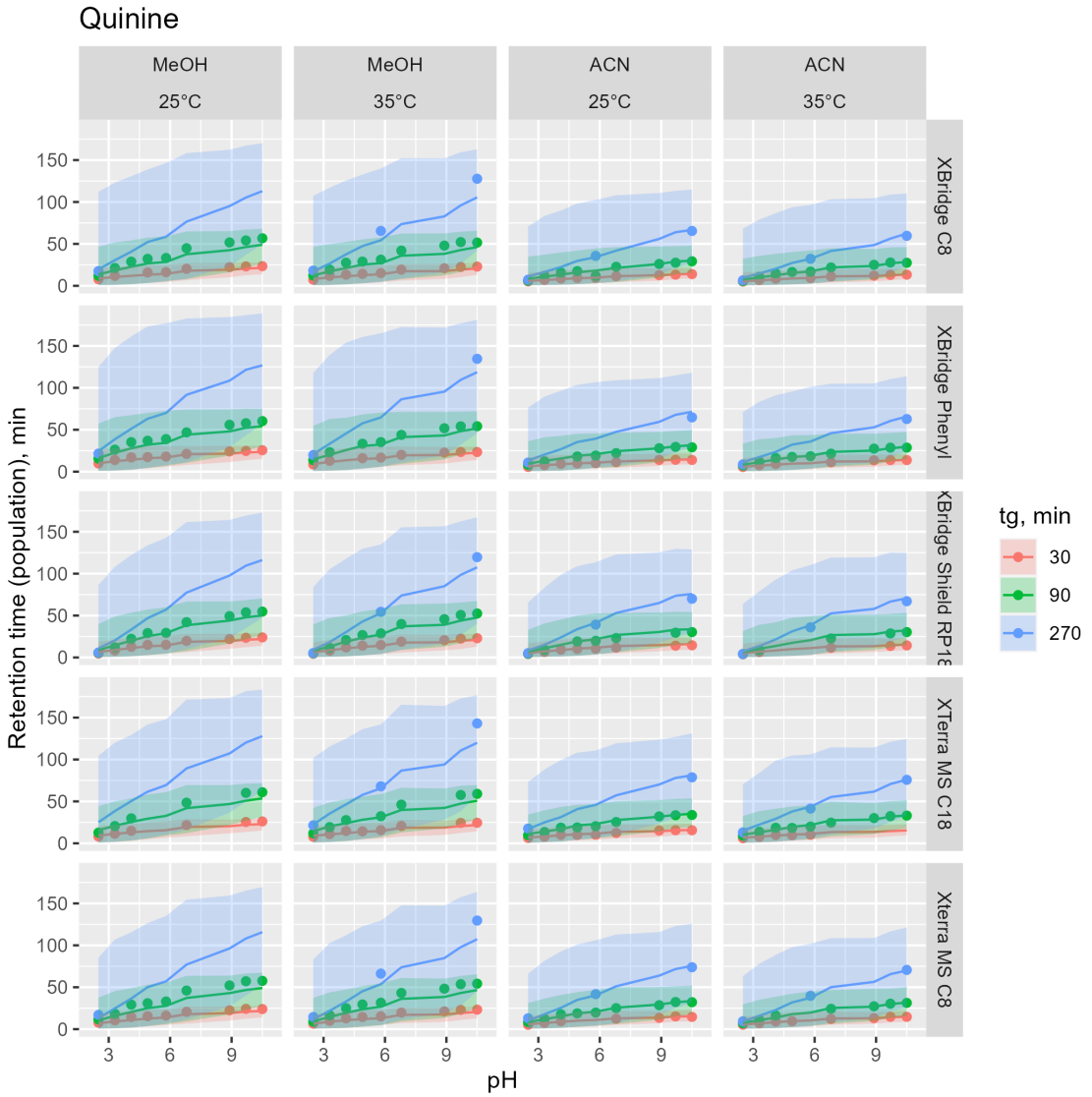



Acridine
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Retention time (population), min
pH
tg, min
30
90
270


Tolbutamide
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time (population), min
150
100
50
0
3
6
9
pH
tg, min
30
90
270

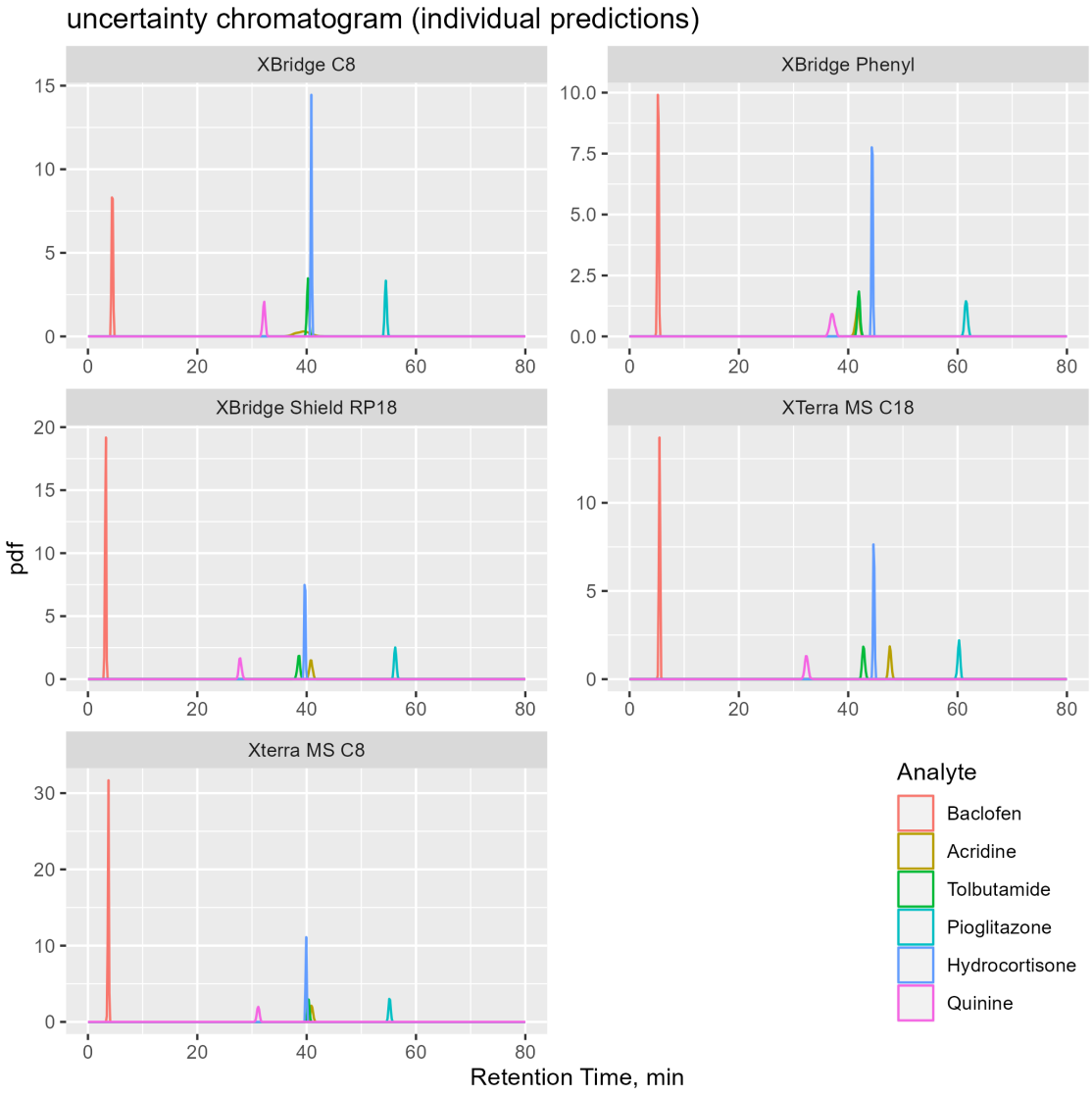

Pioglitazone
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time (population), min
pH
tg, min
30
90
270

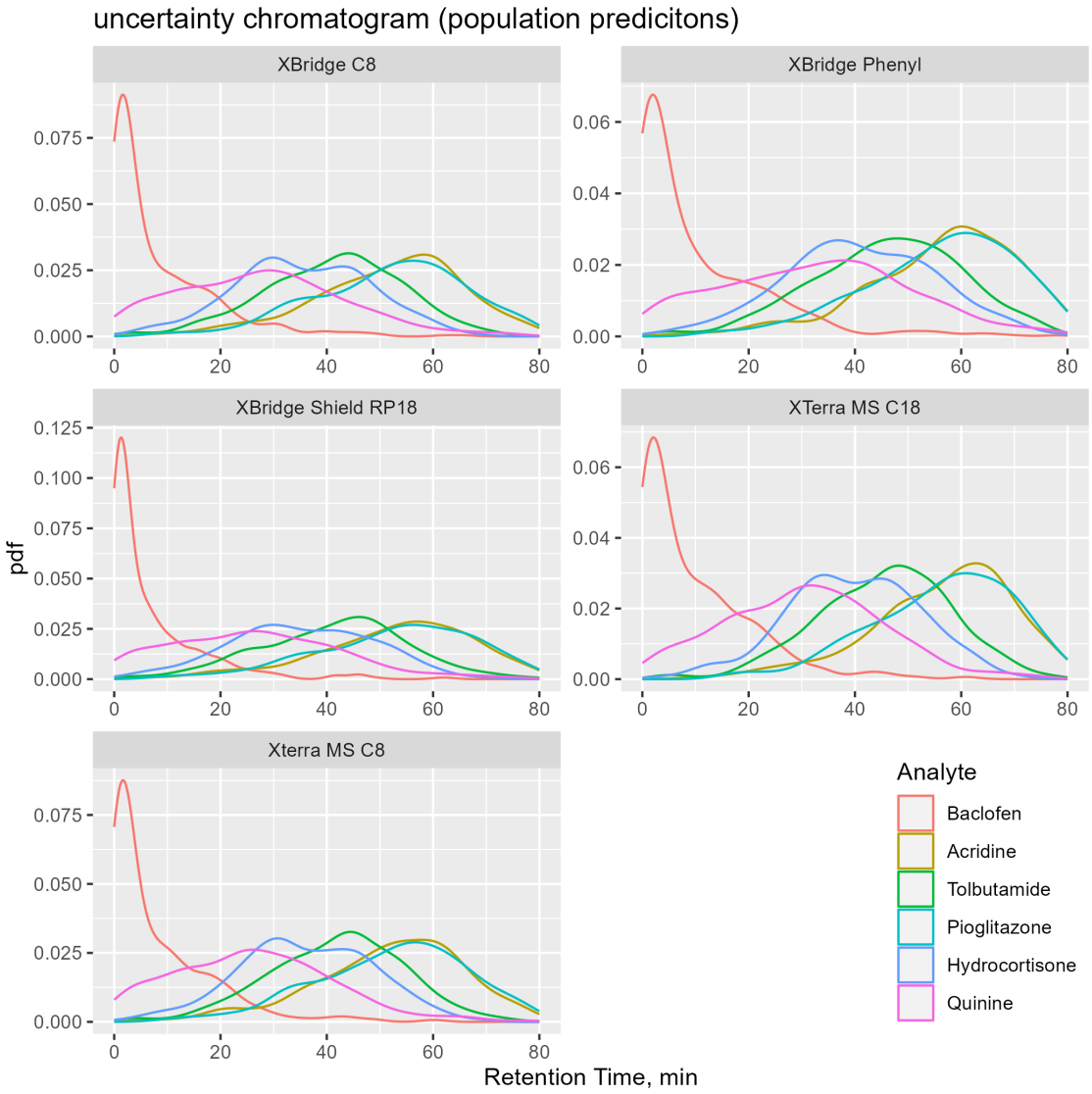

Hydrocortisone
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Retention time (population), min
150
100
50
0
3
6
9
pH
tg, min
30
90
270



8 Uncertainity chromatogram

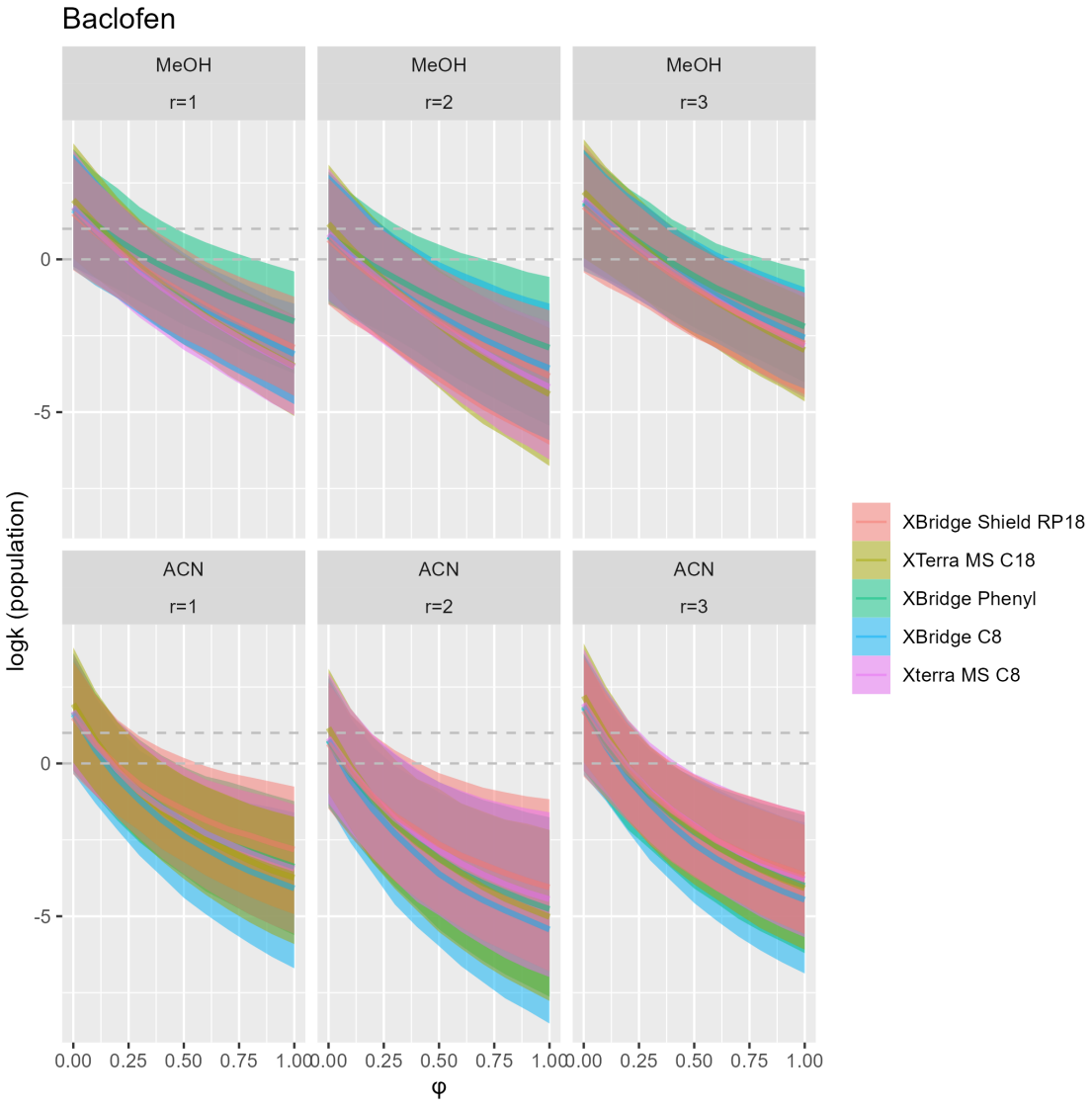
Below is the example of an uncertainty chromatogram expected for tg = 90 min, pH = 4.9, Temp = 25^oC in MeOH. The individual and population predictions are shown.

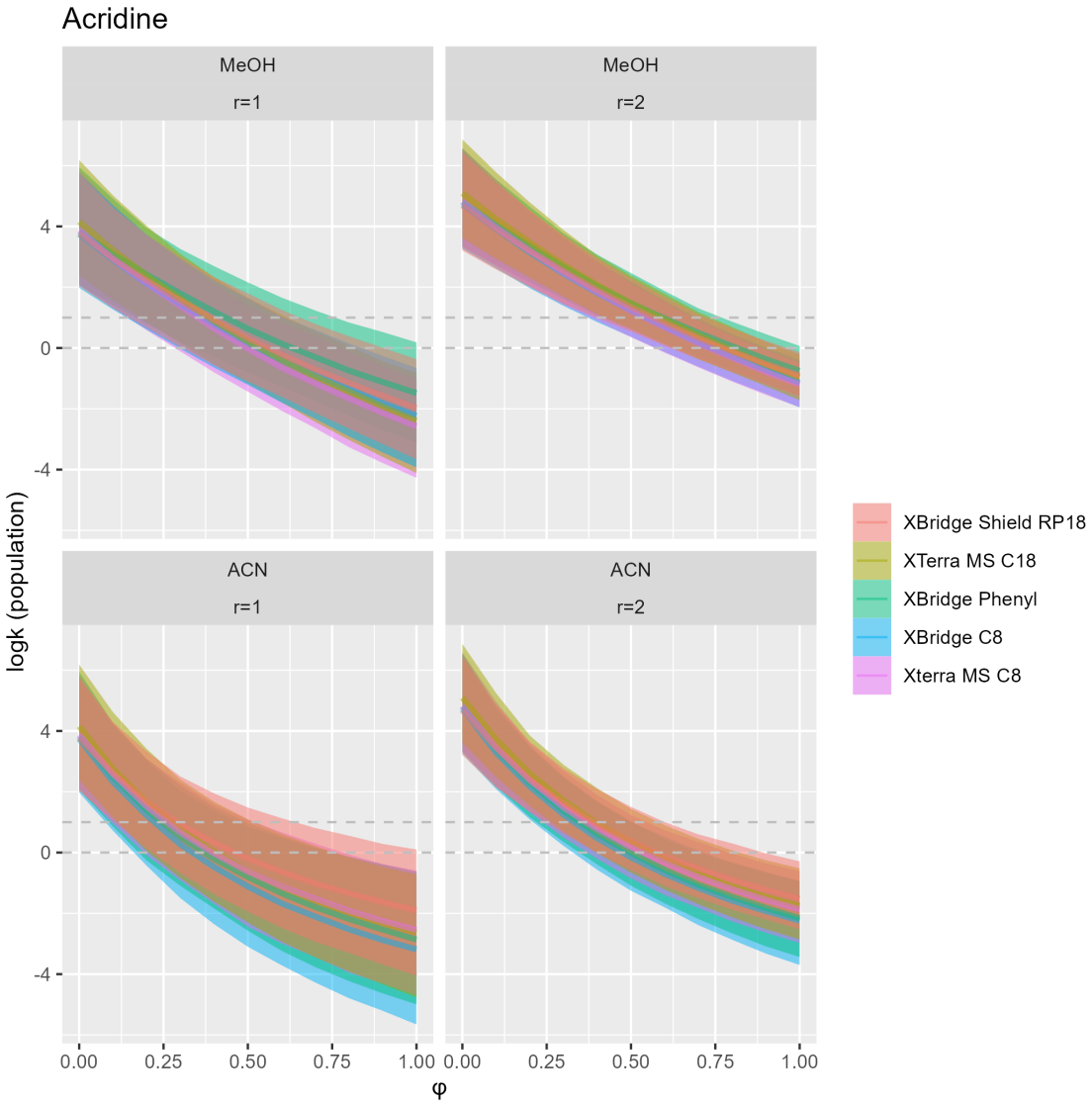

uncertainty chromatogram (individual predictions)
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
pdf
Retention Time, min
Analyte
Baclofen
Acridine
Tolbutamide
Pioglitazone
Hydrocortisone
Quinine

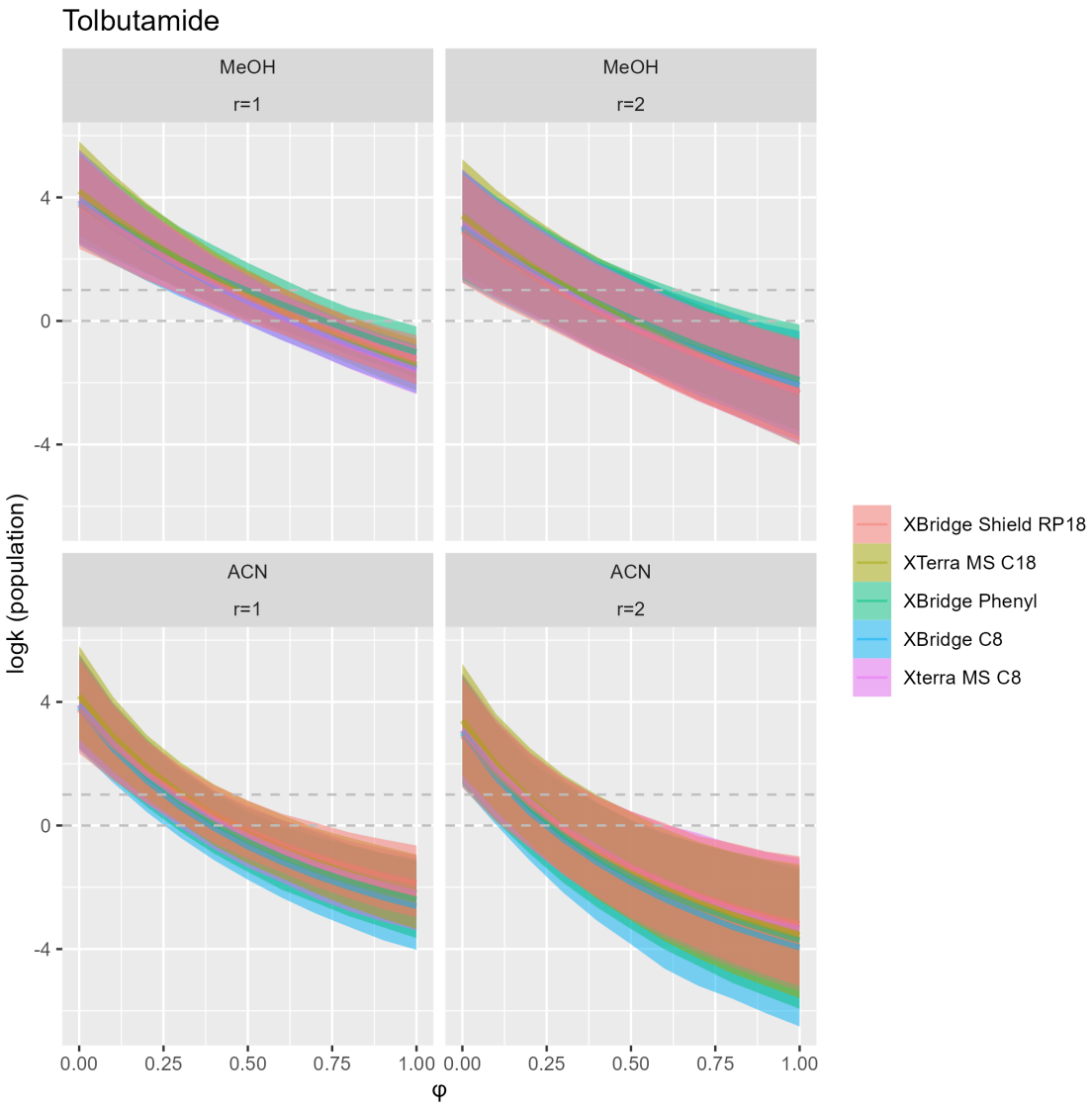


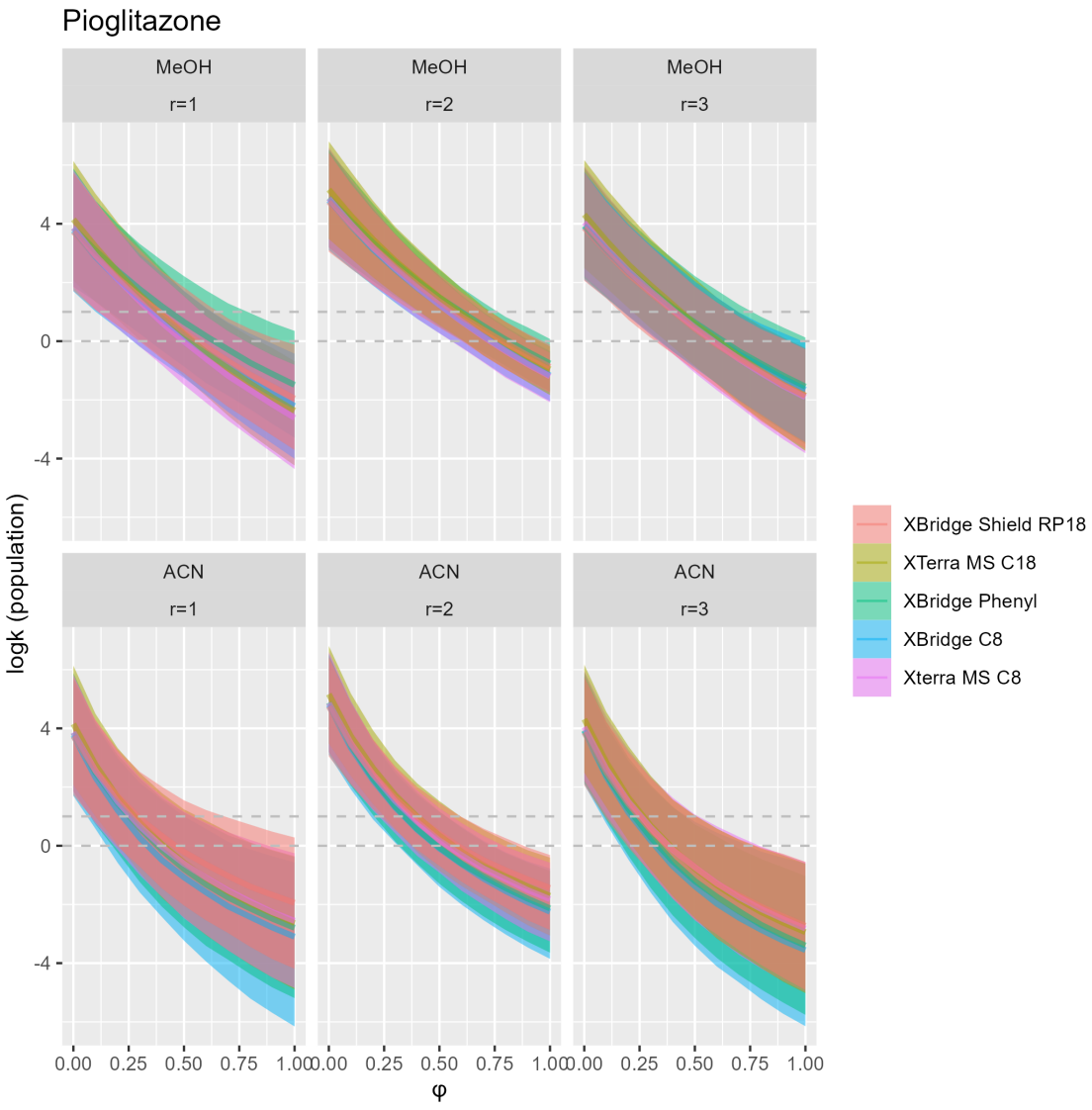
8.1 Isocratic predictions (population predictions)

