vs_sup

drsimonturega

2025-12-16

Virtual screening for high affinity guests for synthetic supramolecular receptors

Orginal literature

https://doi.org/10.1039/C5SC00534E

Function used in this analysis

I don't have a R data science package(library) yet...

Replication of our analysis in the manuscript using python

We load cleaned data from the supporting information

```
df = read.csv("tab_gold_wt.csv", header = TRUE)
```

Run some exploratory data analysis

Data summary

```
summary(df)
```

```
##      Guest     Ligand_clash     Ligand_torsion    Part_buried
##  Min.   : 1   Min.   :0.00000   Min.   :0.0000   Min.   :-4.621
##  1st Qu.:18   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:-3.046
##  Median :35   Median :0.00000   Median :0.0000   Median :-2.193
##  Mean   :35   Mean   :0.02582   Mean   :0.0650   Mean   :-2.027
##  3rd Qu.:52   3rd Qu.:0.00000   3rd Qu.:0.0016   3rd Qu.:-1.708
##  Max.   :69   Max.   :1.78130   Max.   :1.0855   Max.   : 8.908
##    Non.polar      Ligand_flexibility    logKexp
##  Min.   :-72.20   Min.   :0.0000     Min.   :-1.000
##  1st Qu.:-49.49   1st Qu.:0.0000     1st Qu.: 1.860
##  Median :-40.64   Median :0.0000     Median : 3.600
##  Mean   :-42.43   Mean   :0.7971     Mean   : 3.167
##  3rd Qu.:-34.32   3rd Qu.:1.0000     3rd Qu.: 4.300
##  Max.   :-22.77   Max.   :7.0000     Max.   : 8.000
```

we still have guest number the data frame we will remove it