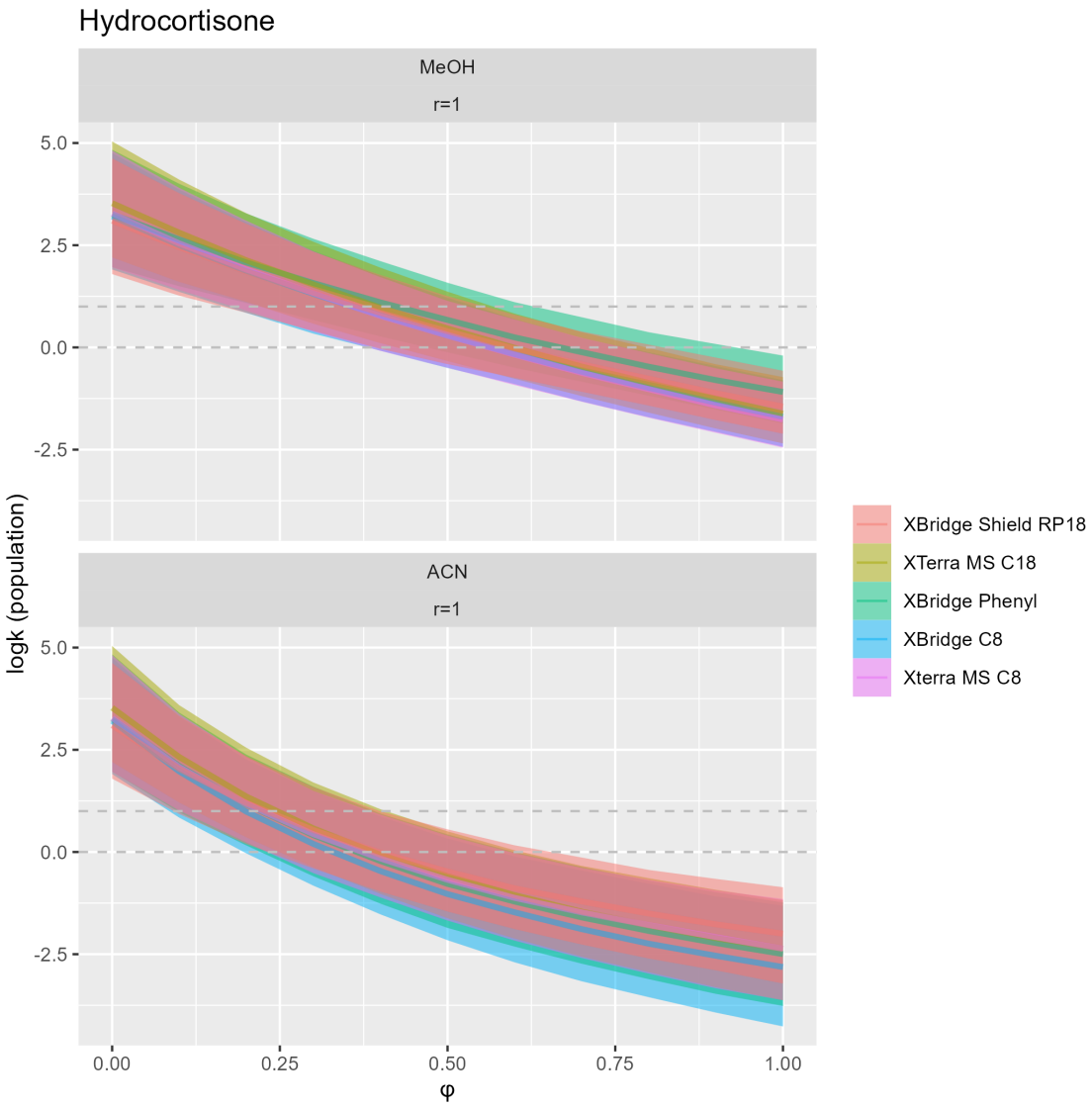
To better asses the impact of parameters on retention we created graphs presenting the isocratic logarithm of retention factor vs. φ for selected analytes. Separate graphs are shown for each dissociation form (r=1, r=2, r=3). Here the population predictions are shown:

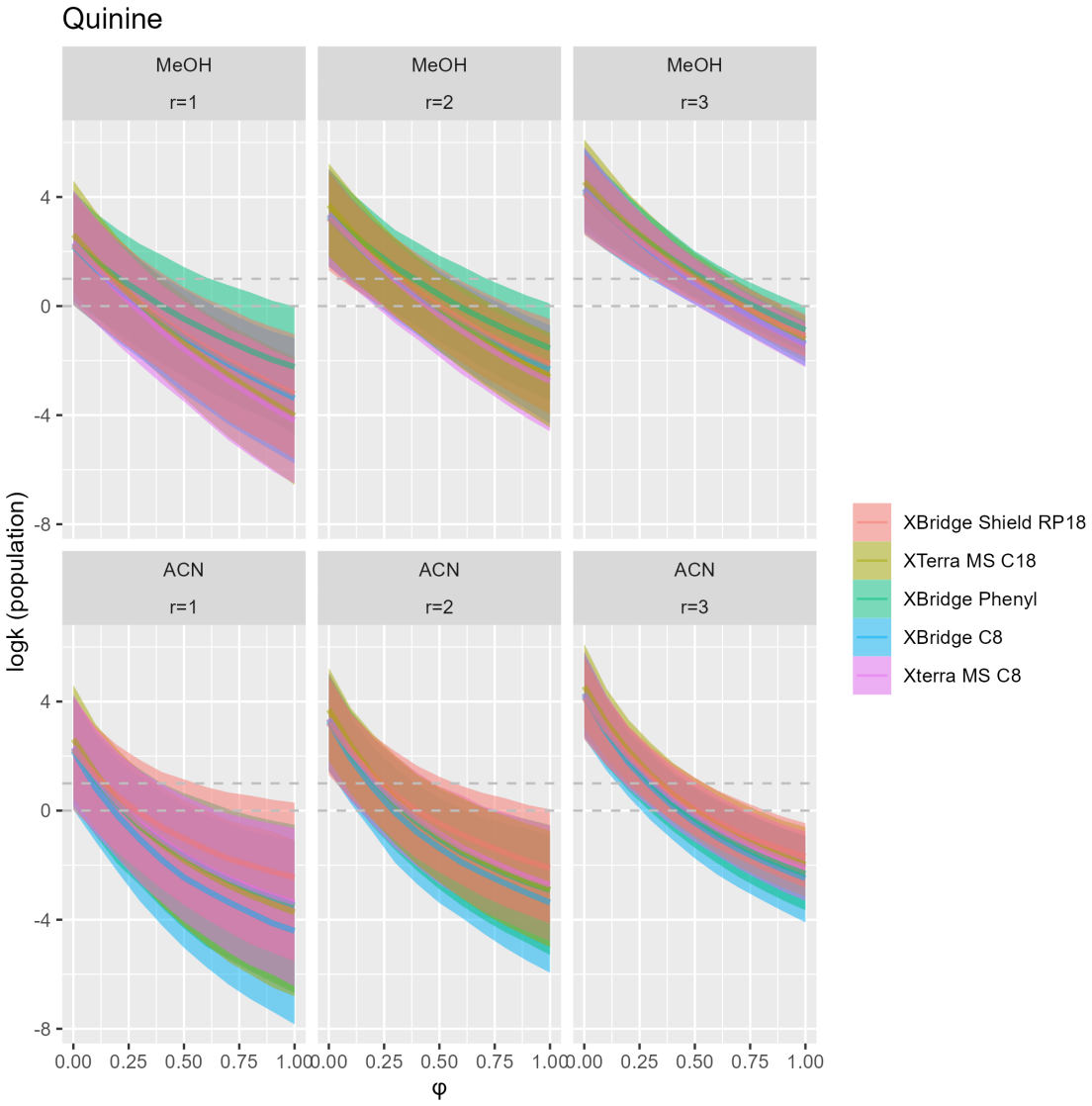

Baclofen
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
logk (population)
0
-5
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Acridine
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
logk (population)
4
0
-4
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

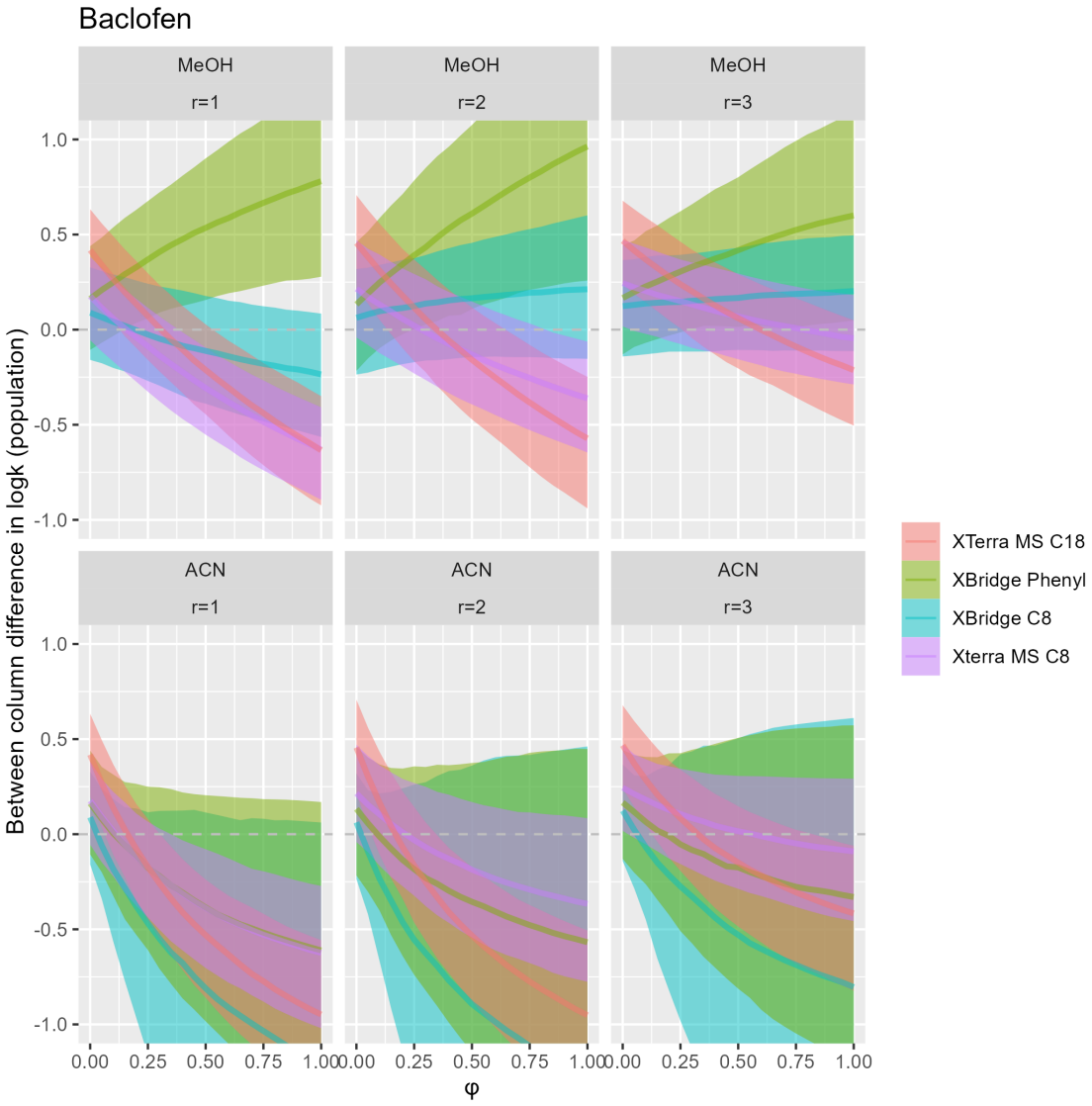

Tolbutamide
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
logk (population)
4
0
-4
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

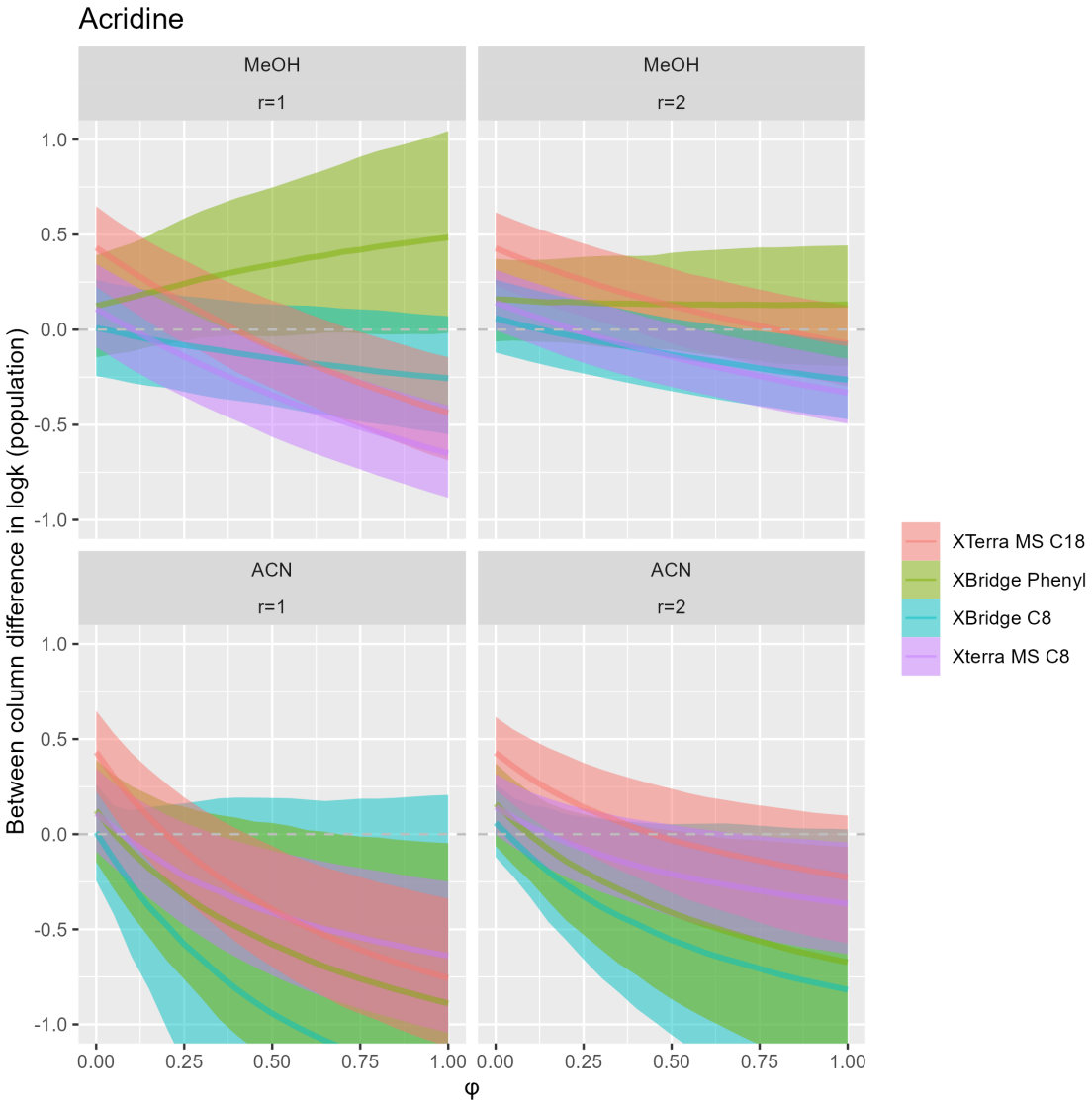

Pioglitazone
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
logk (population)
4
0
-4
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

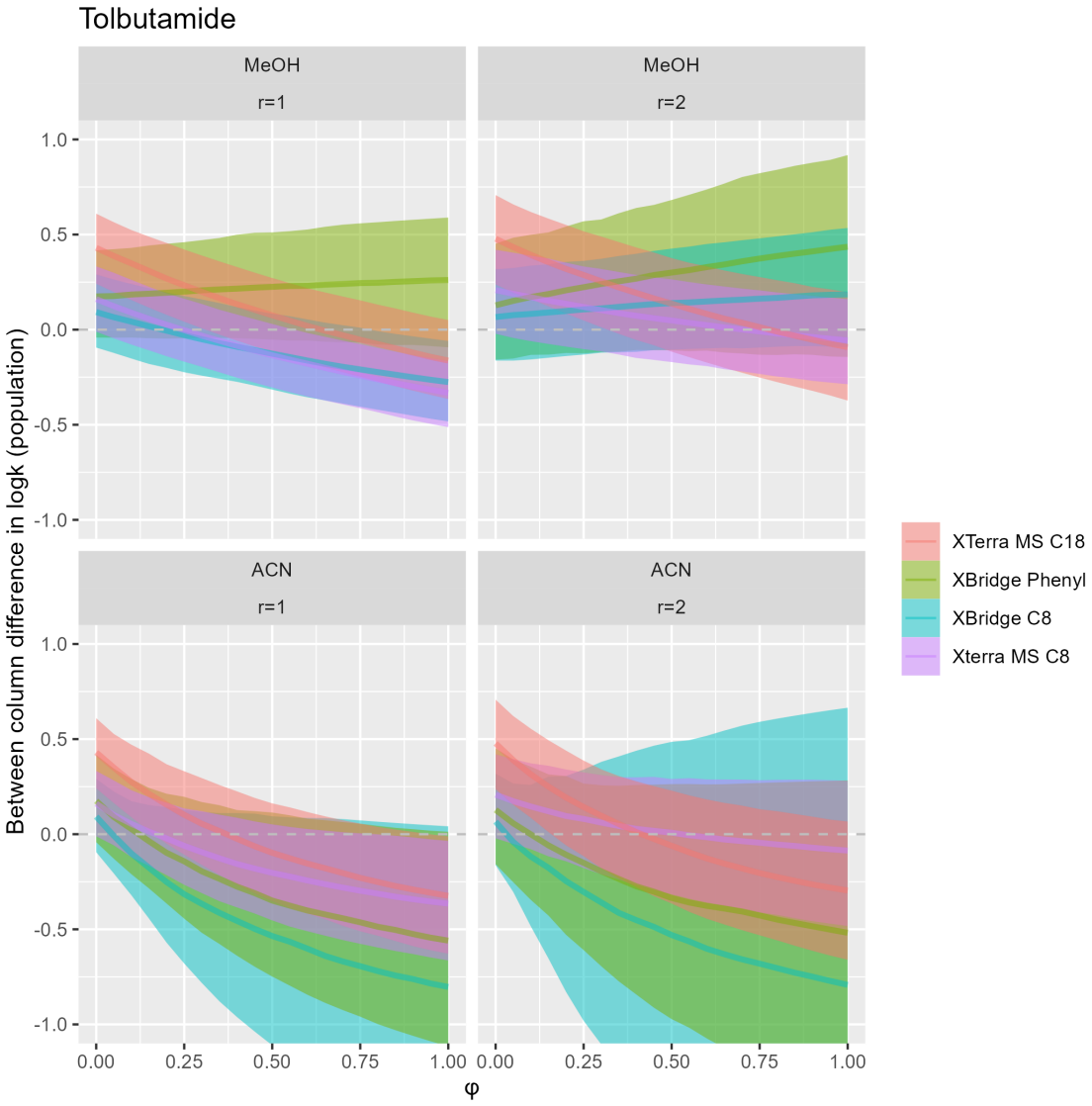

Hydrocortisone
MeOH
r=1
ACN
r=1
logk (population)
5.0
2.5
0.0
-2.5
0.00
0.25
0.50
0.75
1.00
φ
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

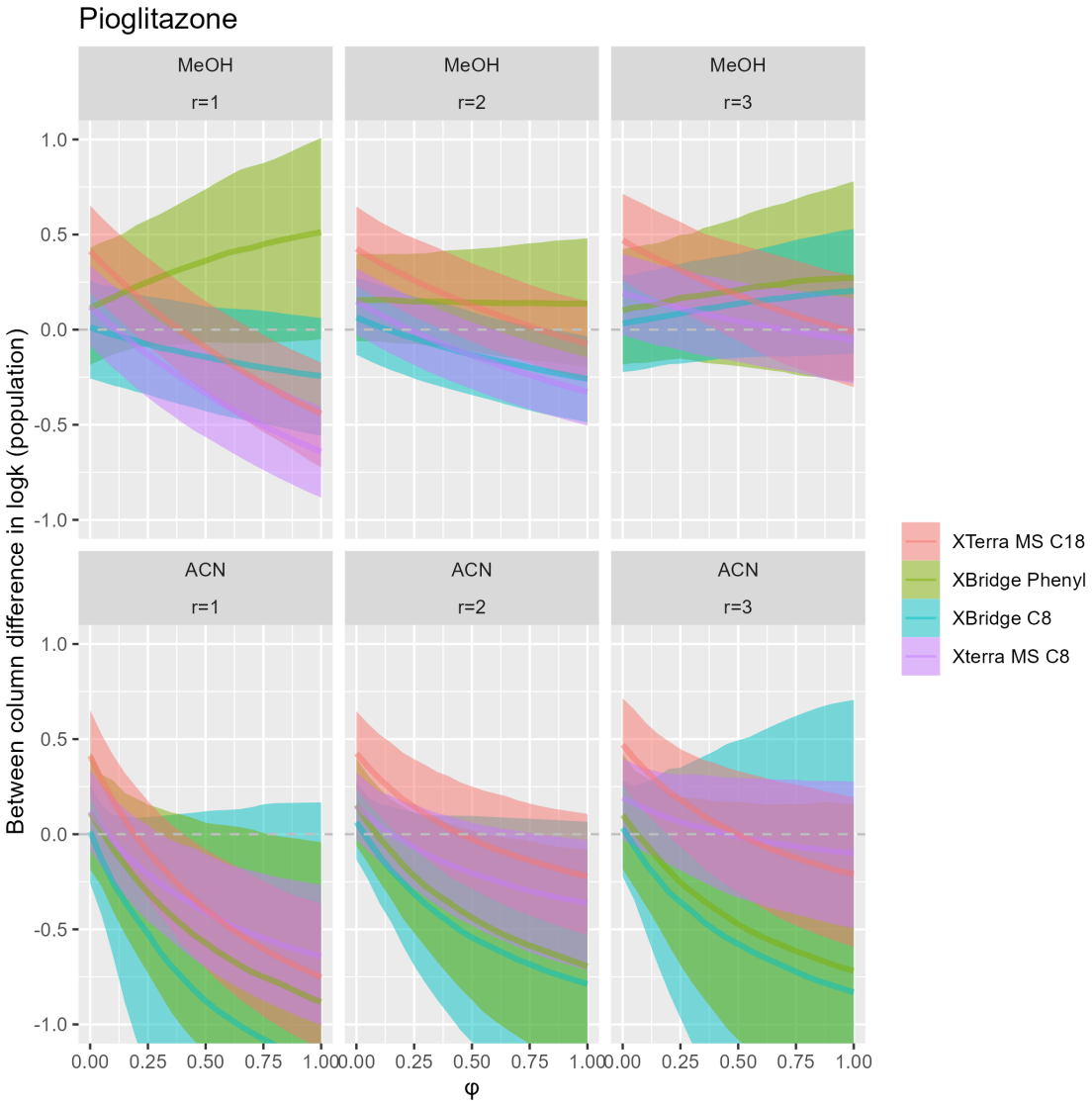


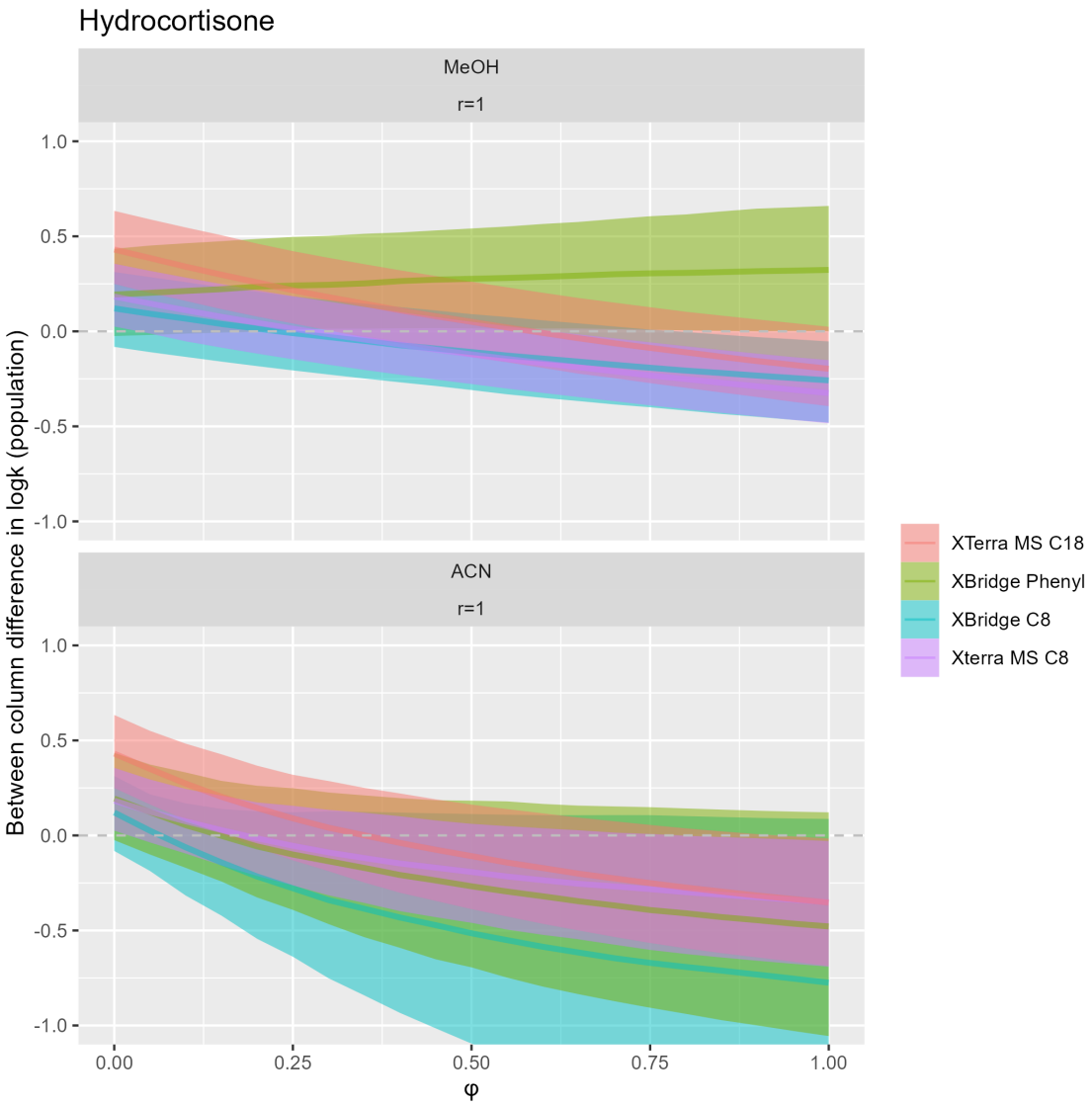
Similarly we can quantify the column effects (between column differences in logk using XBridge Shield RP18 as a reference column):

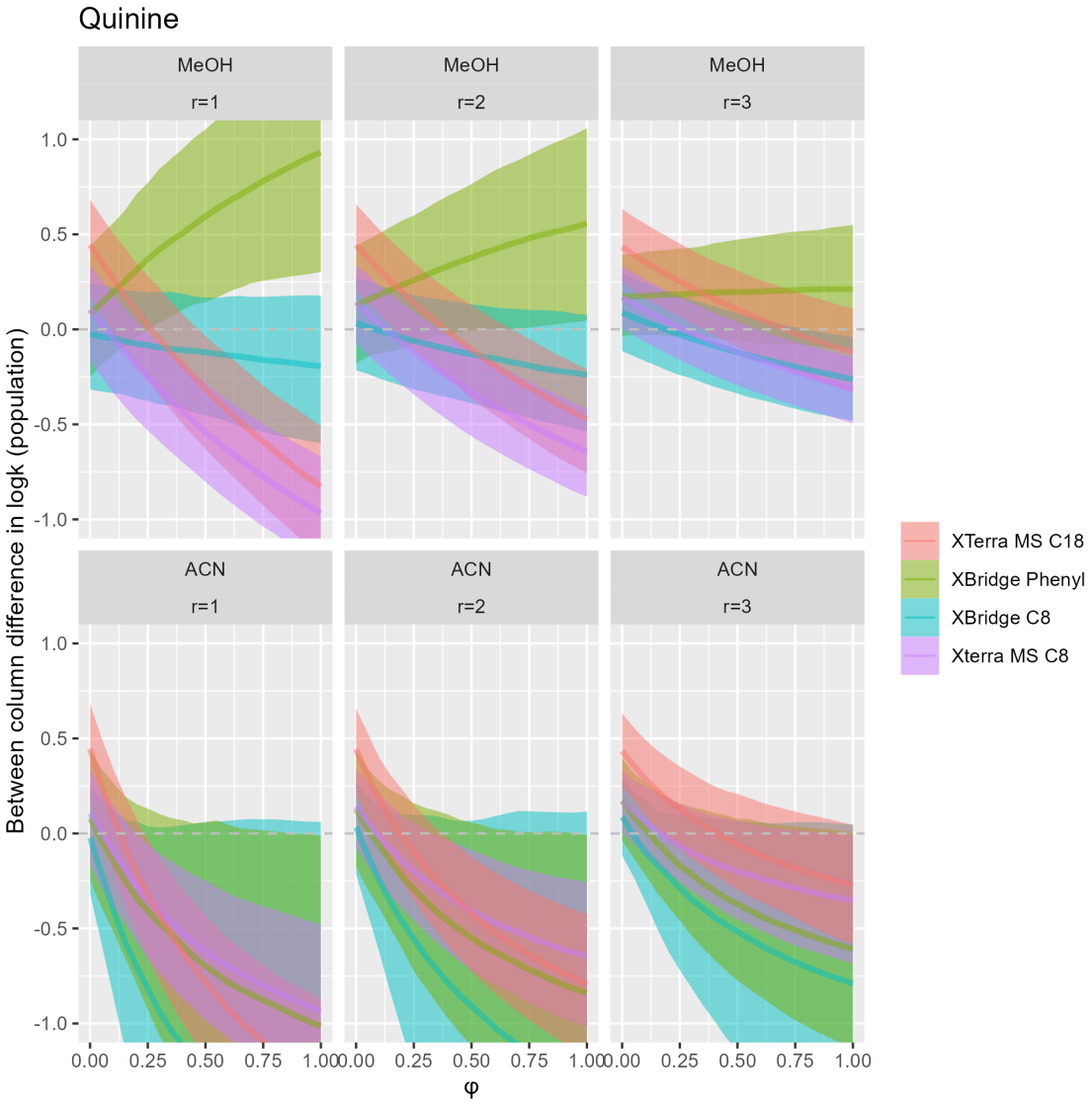

Baclofen
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
Between column difference in logk (population)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Acridine
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
Between column difference in logk (population)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

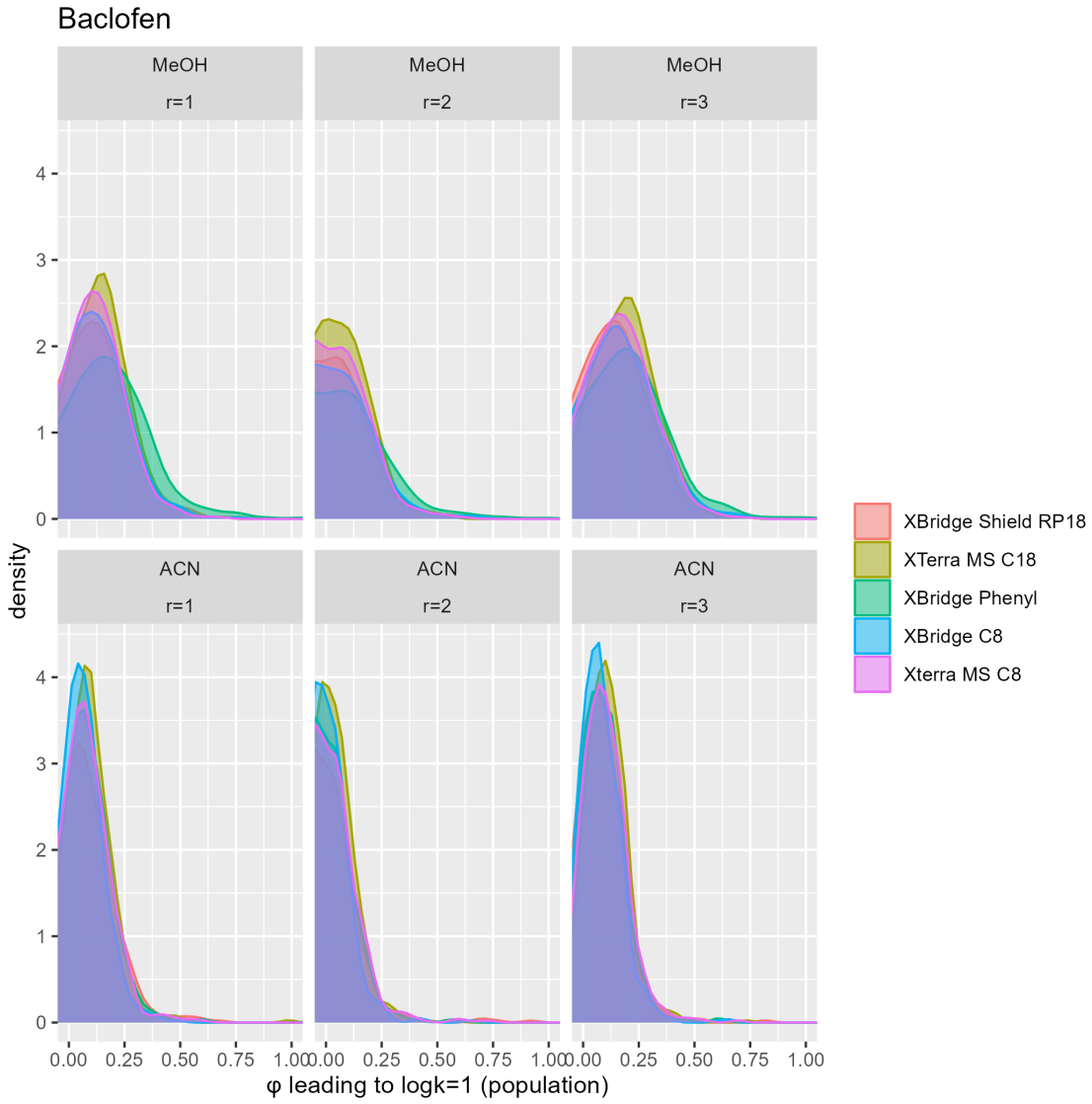

Tolbutamide
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
Between column difference in logk (population)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

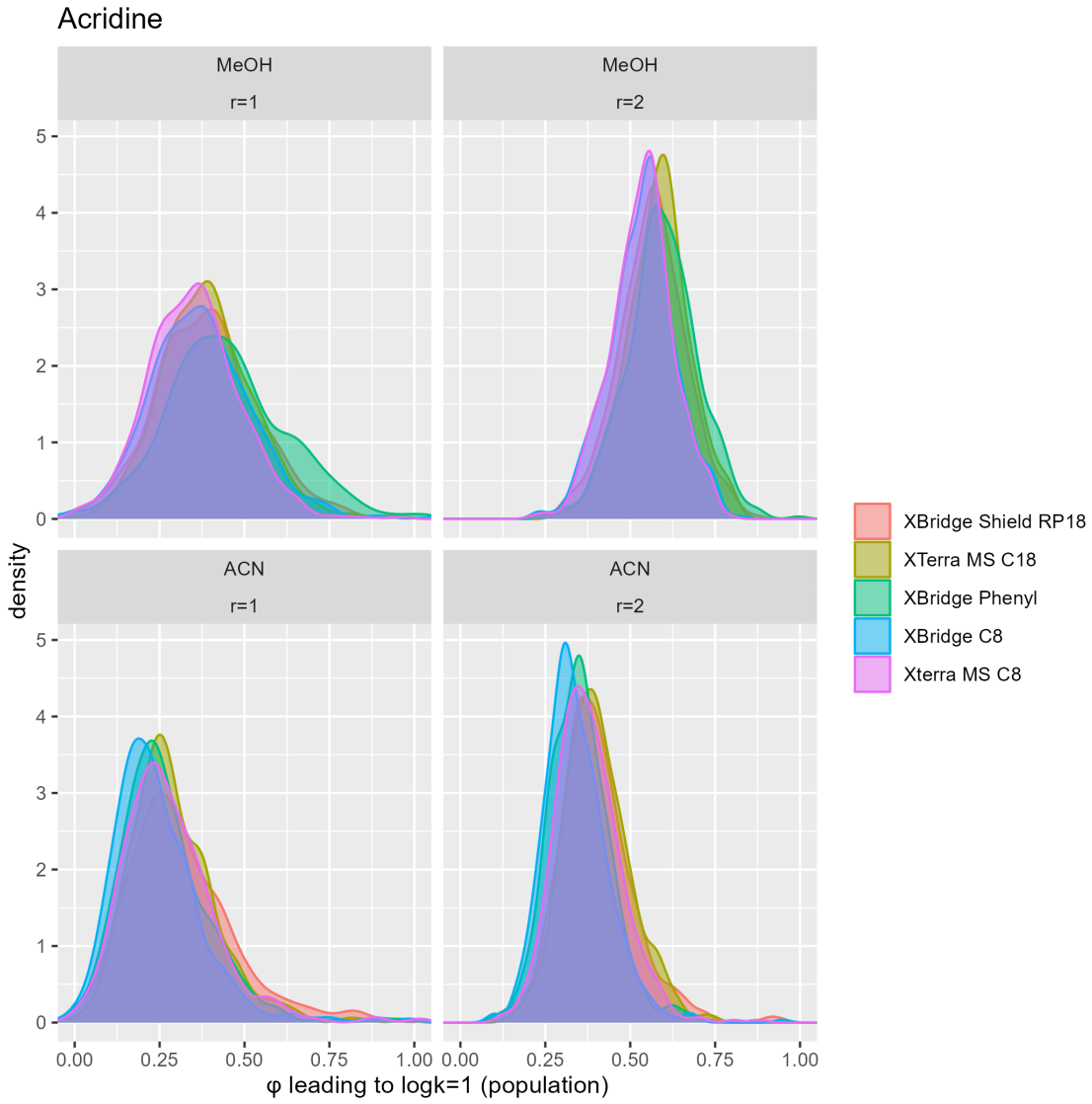

Pioglitazone
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
Between column difference in logk (population)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

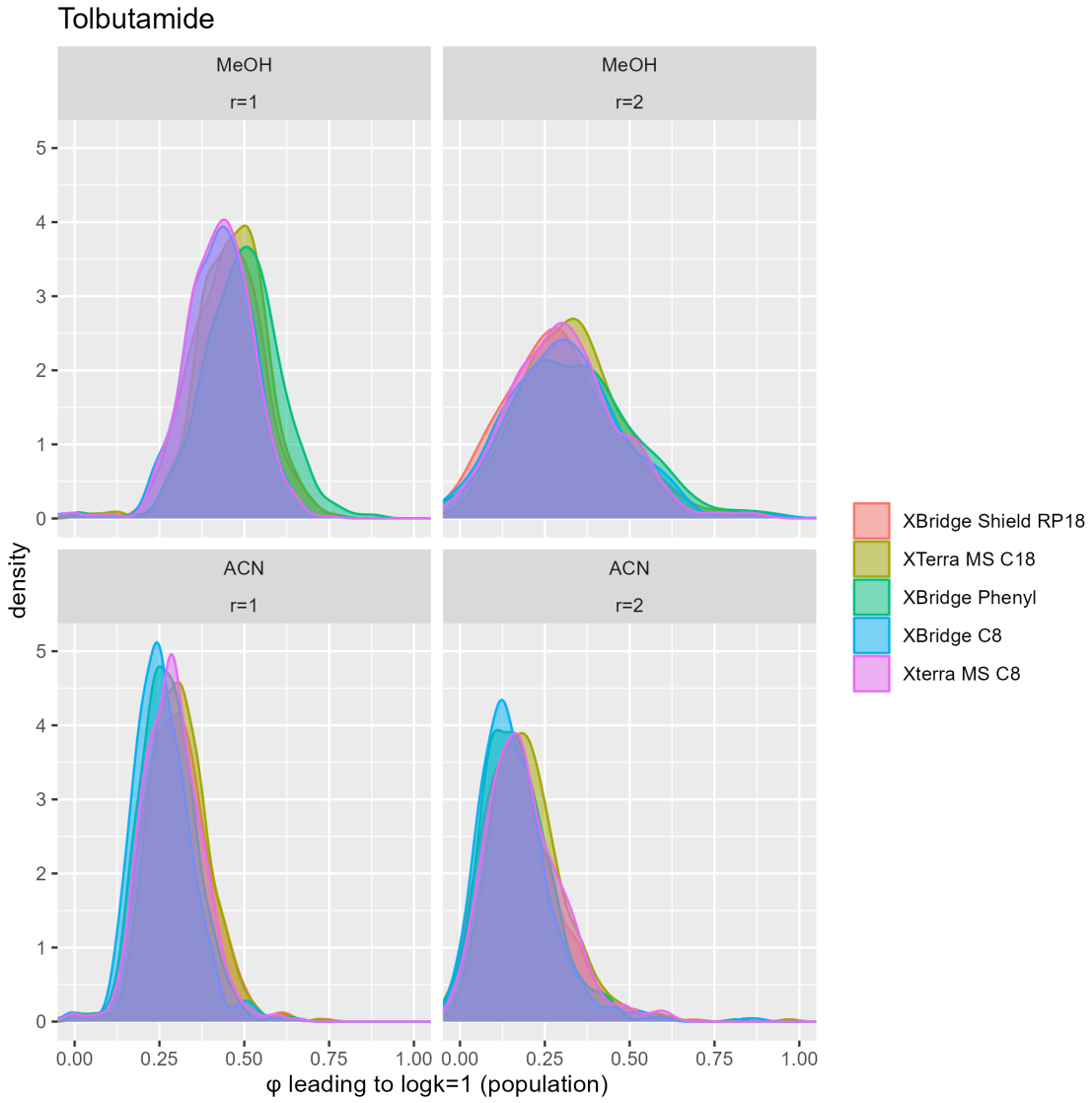

Hydrocortisone
MeOH
r=1
ACN
r=1
Between column difference in logk (population)
1.0
0.5
0.0
-0.5
-1.0
0.00
0.25
0.50
0.75
1.00
φ
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

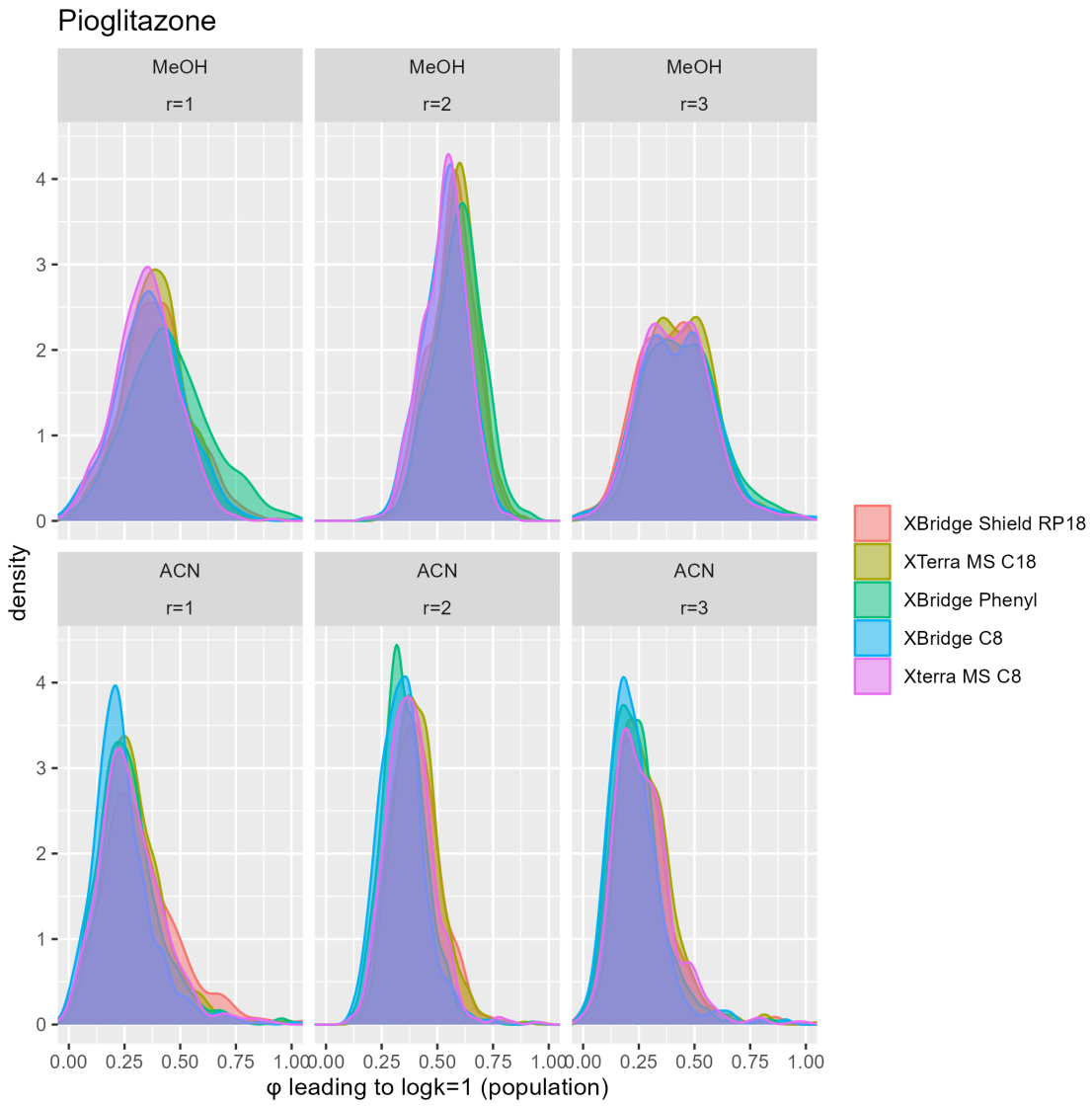


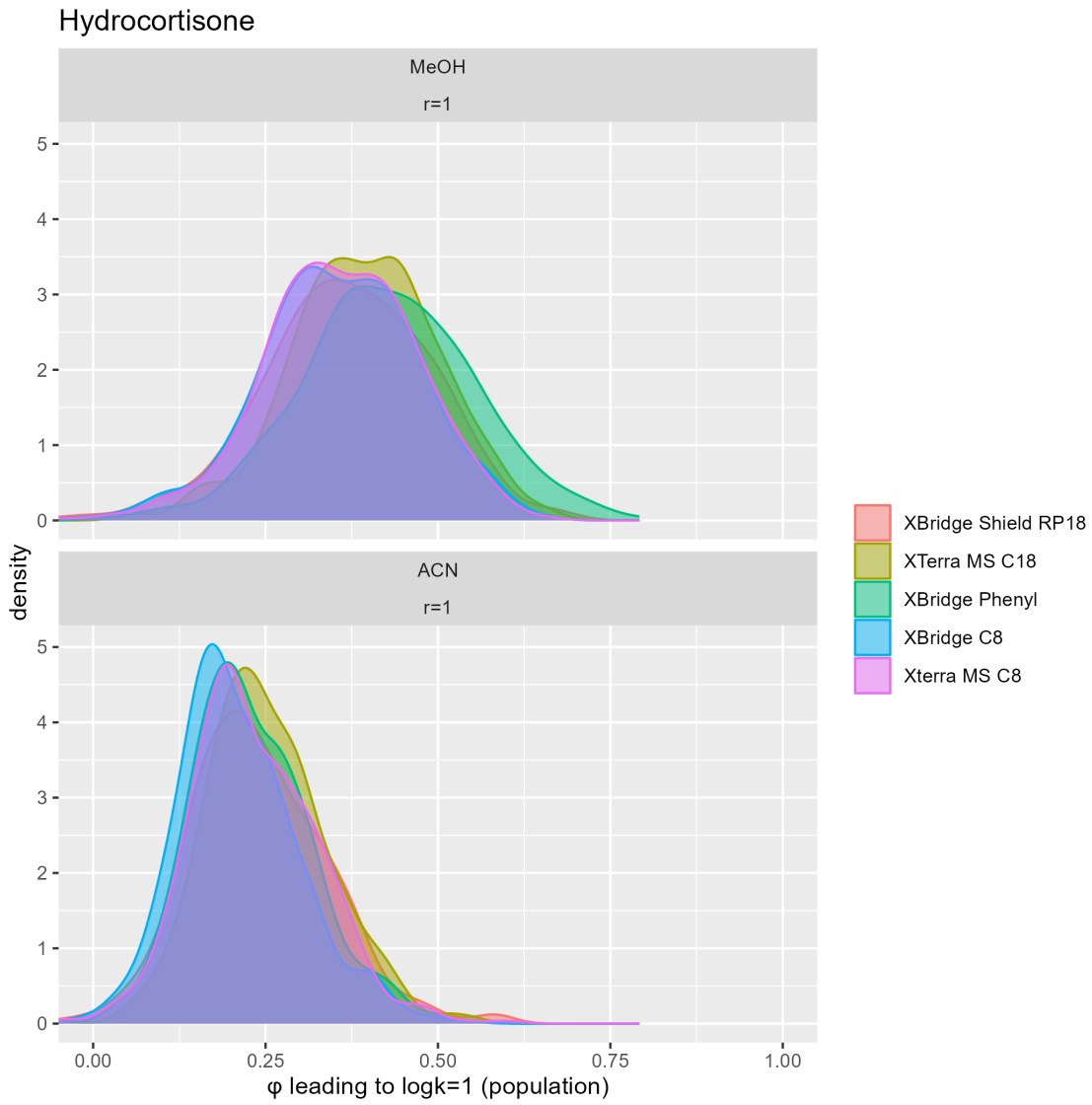
or predict the organic modifier content leading to logk of 1:

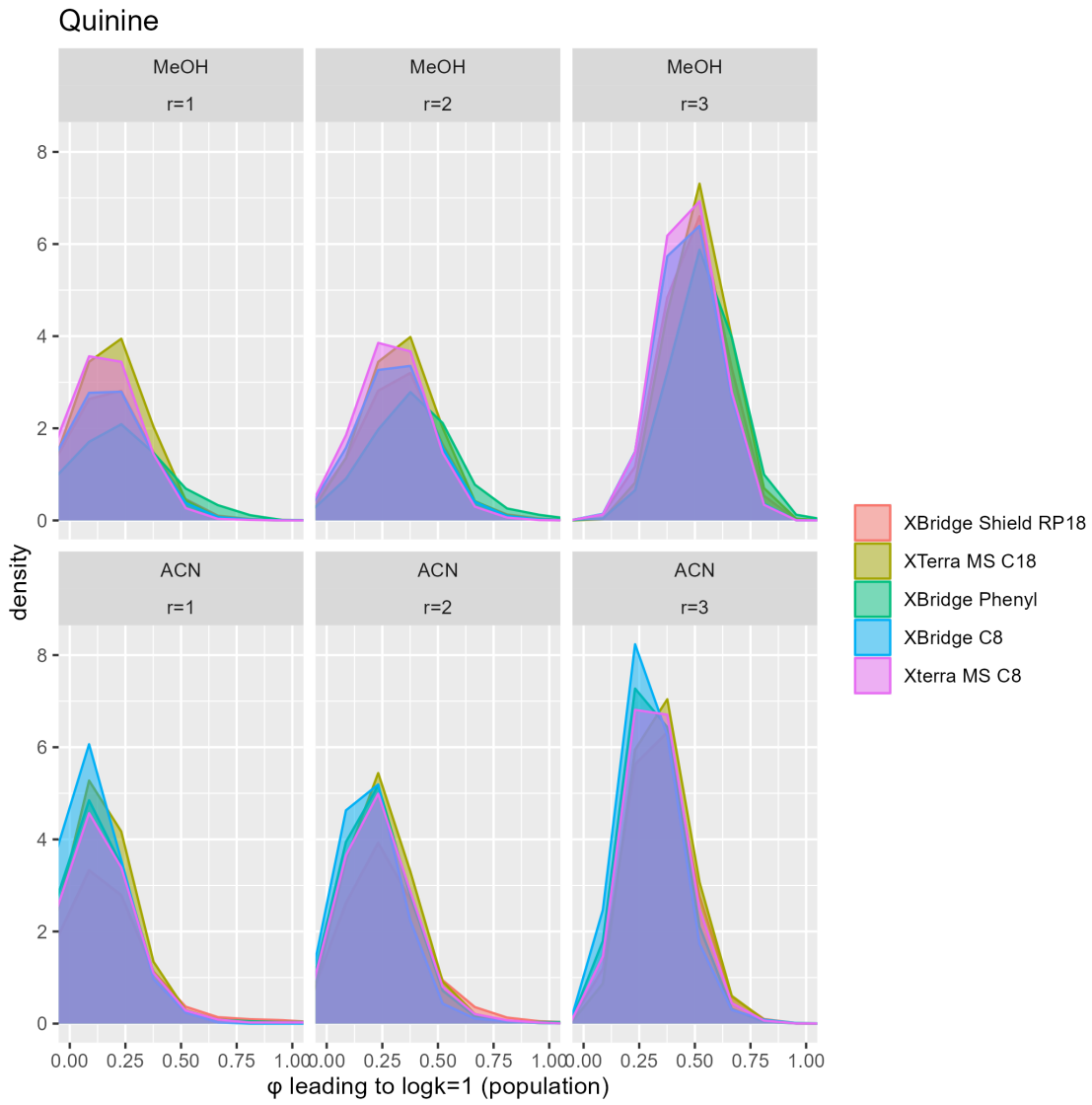

Baclofen
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
density
0
1
2
3
4
0.00
0.25
0.50
0.75
1.00
φ leading to logk=1 (population)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Acridine
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
density
0
1
2
3
4
5
0.00
0.25
0.50
0.75
1.00
φ leading to logk=1 (population)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Tolbutamide
MeOH
r=1
MeOH
r=2
ACN
r=1
ACN
r=2
density
0
1
2
3
4
5
0.00
0.25
0.50
0.75
1.00
φ leading to logk=1 (population)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8

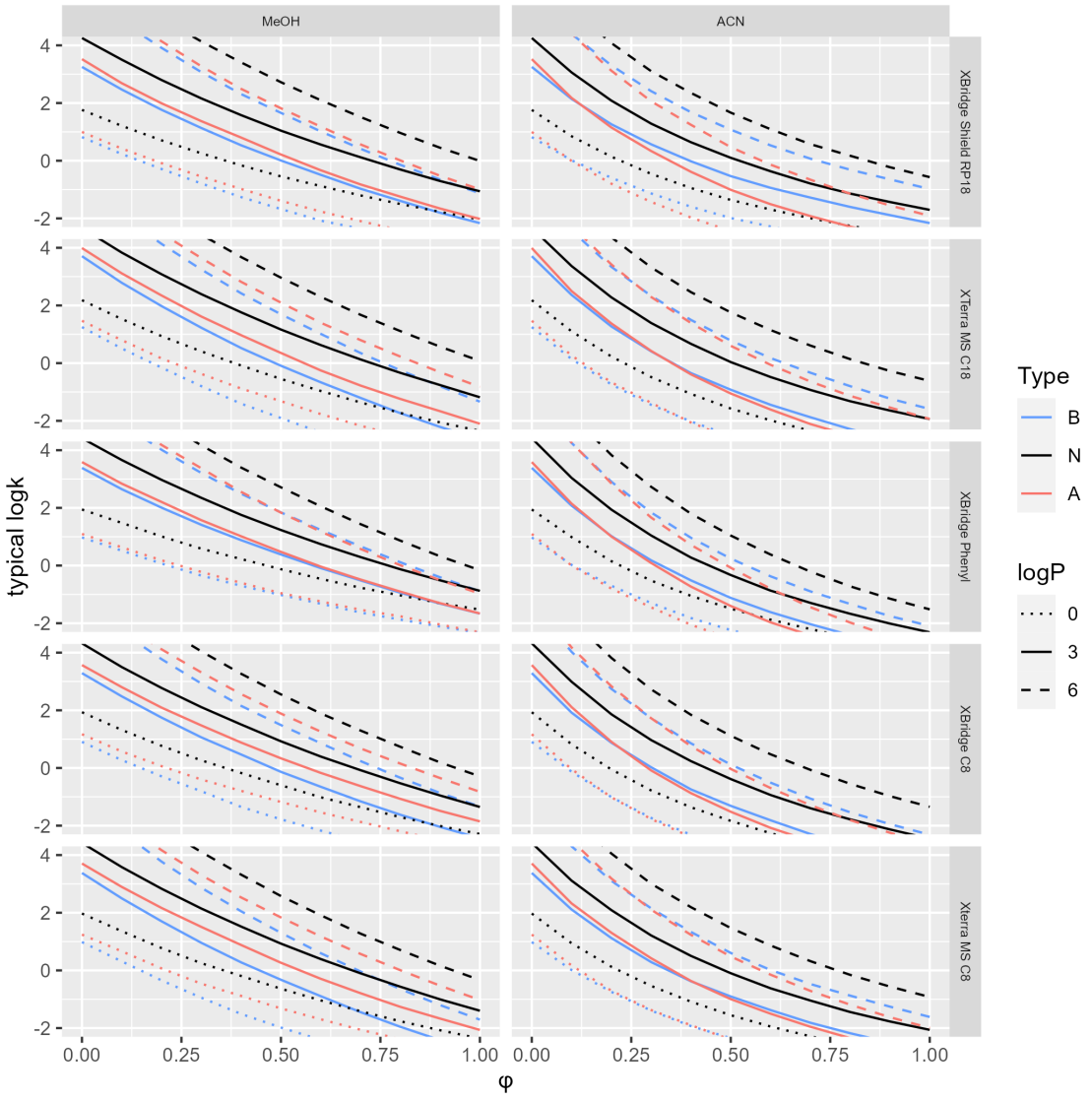

Pioglitazone
MeOH
r=1
MeOH
r=2
MeOH
r=3
ACN
r=1
ACN
r=2
ACN
r=3
density
0
1
2
3
4
0.00
0.25
0.50
0.75
1.00
φ leading to logk=1 (population)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8


Hydrocortisone
MeOH
r=1
ACN
r=1
density
0
1
2
3
4
5
0.00
0.25
0.50
0.75
1.00
φ leading to logk=1 (population)
XBridge Shield RP18
XTerra MS C18
XBridge Phenyl
XBridge C8
Xterra MS C8



9 Effect of logP

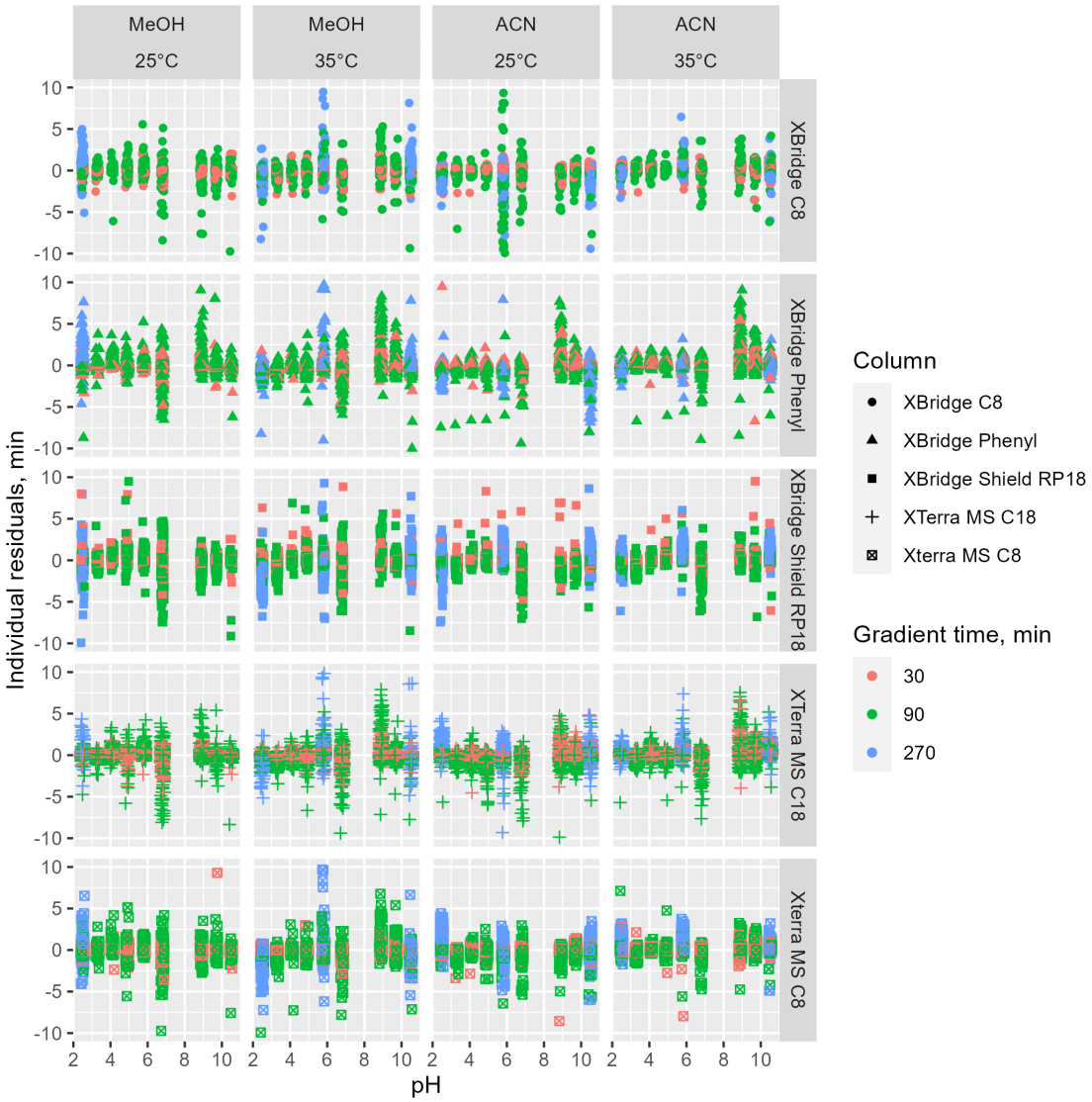
The following plot show the effect of log P on isocratic retention times (population predictions) for typical neutral, basic and acidic analyte. The uncertainties are large. They are not shown to improve visibility.



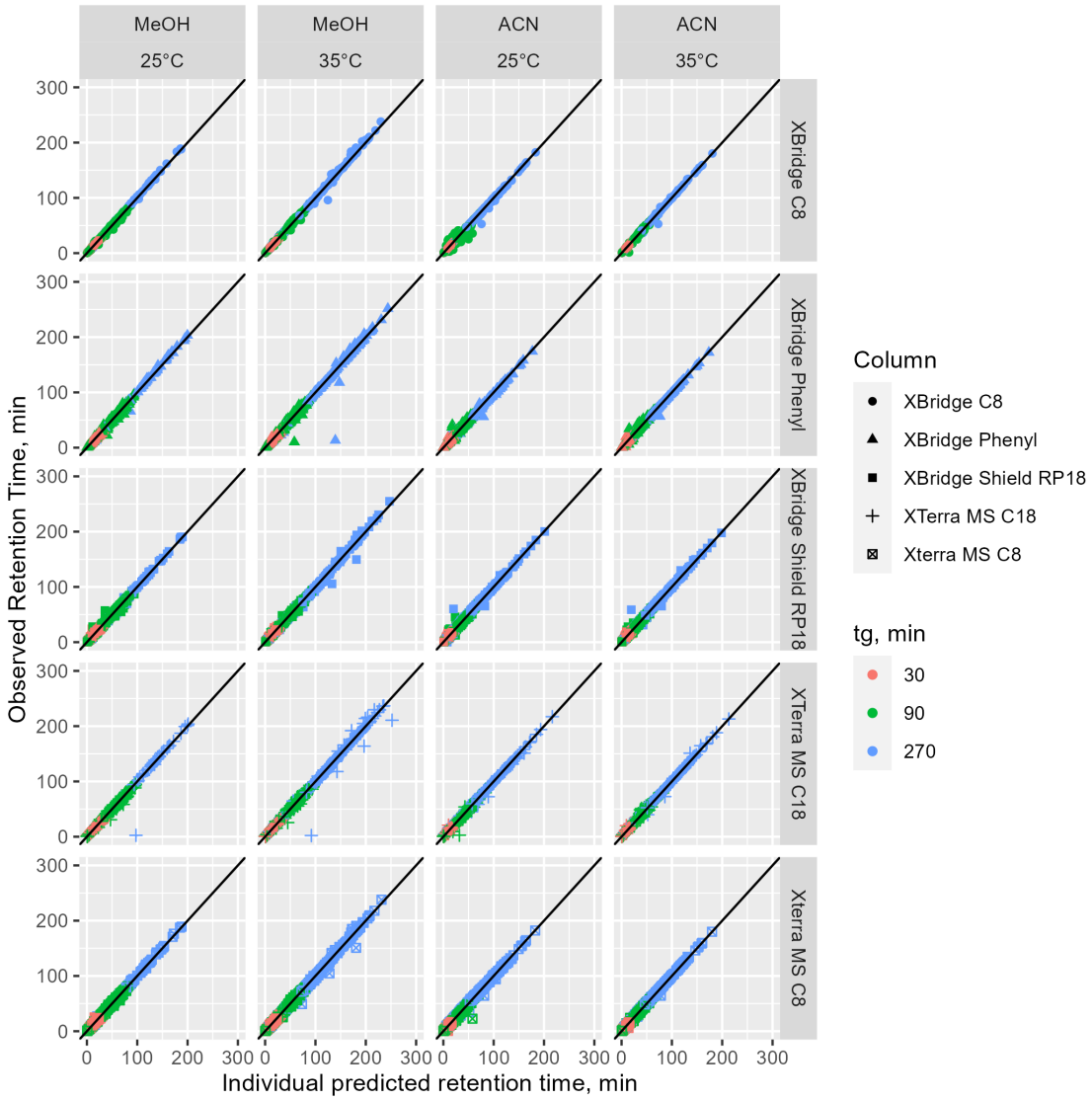
10 Goodness of fit plots

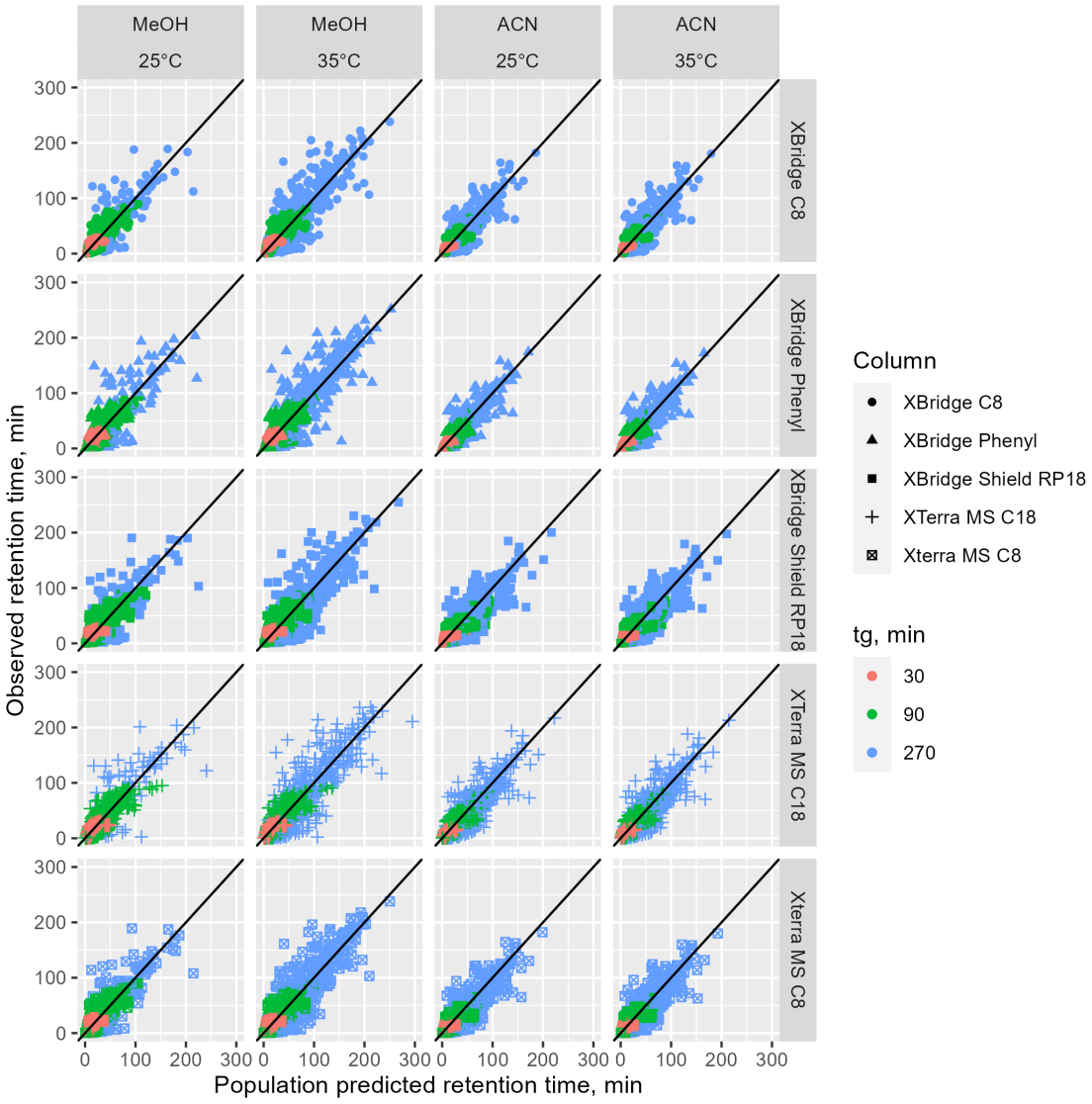
Several goodness of fit plots were used to describe how well the model fits our set of observations.

Residuals:



Observed vs. population and individual predictions:


MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Observed Retention Time, min
Individual predicted retention time, min
0
100
200
300
Column
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
tg, min
30
90
270



11 Predictions and decision making. Case Study 1

Here we want to show applicability of the model to predicting the best chromatogram using all the available data.

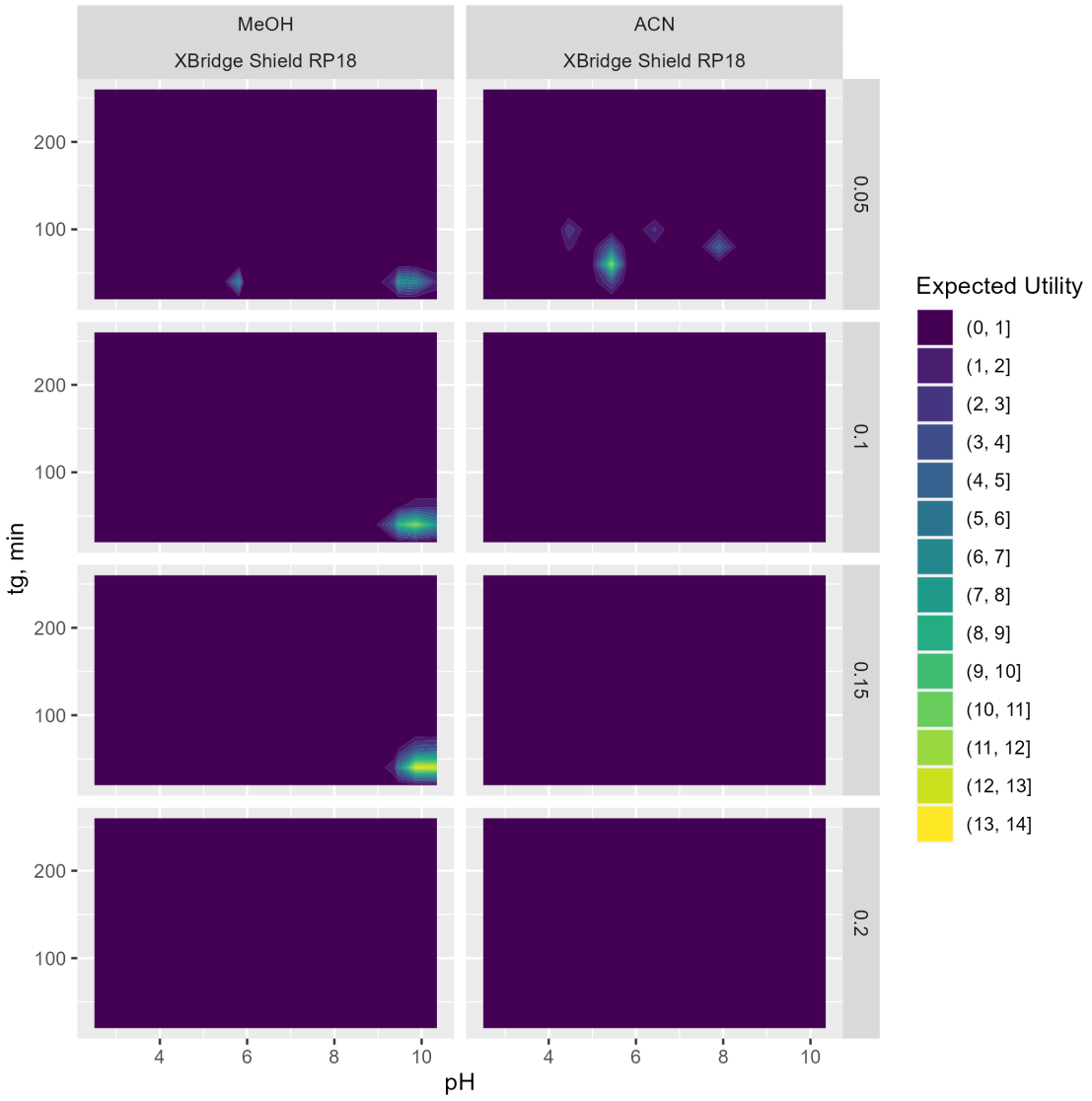
11.1 Utility maps

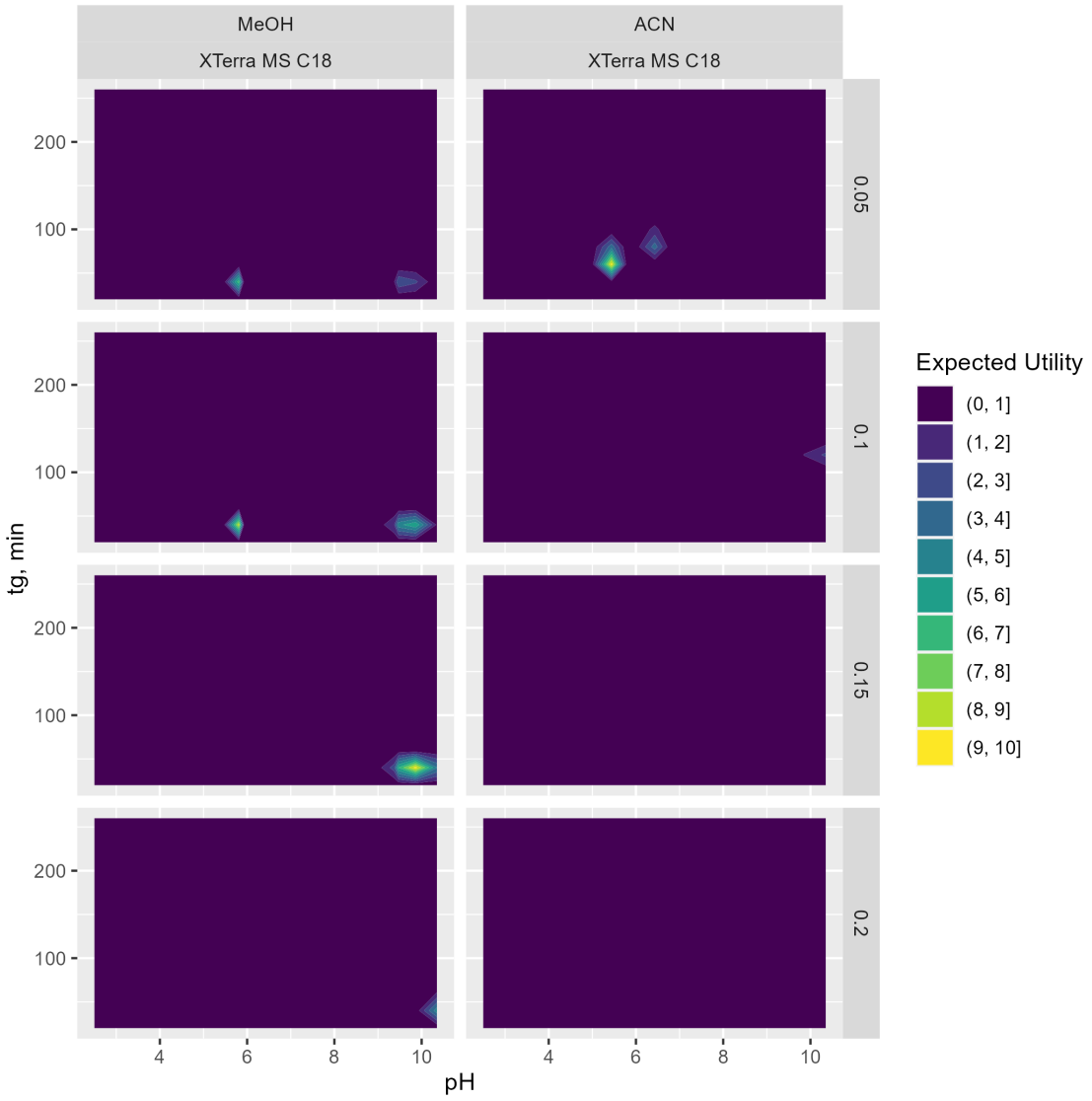
Let's find the best chromatogram that maximize utility using the set of experiments used to build the model.

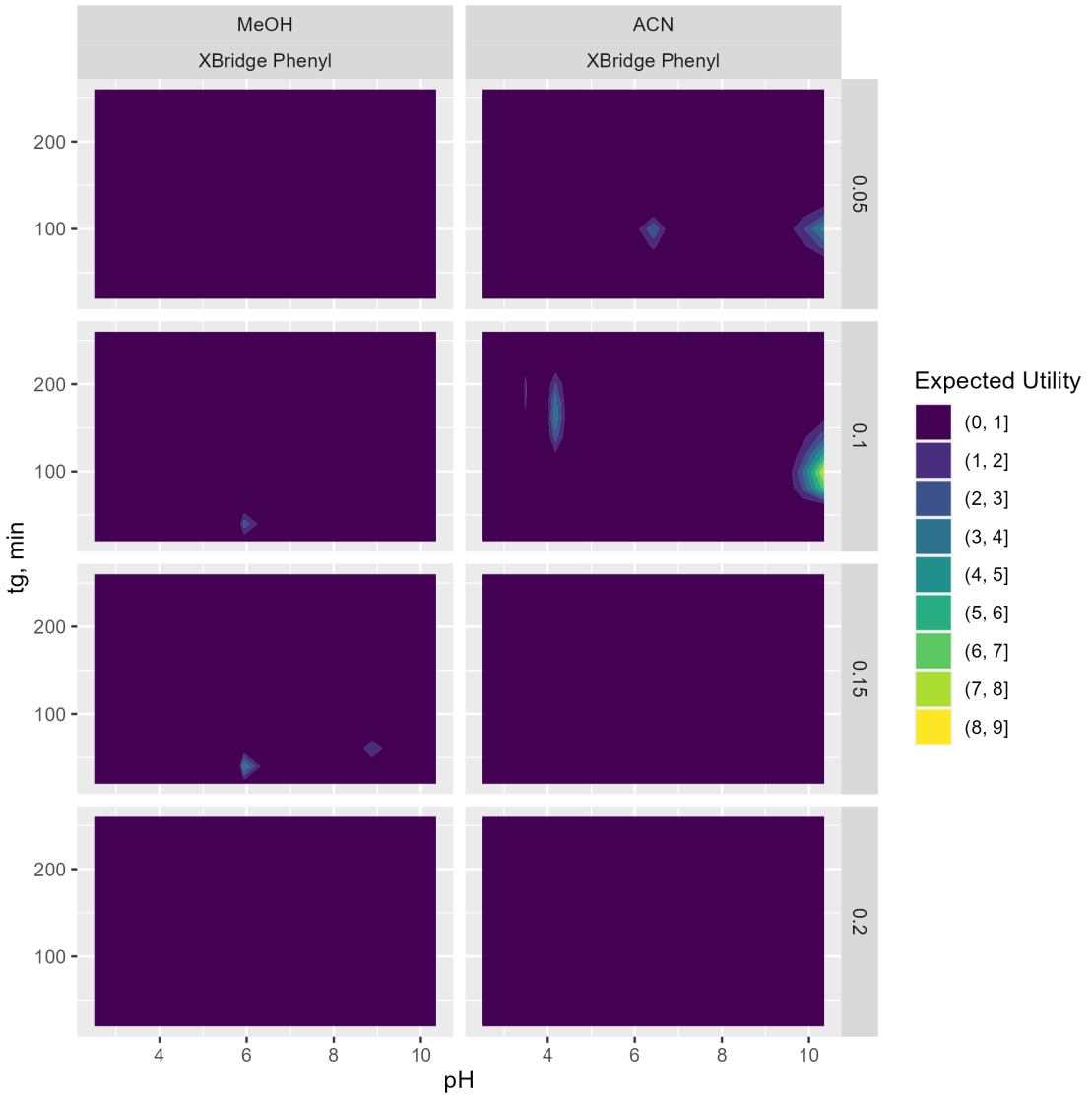
The utility function is based on lowest retention time, highest retention time and the difference of retention times between the critical pair of analytes. This utility is zero if at least one of the analyte has retention higher that 40, less than 2 min or the difference in retention time is less than 2. Otherwise, it favors shorter runs. (.) denotes design variables.

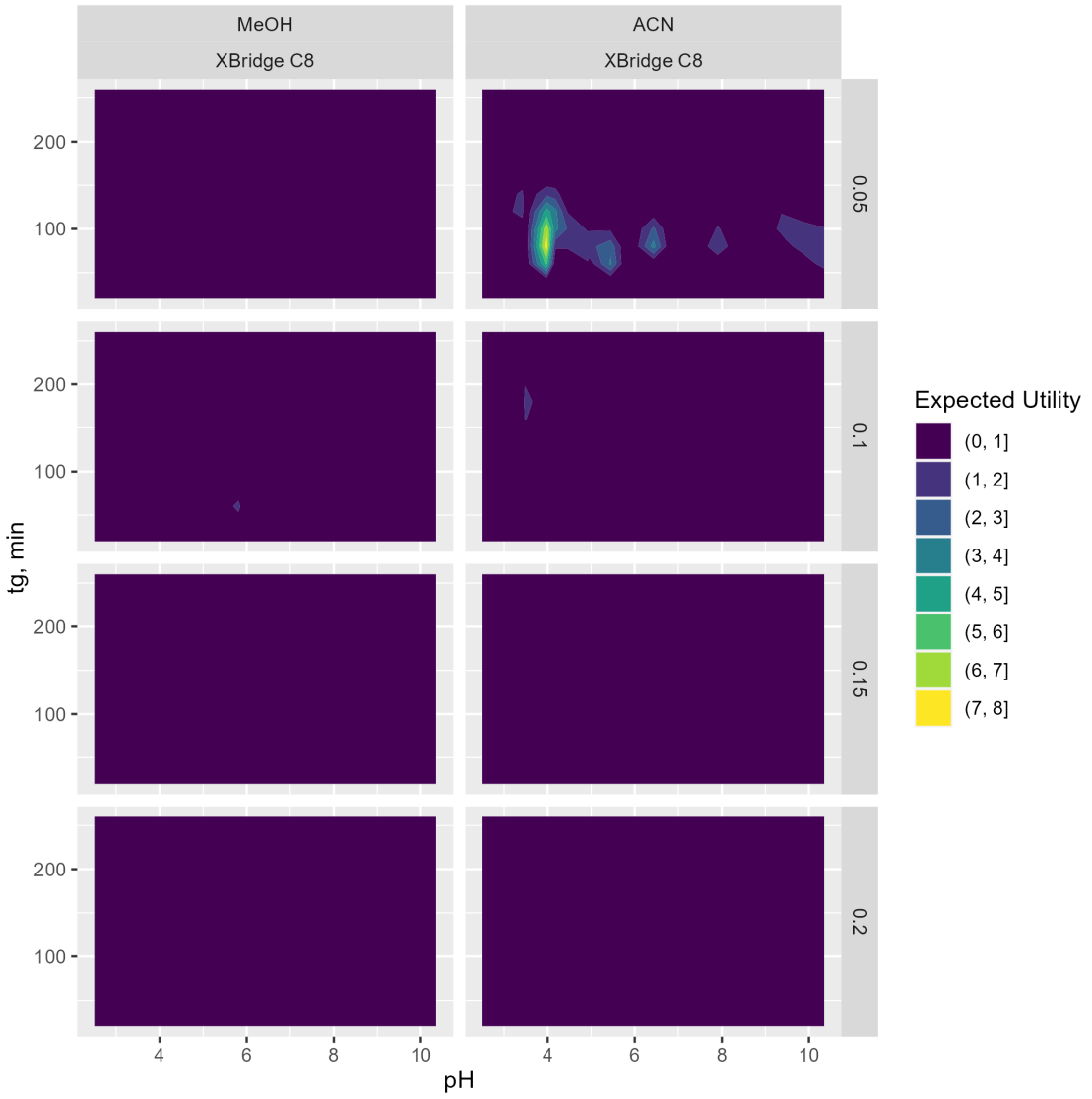
$$U(.) = I((mintr(.) > 2)\ \&\ (res(.) > 2)) \cdot (40 - maxtr(.)) \cdot I(maxtr(.) < 40)$$

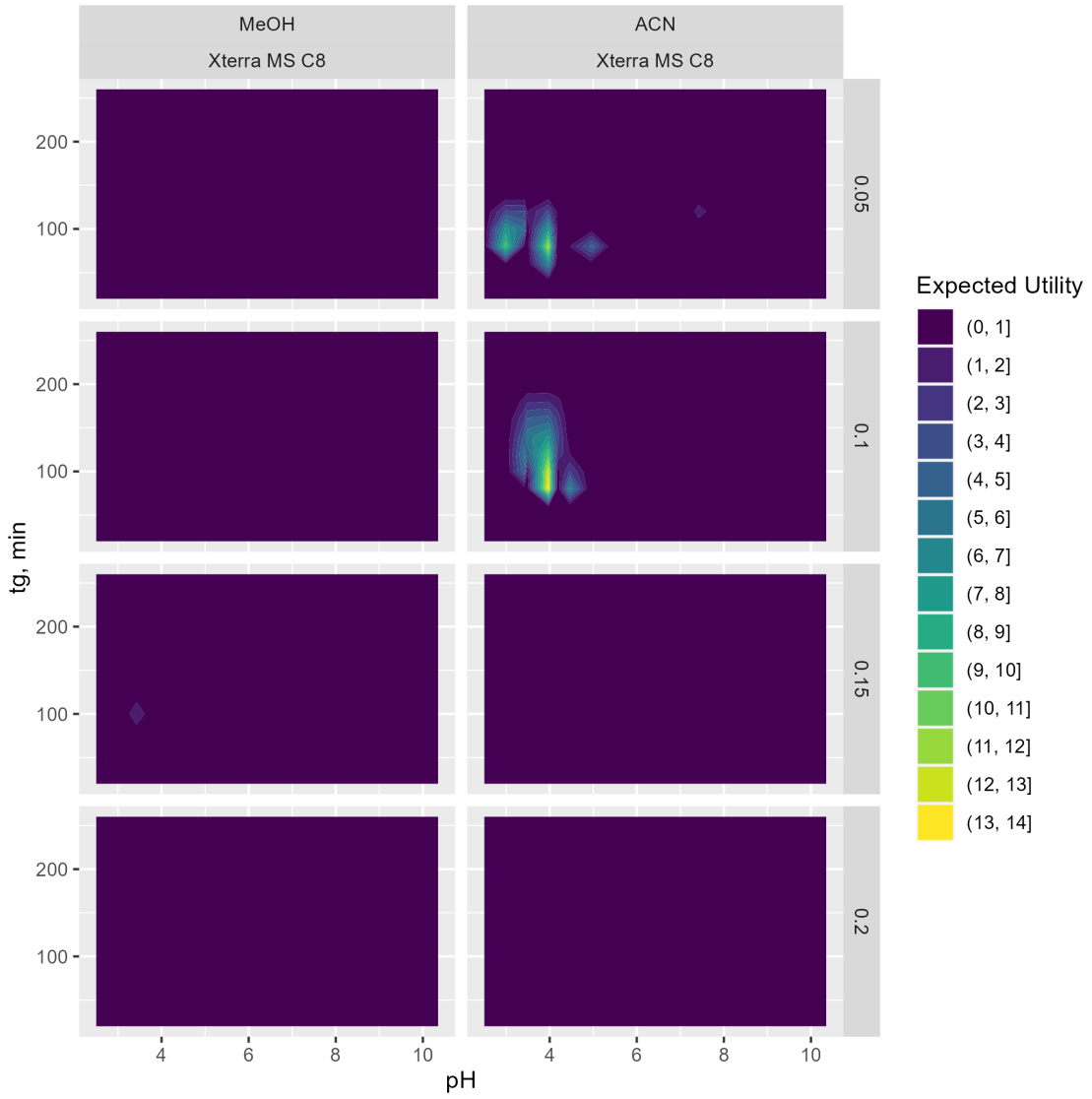
$$I(x) = 1\ if\ condition\ x\ is\ trueI(x) = 0\ if\ condition\ x\ is\ false$$


MeOH
XBridge Shield RP18
ACN
XBridge Shield RP18
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0, 1]
(1, 2]
(2, 3]
(3, 4]
(4, 5]
(5, 6]
(6, 7]
(7, 8]
(8, 9]
(9, 10]
(10, 11]
(11, 12]
(12, 13]
(13, 14]


MeOH
XTerra MS C18
ACN
XTerra MS C18
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0, 1]
(1, 2]
(2, 3]
(3, 4]
(4, 5]
(5, 6]
(6, 7]
(7, 8]
(8, 9]
(9, 10]


MeOH
XBridge Phenyl
ACN
XBridge Phenyl
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0, 1]
(1, 2]
(2, 3]
(3, 4]
(4, 5]
(5, 6]
(6, 7]
(7, 8]
(8, 9]


MeOH
XBridge C8
ACN
XBridge C8
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0, 1]
(1, 2]
(2, 3]
(3, 4]
(4, 5]
(5, 6]
(6, 7]
(7, 8]




```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName          EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>                  <dbl>
1    60  0.05   0.8 ACN      10  5.50    25 XBridge Shield RP18     11.4
```

```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName    EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>            <dbl>
1    60  0.05   0.8 ACN      10  5.50    25 XTerra MS C18     9.50
```

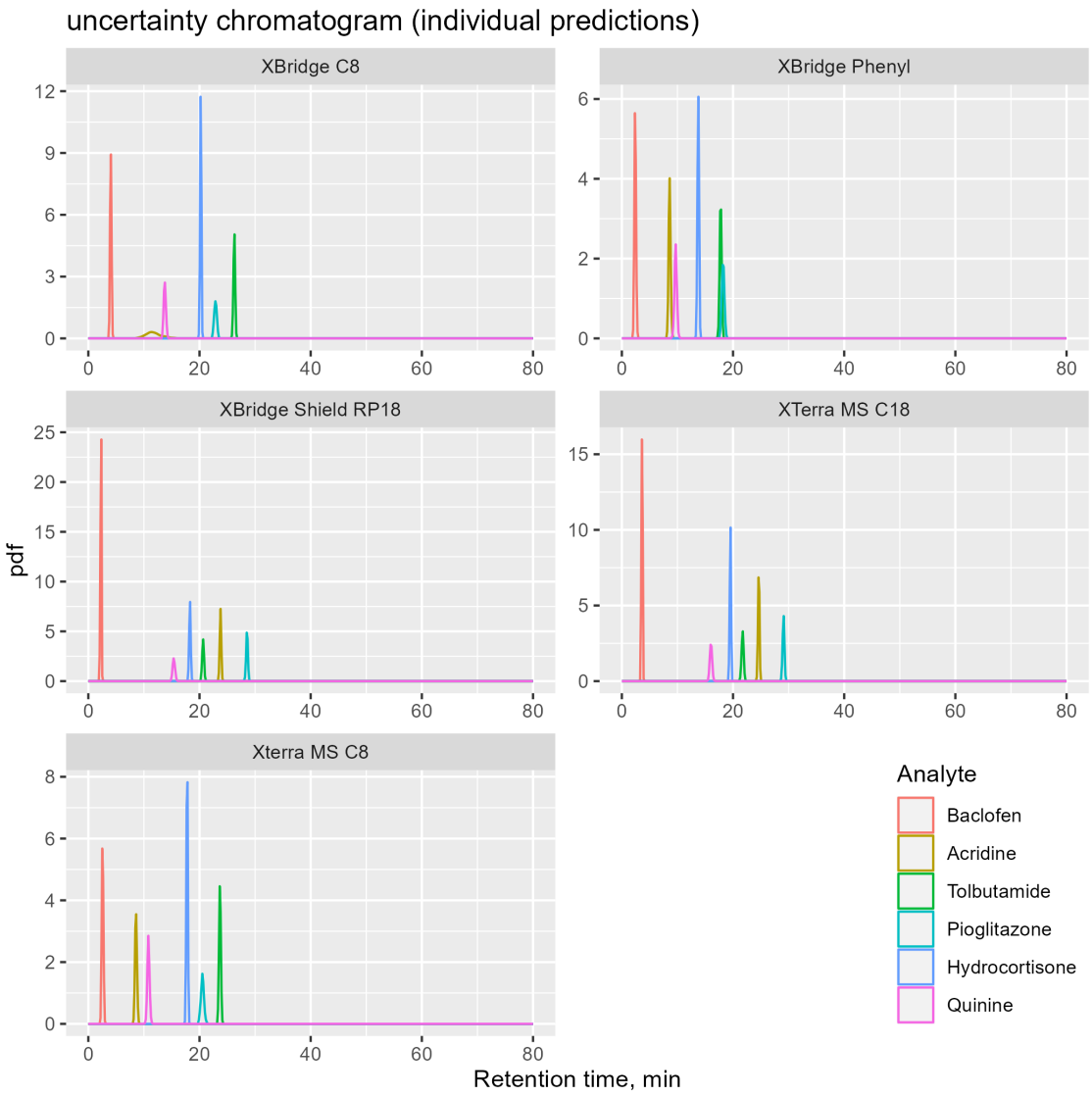
```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName     EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>             <dbl>
1    40  0.15   0.8 MeOH     12  6.00    25 XBridge Phenyl     4.08
```

```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>          <dbl>
1    80  0.05   0.8 ACN       5  4.00    25 XBridge C8     7.73
```

```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName   EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>           <dbl>
1    80   0.1   0.8 ACN       5  4.00    25 Xterra MS C8     13.7
```

11.1.1 Uncertainity chromatogram

Let's visualize chromatograms with the highest expected utility



12 Predictions and decision making. Case 2

The proposed model can be used for decision making having access to limiet data. Here we want to ilustrate this by predicting XTerra MS C18, XBridge Phenyl, XBridge C8, Xterra MS C8 retention times based on XBridge Shield RP18 data.

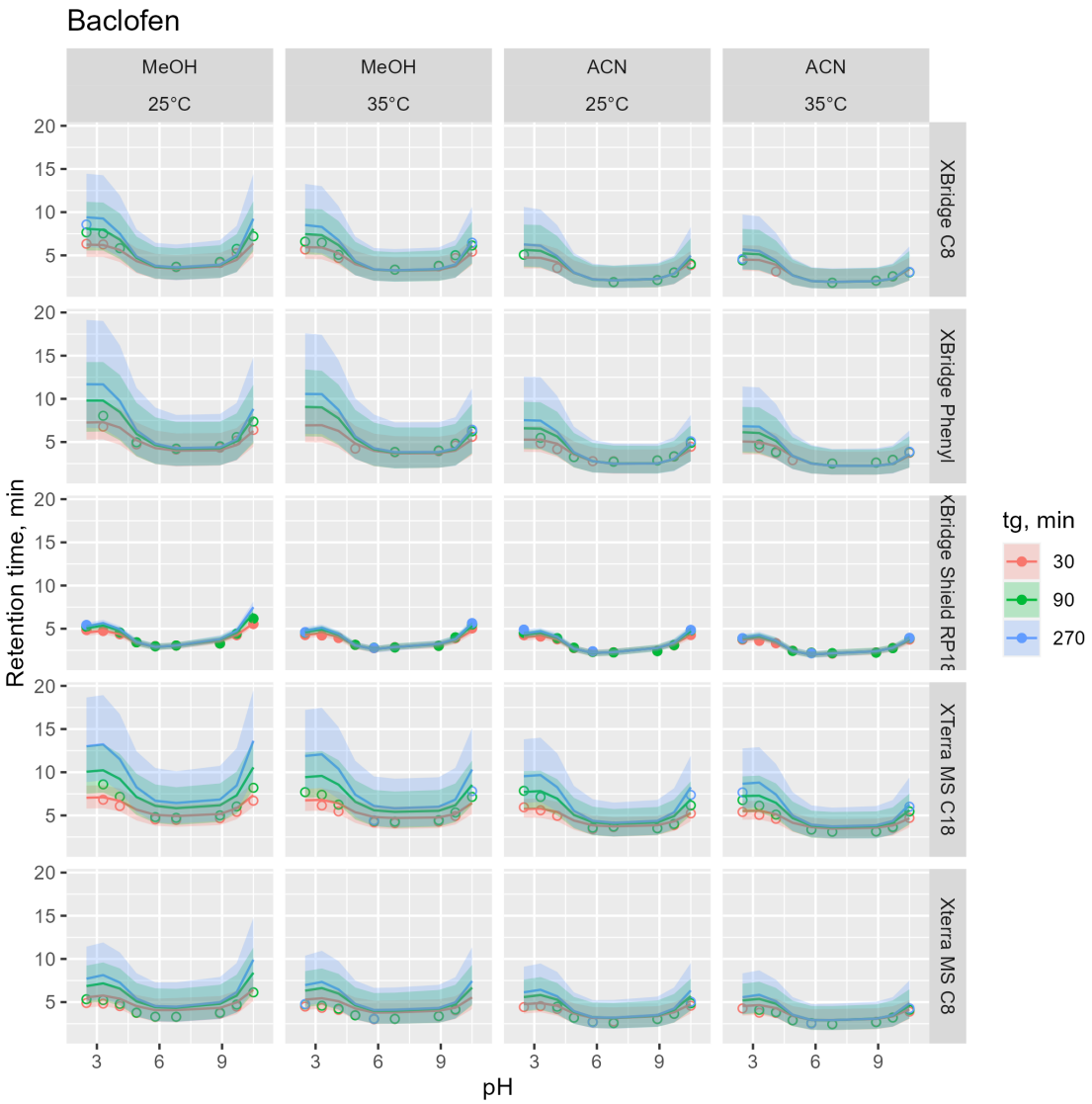
12.1 Summary of individual parameters

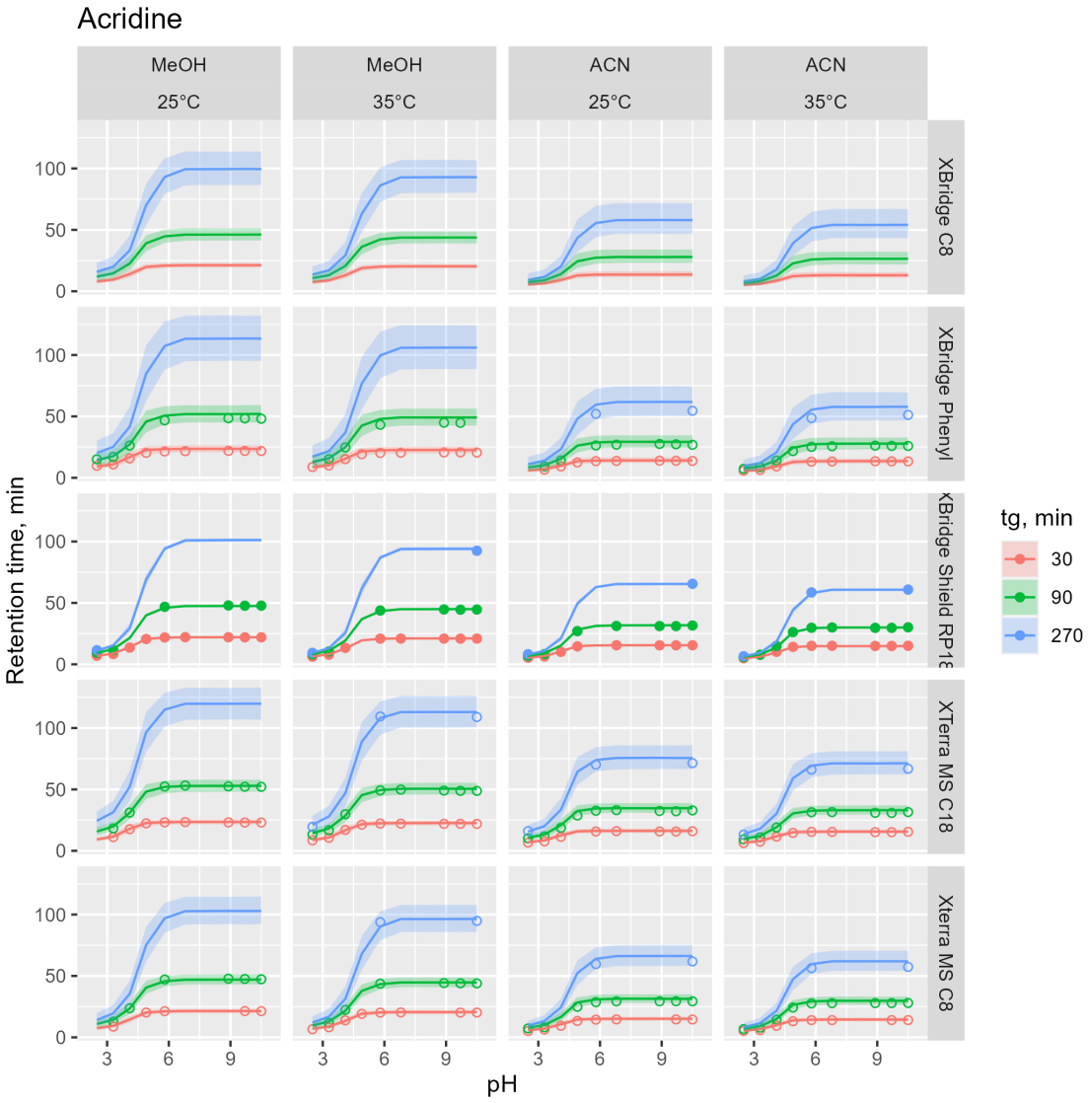
Here shown for analyte 9 (Baclofen)

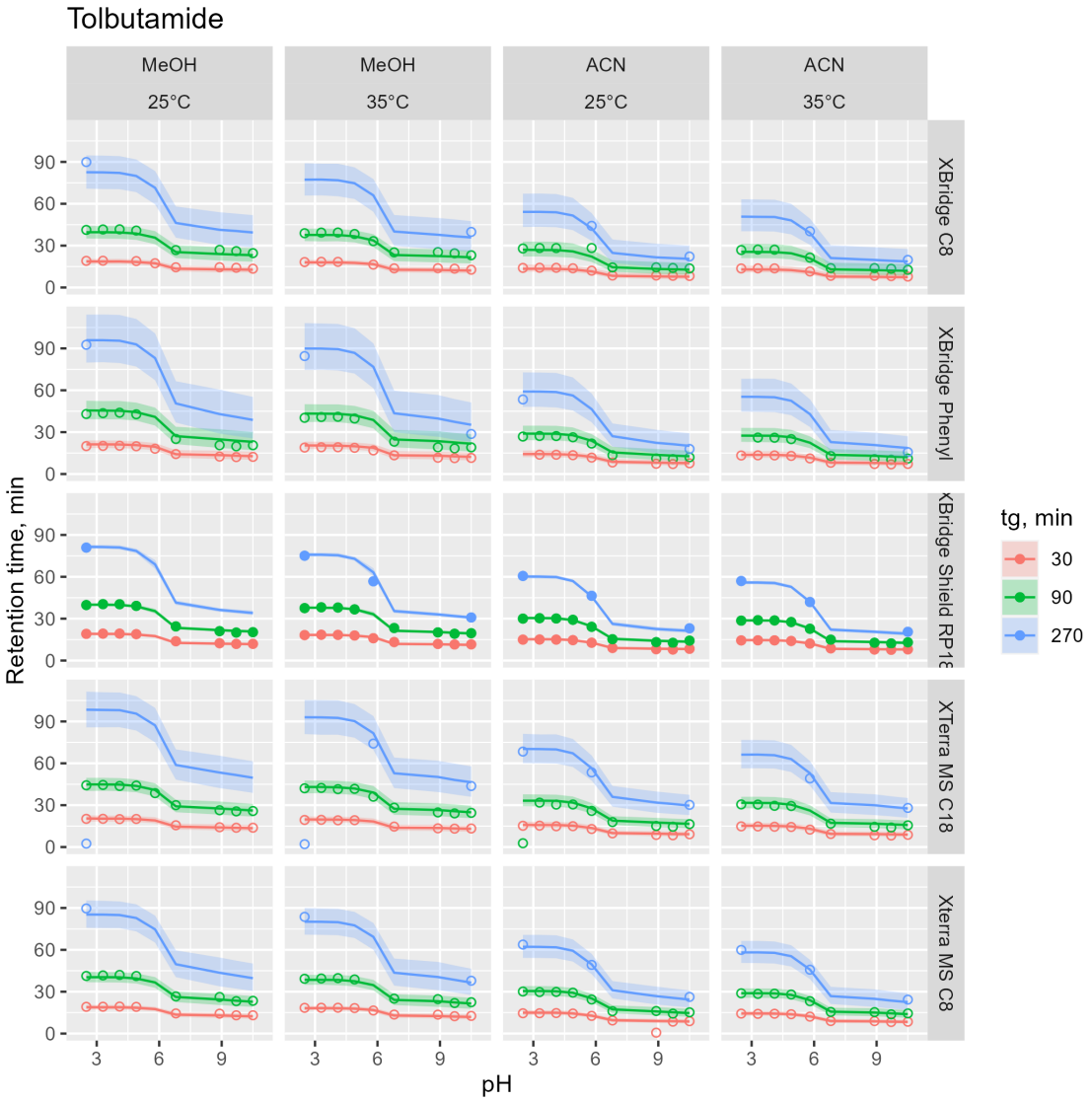
variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
logkwx[3,1,1]	1.84	1.84	0.92	0.92	0.39	3.40	1.00	1704	1267
logkwx[3,1,2]	1.05	1.04	1.11	1.12	-0.75	2.93	1.00	2005	1563
logkwx[3,1,3]	1.05	1.04	1.11	1.12	-0.75	2.93	1.00	2005	1563
logkwx[3,2,1]	2.27	2.25	0.92	0.92	0.77	3.80	1.00	1710	1359
logkwx[3,2,2]	1.53	1.52	1.12	1.14	-0.30	3.42	1.00	2020	1416
logkwx[3,2,3]	1.53	1.52	1.12	1.14	-0.30	3.42	1.00	2020	1416
S1x[3,1,1,1]	3.87	3.88	0.95	0.93	2.27	5.45	1.00	1679	1326
S1x[3,1,1,2]	4.08	4.05	1.17	1.16	2.18	5.96	1.00	1965	1502
S1x[3,1,1,3]	4.08	4.05	1.17	1.16	2.18	5.96	1.00	1965	1502
S1x[3,2,1,1]	4.49	4.50	1.09	1.06	2.66	6.34	1.00	1912	1237
S1x[3,2,1,2]	4.99	5.00	1.39	1.46	2.78	7.28	1.01	2269	1256
S1x[3,2,1,3]	4.99	5.00	1.39	1.46	2.78	7.28	1.01	2269	1256
S1x[3,1,2,1]	4.61	4.62	0.95	0.94	3.03	6.22	1.00	1664	1326
S1x[3,1,2,2]	4.80	4.77	1.18	1.15	2.88	6.71	1.00	1968	1479
S1x[3,1,2,3]	4.80	4.77	1.18	1.15	2.88	6.71	1.00	1968	1479
S1x[3,2,2,1]	5.38	5.41	1.10	1.06	3.57	7.22	1.00	1823	1305
S1x[3,2,2,2]	5.91	5.91	1.41	1.47	3.67	8.20	1.01	2234	1194
S1x[3,2,2,3]	5.91	5.91	1.41	1.47	3.67	8.20	1.01	2234	1194
apHx[3,1,1]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
apHx[3,1,2]	-0.03	-0.03	0.00	0.00	-0.03	-0.03	1.00	1913	1394
apHx[3,1,3]	-0.03	-0.03	0.00	0.00	-0.03	-0.03	1.00	1913	1394
pKawx[3,1]	7.35	7.35	0.84	0.84	5.95	8.74	1.00	3737	1279
pKawx[3,2]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
alphax[3,1,1]	2.19	2.19	0.96	0.95	0.60	3.74	1.00	4654	1273
alphax[3,1,2]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
alphax[3,2,1]	2.39	2.39	1.29	1.31	0.25	4.54	1.00	4518	1318
alphax[3,2,2]	0.00	0.00	0.00	0.00	0.00	0.00	NA	NA	NA
S2x[1,1]	0.47	0.47	0.02	0.02	0.44	0.50	1.00	893	1183
S2x[2,1]	1.20	1.20	0.04	0.04	1.13	1.27	1.00	670	1346
sigmax[3,1]	0.54	0.39	0.54	0.29	0.10	1.45	1.00	3967	1379
sigmax[3,2]	0.66	0.46	0.65	0.35	0.12	1.76	1.00	3947	1422

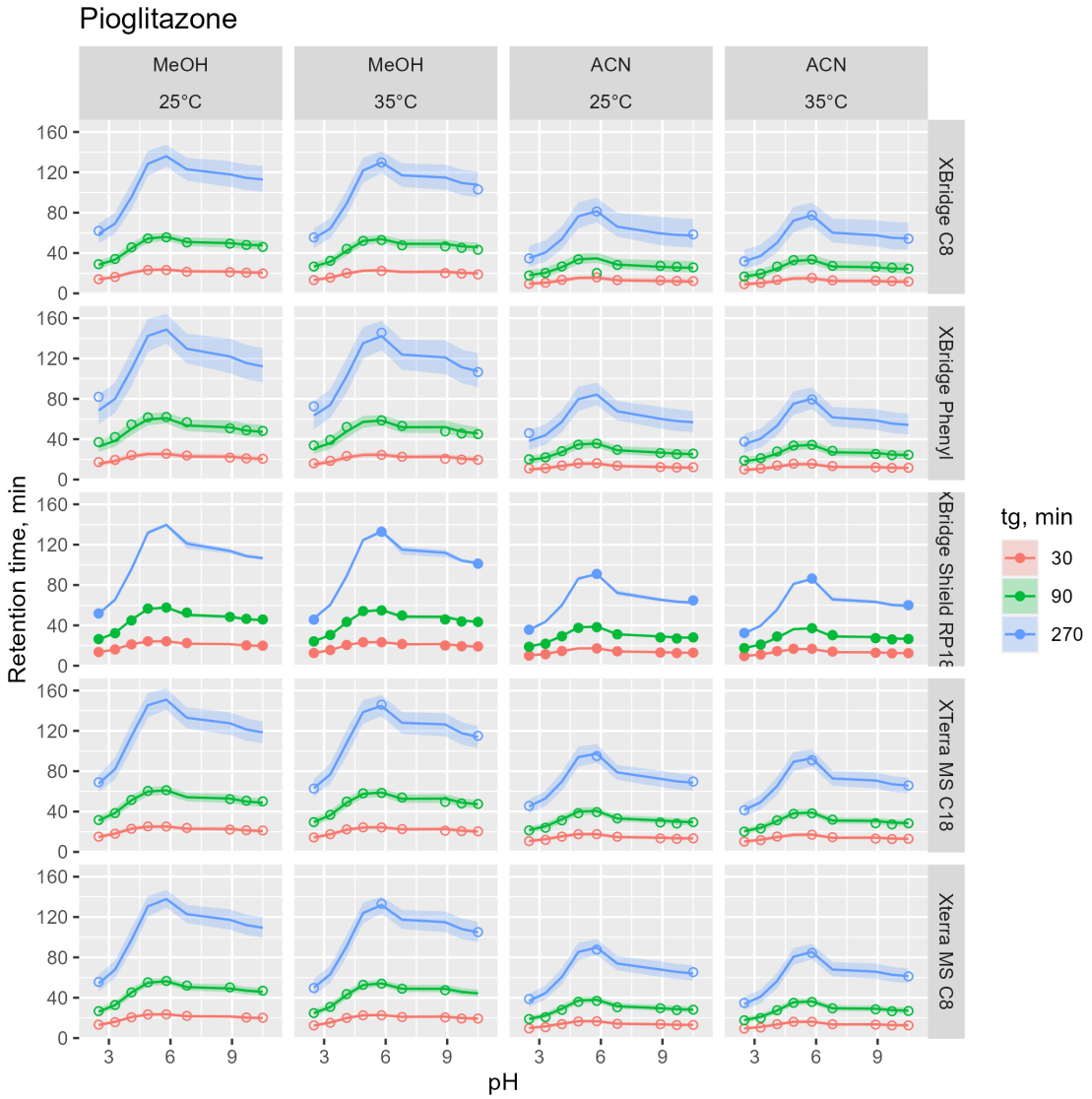
12.2 Predicted retention times

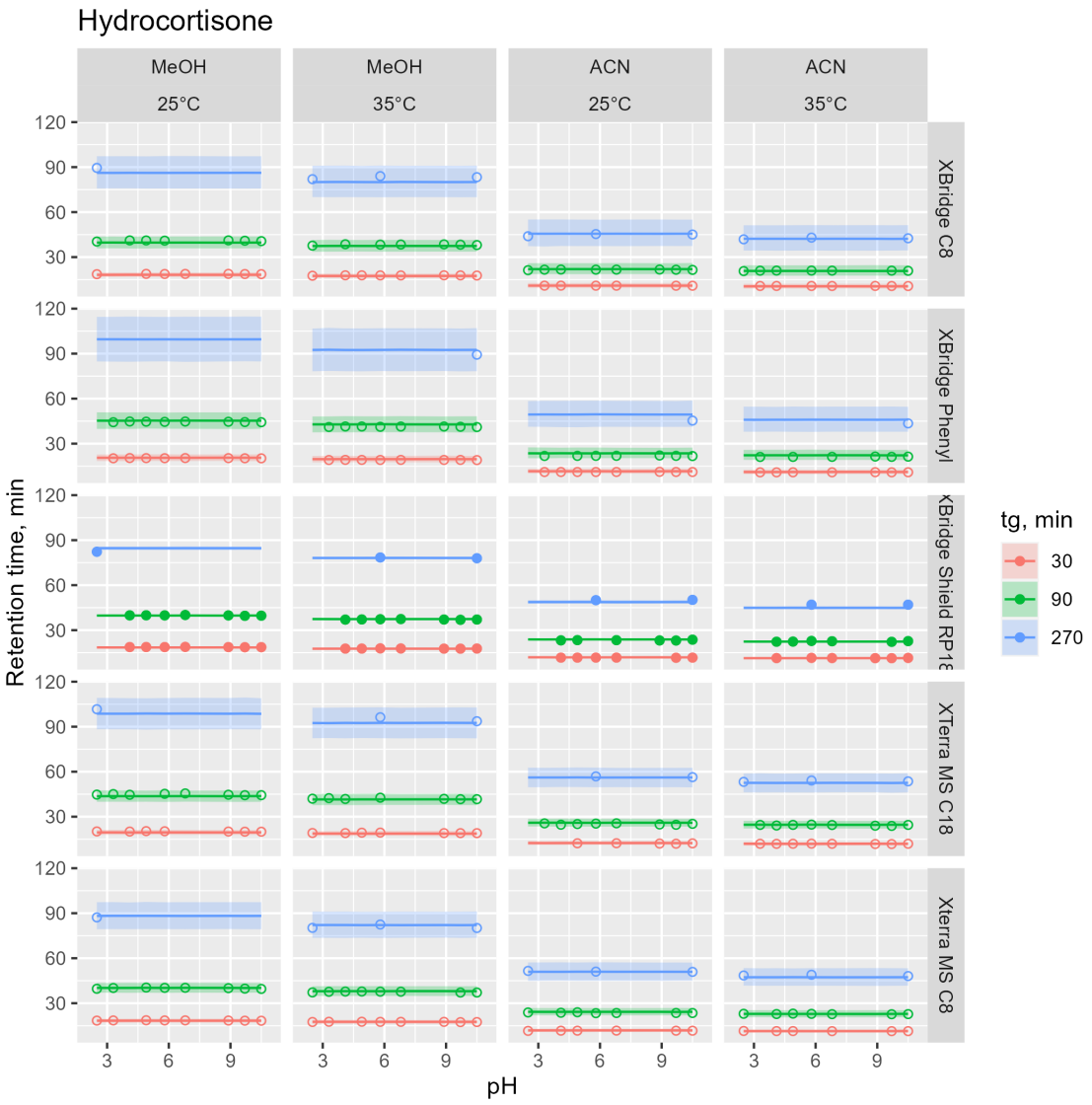
12.2.1 Plot the predicted vs. observed:

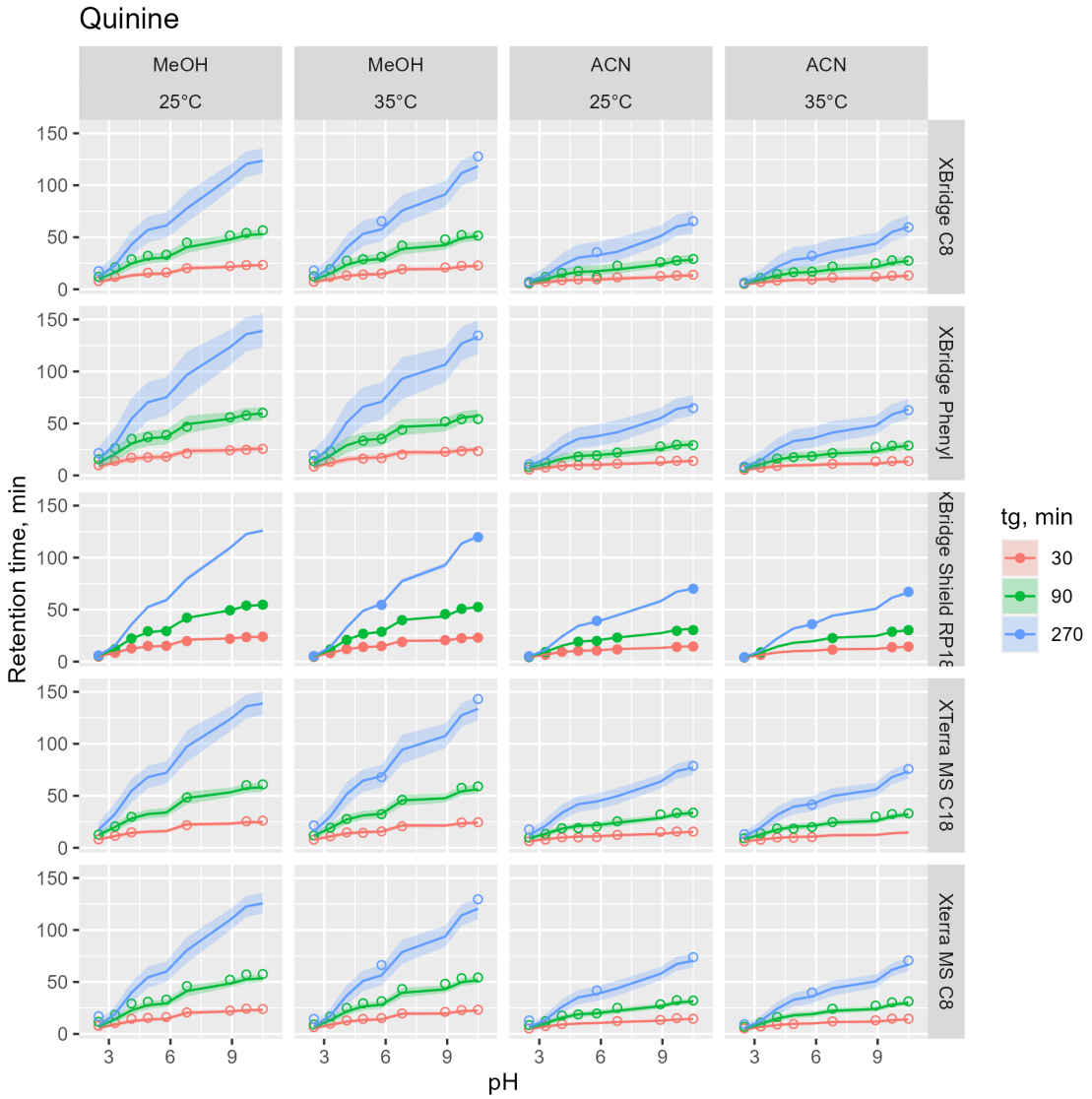



Acridine
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time, min
pH
tg, min
30
90
270


Tolbutamide
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time, min
pH
tg, min
30
90
270

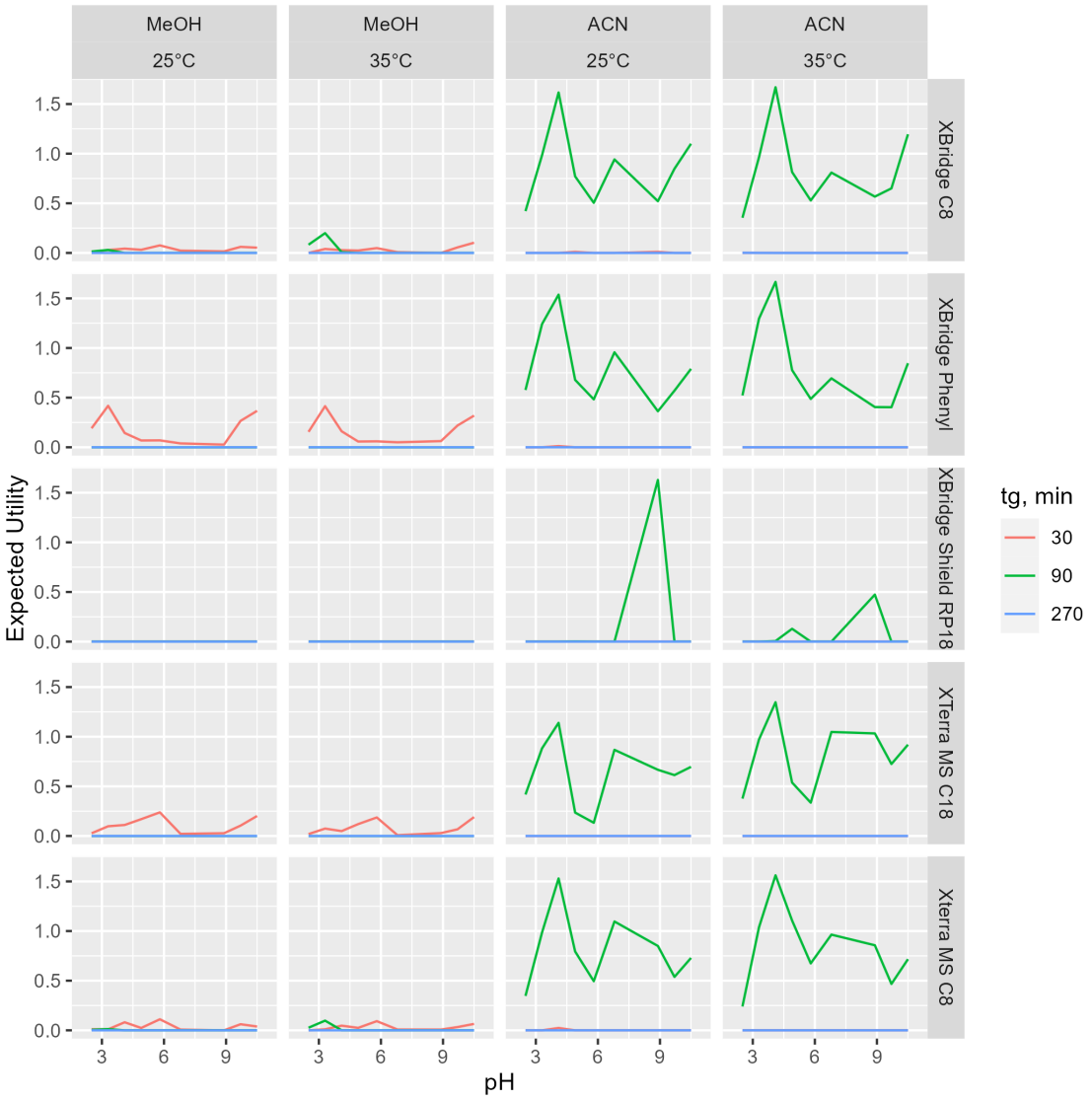

Pioglitazone
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XTerra MS C18
Xterra MS C8
Retention time, min
pH
tg, min
30
90
270

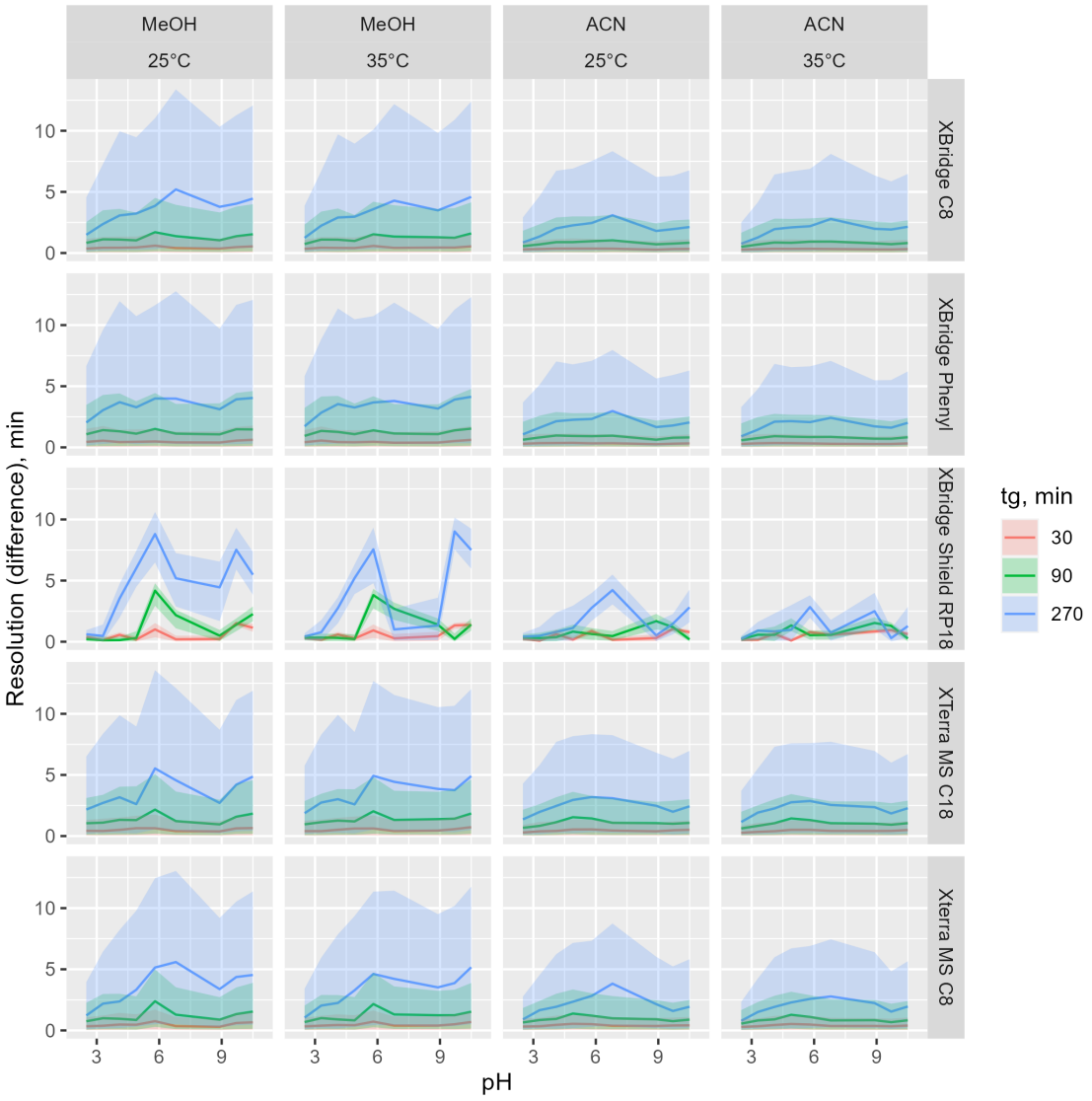

Hydrocortisone
MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Retention time, min
pH
tg, min
30
90
270

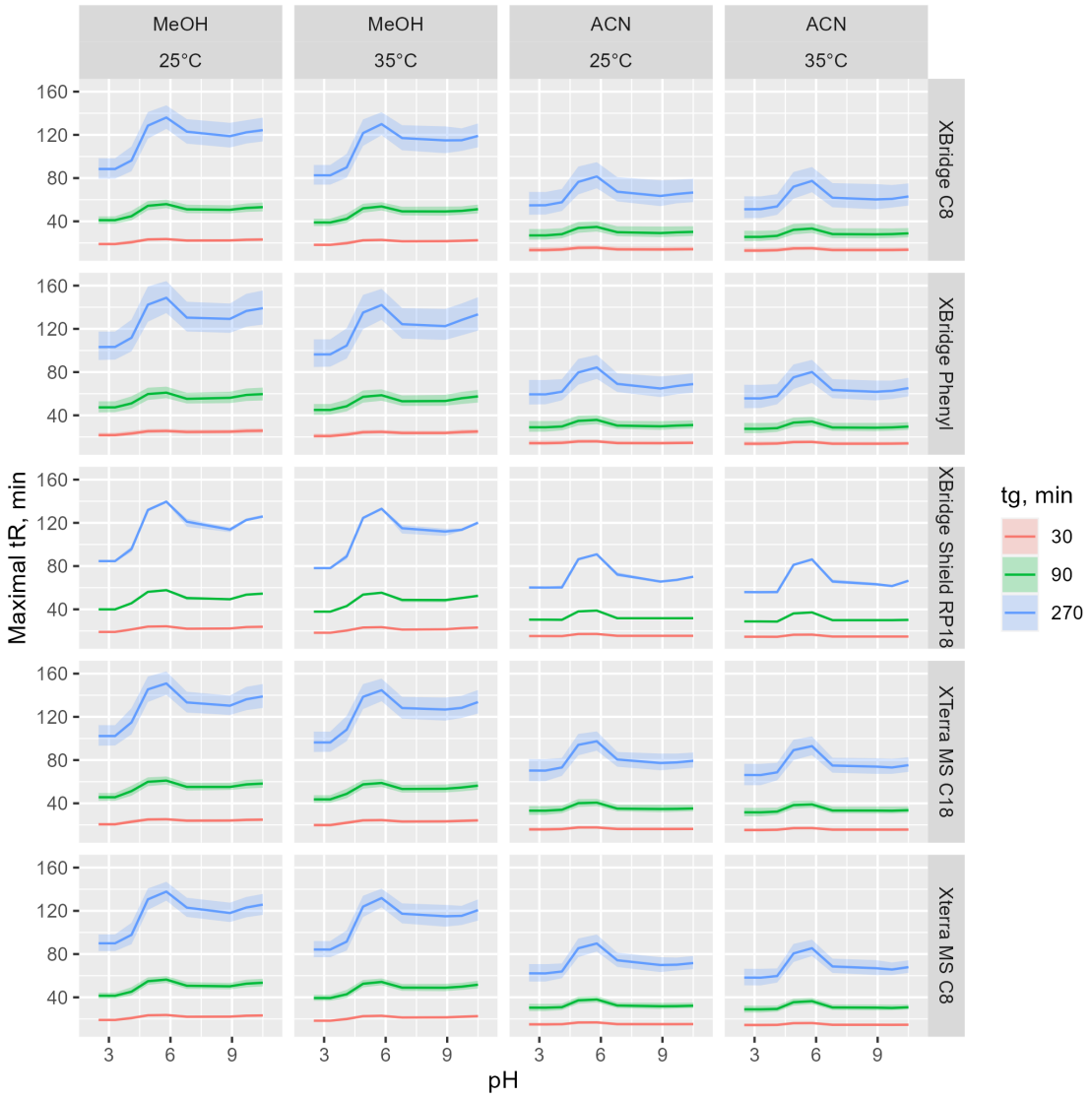


12.3 Decision making

The same utility function was used:


MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Expected Utility
1.5
1.0
0.5
0.0
3
6
9
pH
tg, min
30
90
270


MeOH
25°C
MeOH
35°C
ACN
25°C
ACN
35°C
XBridge C8
XBridge Phenyl
XBridge Shield RP18
XTerra MS C18
Xterra MS C8
Resolution (difference), min
pH
tg, min
30
90
270



1. The best choice for XBridge Shield RP18

```
    tg  fio fik Mod pH     pHo Temp          ColumnName EUtility
158 90 0.05 0.8 ACN  7 8.87588   25 XBridge Shield RP18 1.629025
```

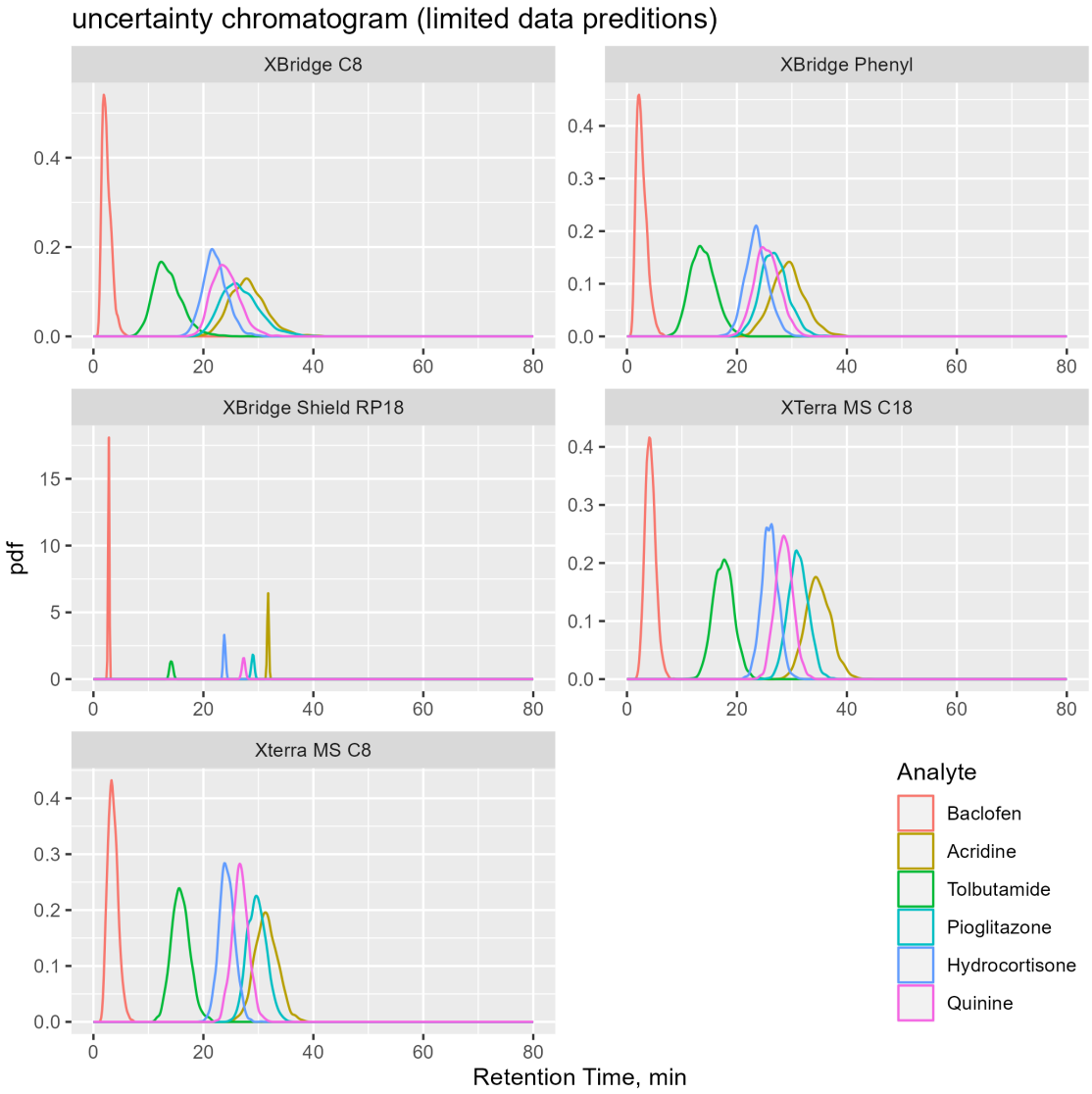
2. The best choice for XTerra MS C18

```
      tg  fio fik Mod pH      pHo Temp    ColumnName EUtility
15404 90 0.05 0.8 ACN  3 4.202902   35 XTerra MS C18 1.347357
```

```
3. The best choice for XBridge Phenyl


      tg  fio fik Mod pH      pHo Temp     ColumnName EUtility
30632 90 0.05 0.8 ACN  3 4.202902   35 XBridge Phenyl  1.66615


4. The best choice for XBridge C8


      tg  fio fik Mod pH      pHo Temp ColumnName EUtility
45860 90 0.05 0.8 ACN  3 4.202902   35 XBridge C8 1.669043


5. The best choice for Xterra MS C8


      tg  fio fik Mod pH      pHo Temp   ColumnName EUtility
61088 90 0.05 0.8 ACN  3 4.202902   35 Xterra MS C8 1.561612
```

12.3.1 Uncertainity chromatogram

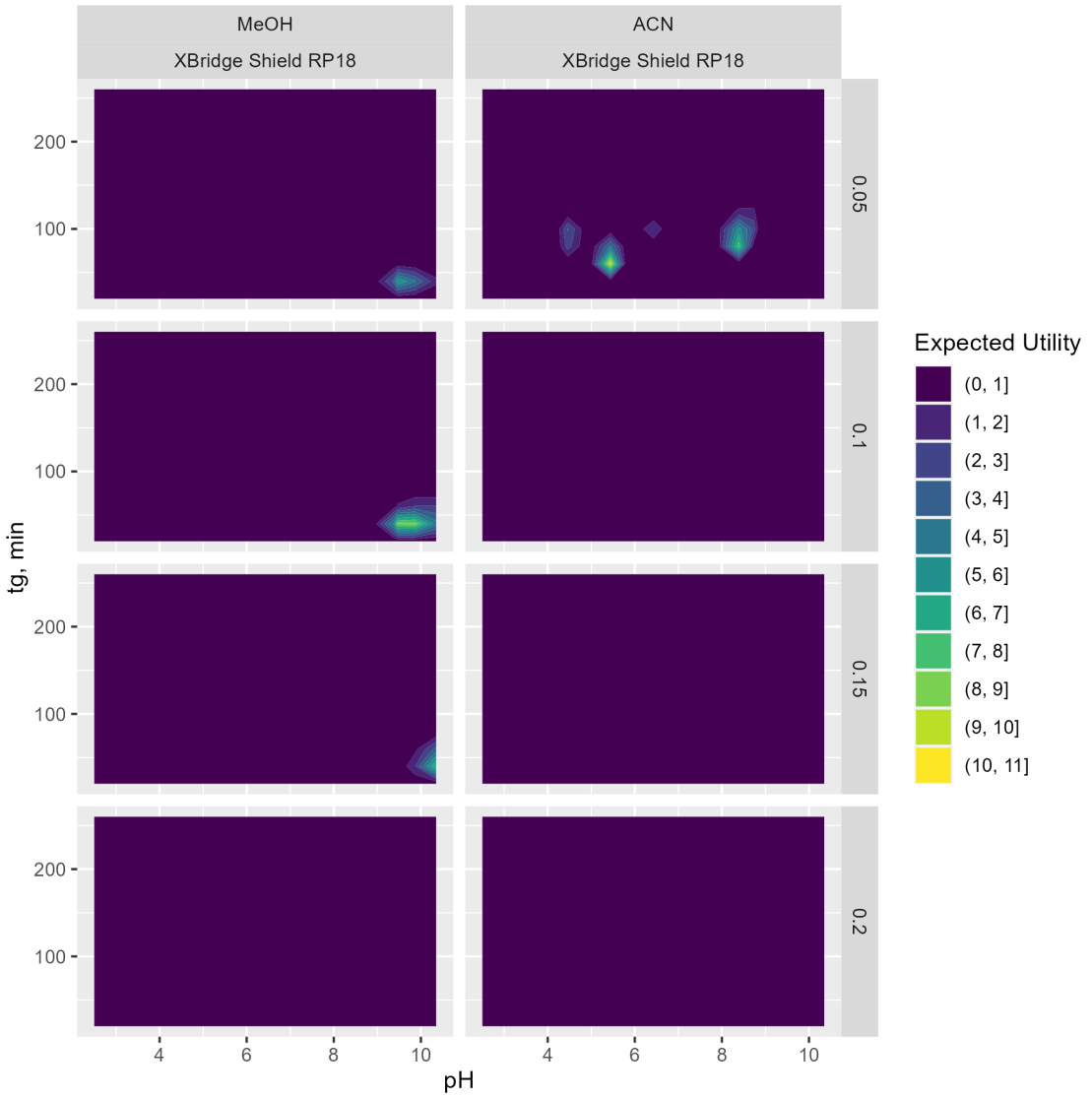
Below is the example of a chromatogram (along with uncertainty) expected for tg = 90 min, pH = 8.9, Temp = 25oC in ACN.

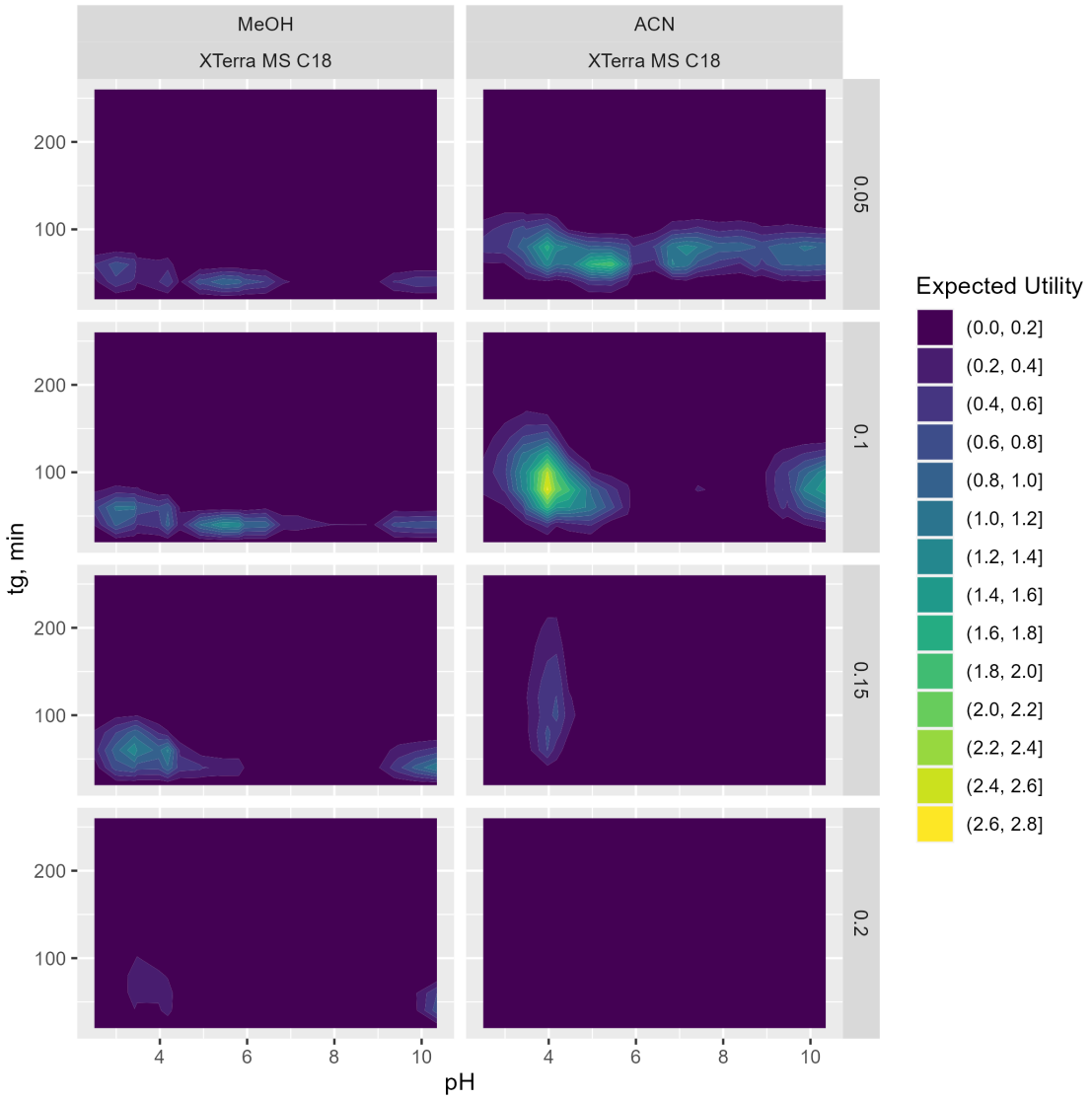


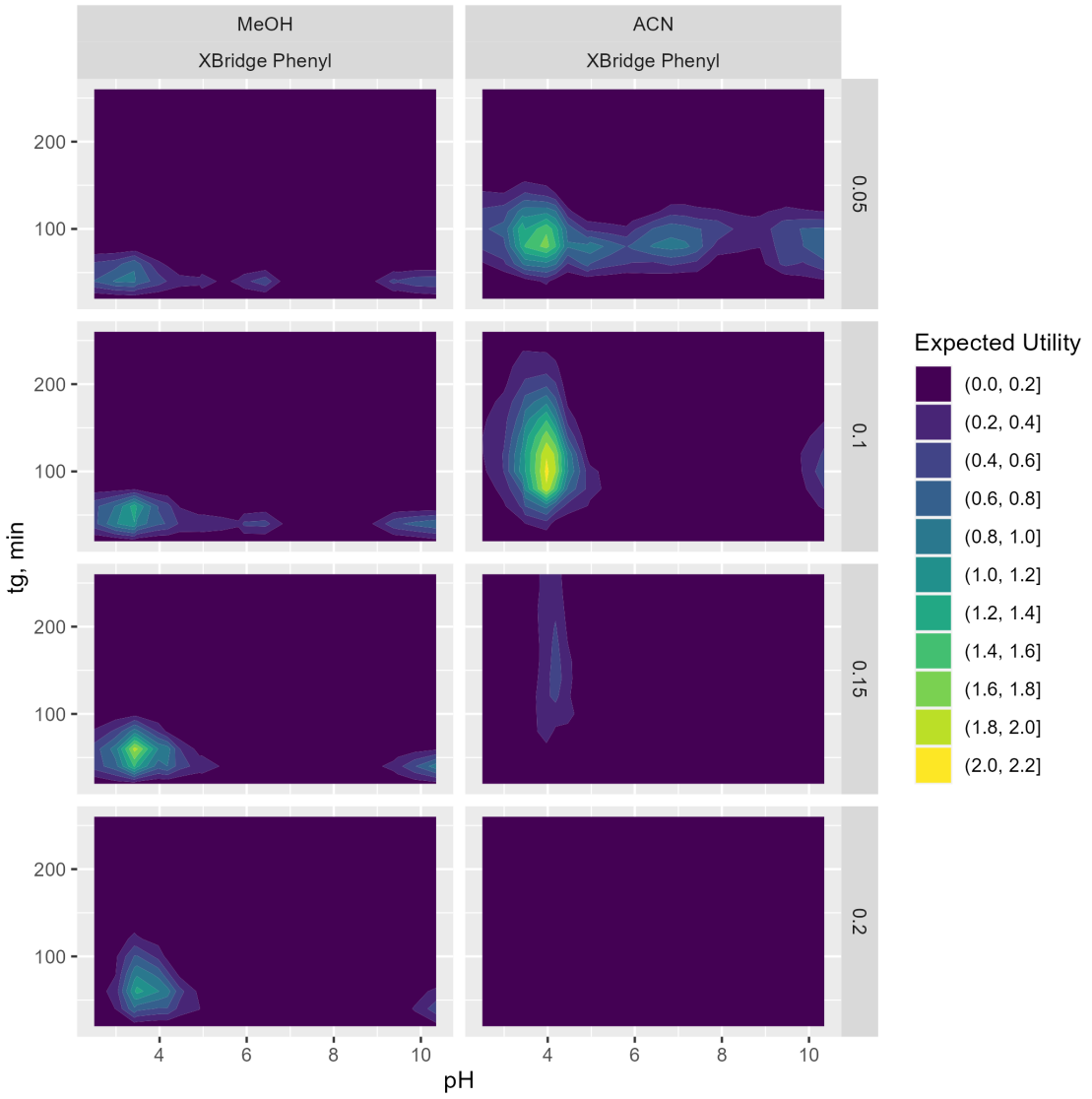
12.4 Utility maps

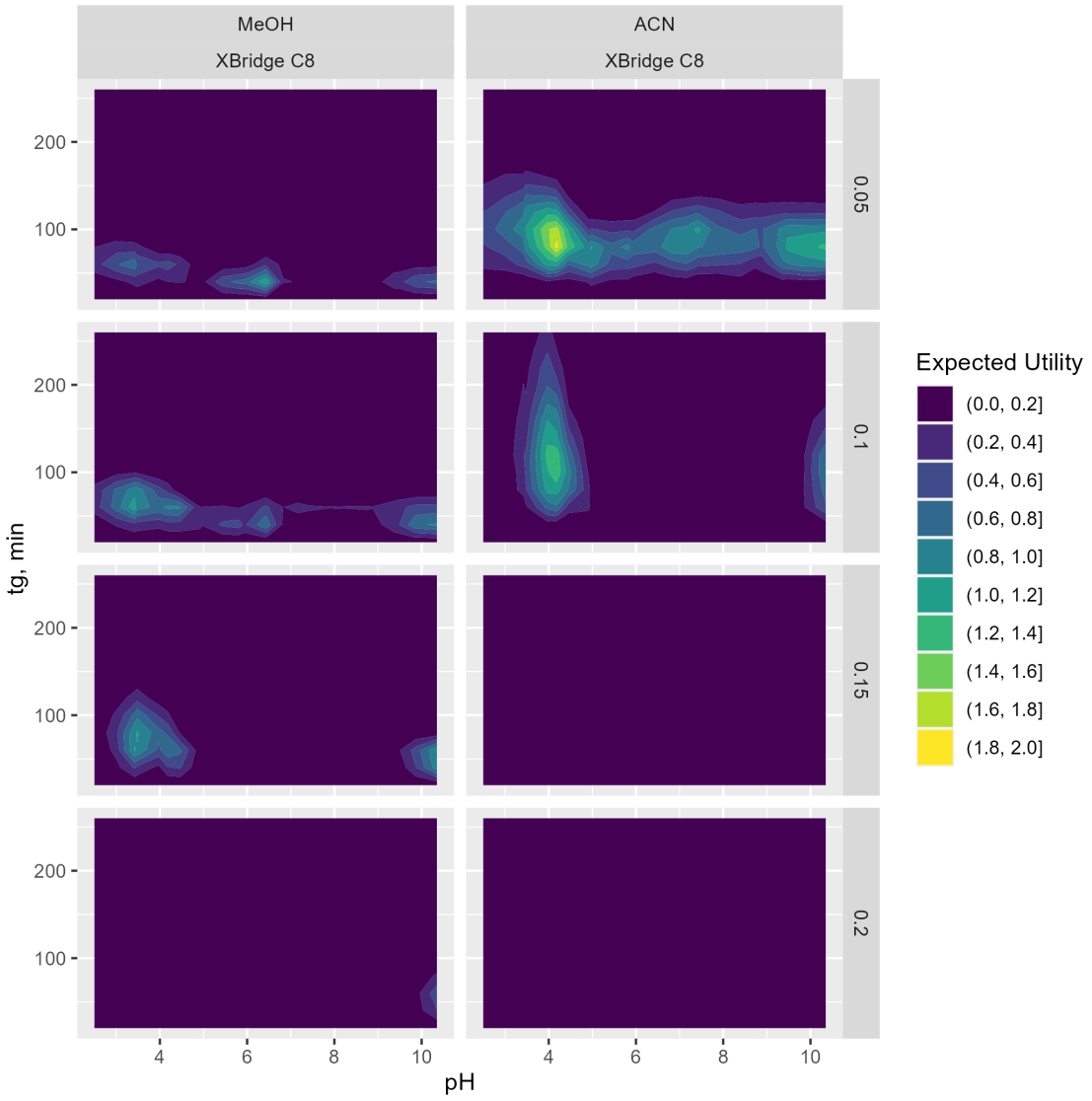
We can also determine more detailed graph presenting relationship between the expected utility and design variables:

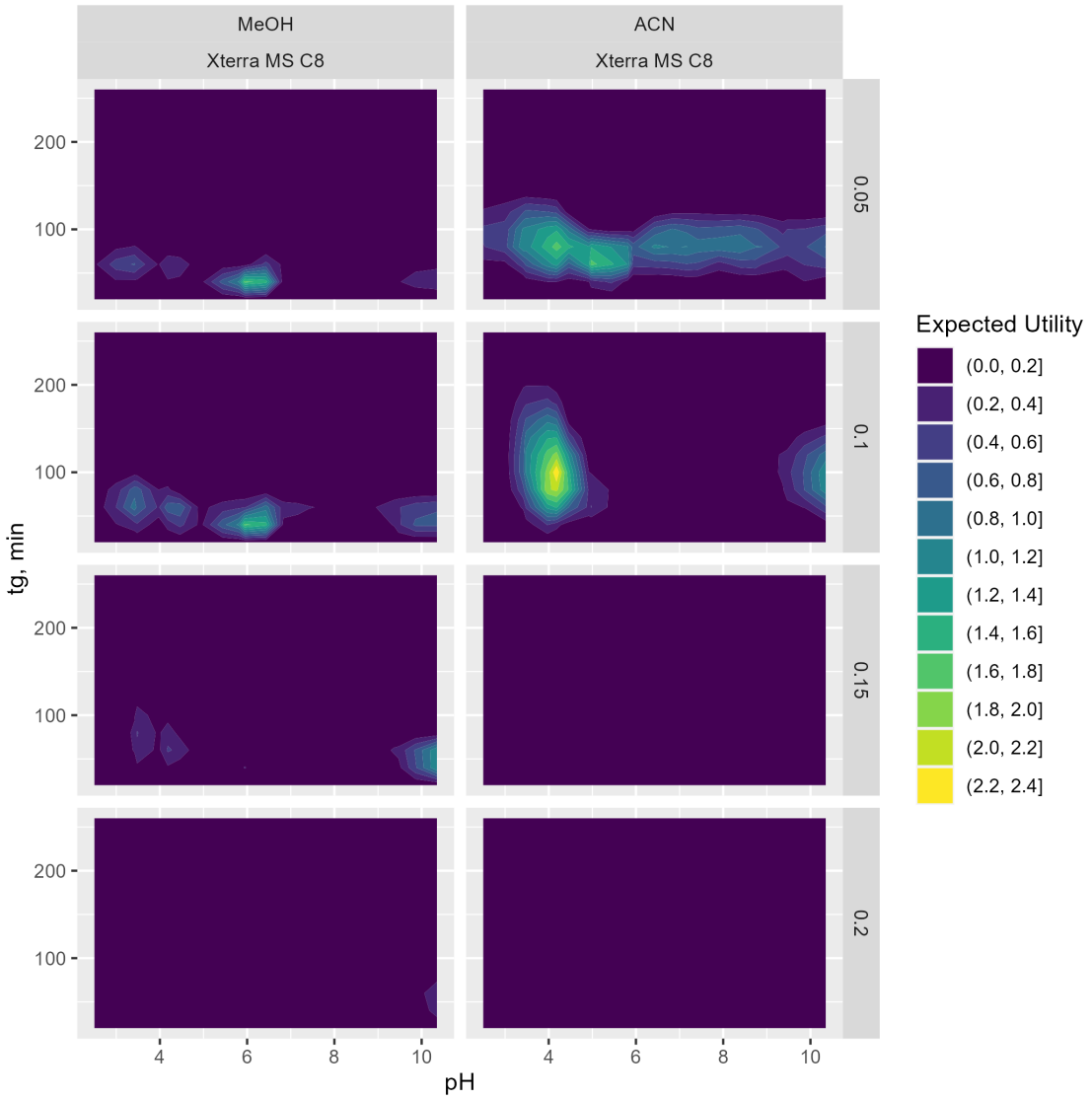
The best chromatogram can be sought based on utility map:


MeOH
XBridge Shield RP18
ACN
XBridge Shield RP18
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0, 1]
(1, 2]
(2, 3]
(3, 4]
(4, 5]
(5, 6]
(6, 7]
(7, 8]
(8, 9]
(9, 10]
(10, 11]


MeOH
XTerra MS C18
ACN
XTerra MS C18
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0.0, 0.2]
(0.2, 0.4]
(0.4, 0.6]
(0.6, 0.8]
(0.8, 1.0]
(1.0, 1.2]
(1.2, 1.4]
(1.4, 1.6]
(1.6, 1.8]
(1.8, 2.0]
(2.0, 2.2]
(2.2, 2.4]
(2.4, 2.6]
(2.6, 2.8]


MeOH
XBridge Phenyl
ACN
XBridge Phenyl
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0.0, 0.2]
(0.2, 0.4]
(0.4, 0.6]
(0.6, 0.8]
(0.8, 1.0]
(1.0, 1.2]
(1.2, 1.4]
(1.4, 1.6]
(1.6, 1.8]
(1.8, 2.0]
(2.0, 2.2]


MeOH
XBridge C8
ACN
XBridge C8
0.05
0.1
0.15
0.2
200
100
tg, min
4
6
8
10
pH
Expected Utility
(0.0, 0.2]
(0.2, 0.4]
(0.4, 0.6]
(0.6, 0.8]
(0.8, 1.0]
(1.0, 1.2]
(1.2, 1.4]
(1.4, 1.6]
(1.6, 1.8]
(1.8, 2.0]




```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName          EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>                  <dbl>
1    60  0.05   0.8 ACN      10  5.50    25 XBridge Shield RP18     10.1
```

```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName    EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>            <dbl>
1    80   0.1   0.8 ACN       5  4.00    25 XTerra MS C18     2.64
```

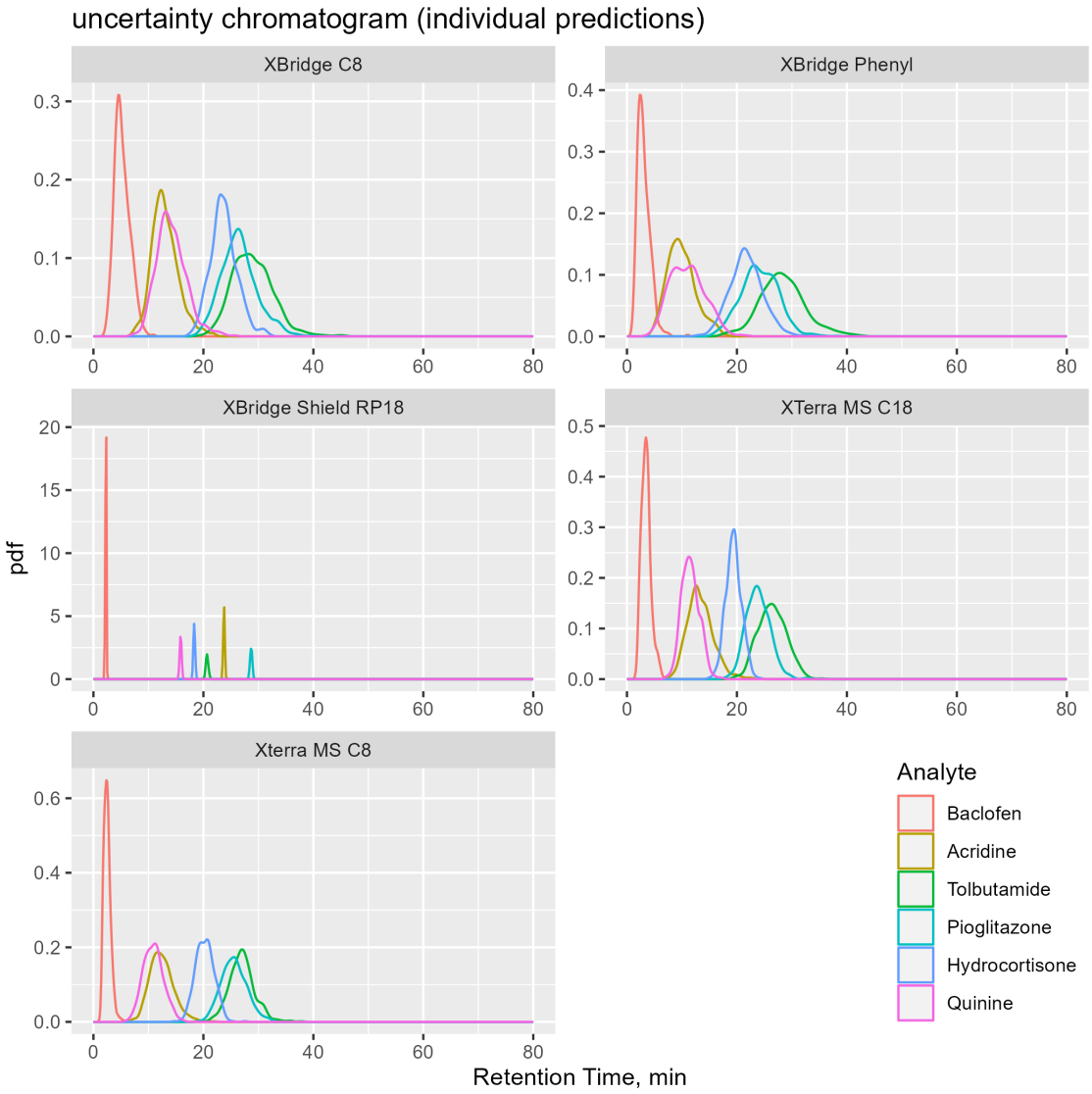
```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName     EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>             <dbl>
1   100   0.1   0.8 ACN       5  4.00    25 XBridge Phenyl     2.06
```

```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>          <dbl>
1    80  0.05   0.8 ACN       6  4.17    25 XBridge C8     1.86
```

```
# A tibble: 1 x 9
     tg   fio   fik Mod    pHid   pHo  Temp ColumnName   EUtility
  <dbl> <dbl> <dbl> <chr> <int> <dbl> <int> <chr>           <dbl>
1   100   0.1   0.8 ACN       6  4.17    25 Xterra MS C8     2.39
```

12.4.1 Uncertainity chromatogram

Let's visualize chromatograms with the highest expected utility given the experimental data.



13 Conclusions

The large gradient retention time data and multilevel modeling allow to characterize separation of RP HPLC stationary phases.

References

Kamedulska, Agnieszka, Łukasz Kubik, Julia Jacyna, Wiktoria Struck-Lewicka, Michał J. Markuszewski, and Paweł Wiczling. 2022. "Toward the General Mechanistic Model of Liquid Chromatographic Retention." *Analytical Chemistry* 94 (31): 11070–80. https://doi.org/10.1021/acs.analchem.2c02034.

Kamedulska, Agnieszka, Łukasz Kubik, and Paweł Wiczling. 2022. "Statistical Analysis of Isocratic Chromatographic Data Using Bayesian Modeling." *Analytical and Bioanalytical Chemistry* 414 (11): 3471–3348. https://doi.org/10.1007/s00216-022-03968-x.

Kubik, Łukasz, Julia Jacyna, Wiktoria Struck-Lewicka, Michał J. Markuszewski, and Paweł Wiczling. 2022a. "LC-TOF-MS Data Collected for 300 Small Molecules. XBridge Shield RP18 Column." *Osf.io/* 1 (1): 1. https://doi.org/10.17605/OSF.IO/ZQTJ7.

———. 2022b. "LC-TOF-MS Data Collected for 300 Small Molecules. XBridge-C8 Column." *Osf.io/* 1 (1): 1. https://doi.org/10.17605/OSF.IO/Y6S8P.

———. 2022c. "LC-TOF-MS Data Collected for 300 Small Molecules. XTerra MS C18 Column." *Osf.io/* 1 (1): 1. https://doi.org/10.17605/OSF.IO/QBV7J.

———. 2022d. "LC-TOF-MS Data Collected for 300 Small Molecules. XTerra-C8 Column." *Osf.io/* 1 (1): 1. https://doi.org/10.17605/OSF.IO/2MCNW.

———. 2022e. "LC-TOF-MS Data Collected for 300 Small Molecules.XBridge Phenyl Column." *Osf.io/* 1 (1): 1. https://doi.org/10.17605/OSF.IO/EVUJ9.

Kubik, Łukasz, Roman Kaliszan, and Paweł Wiczling. 2018. "Analysis of Isocratic-Chromatographic-Retention Data Using Bayesian Multilevel Modeling." *Analytical Chemistry* 90 (22): 13670–79. https://doi.org/10.1021/acs.analchem.8b04033.

Nikitas, P., and A. Pappa-Louisi. 2002. "New Equations Describing the Combined Effect of pH and Organic Modifier Concentration on the Retention in Reversed-Phase Liquid Chromatography." *Journal of Chromatography. A* 971 (1-2): 47–60. https://doi.org/10.1016/s0021-9673(02)00965-2.

Wiczling, Paweł, Agnieszka Kamedulska, and Łukasz Kubik. 2021. "Application of Bayesian Multilevel Modeling in the Quantitative Structure–Retention Relationship Studies of Heterogeneous Compounds." *Analytical Chemistry* 93 (18): 6961–71. https://doi.org/10.1021/acs.analchem.0c05227.

Licenses

Original Computing Environment

```
R version 4.1.3 (2022-03-10)
```

```
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 22621)

Matrix products: default

locale:
[1] LC_COLLATE=Polish_Poland.1250  LC_CTYPE=Polish_Poland.1250
[3] LC_MONETARY=Polish_Poland.1250 LC_NUMERIC=C
[5] LC_TIME=Polish_Poland.1250

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] kableExtra_1.3.4 GGally_2.1.2     posterior_1.3.1  bayesplot_1.9.0
 [5] reshape2_1.4.4   knitr_1.40       cmdstanr_0.5.3   gridExtra_2.3
 [9] ggplot2_3.4.2    dplyr_1.1.2      pracma_2.3.8

loaded via a namespace (and not attached):
 [1] Rcpp_1.0.9           lattice_0.20-45      svglite_2.1.1
 [4] tidyr_1.3.0          zoo_1.8-11           digest_0.6.29
 [7] utf8_1.2.2           R6_2.5.1             plyr_1.8.7
[10] ggridges_0.5.3       backports_1.4.1      coda_0.19-4
[13] evaluate_0.16        httr_1.4.5           pillar_1.9.0
[16] tidybayes_3.0.2      rlang_1.1.0          data.table_1.14.2
[19] rstudioapi_0.14      checkmate_2.1.0      rmarkdown_2.16
[22] textshaping_0.3.6    labeling_0.4.2       webshot_0.5.4
[25] stringr_1.5.0        munsell_0.5.0        compiler_4.1.3
[28] xfun_0.32            pkgconfig_2.0.3      systemfonts_1.0.4
[31] htmltools_0.5.3      tidyselect_1.2.0     tibble_3.2.1
[34] tensorA_0.36.2       arrayhelpers_1.1-0   matrixStats_0.62.0
[37] codetools_0.2-18     reshape_0.8.9        fansi_1.0.3
[40] viridisLite_0.4.1    withr_2.5.0          ggdist_3.2.0
[43] grid_4.1.3           distributional_0.3.1 jsonlite_1.8.4
[46] gtable_0.3.1         lifecycle_1.0.3      magrittr_2.0.3
[49] scales_1.2.1         cli_3.4.0            stringi_1.7.8
[52] farver_2.1.1         xml2_1.3.3           ragg_1.2.5
[55] generics_0.1.3       vctrs_0.6.2          RColorBrewer_1.1-3
[58] tools_4.1.3          svUnit_1.0.6         glue_1.6.2
[61] purrr_1.0.1          abind_1.4-5          fastmap_1.1.0
[64] yaml_2.3.5           colorspace_2.0-3     isoband_0.2.5
[67] rvest_1.0.3
```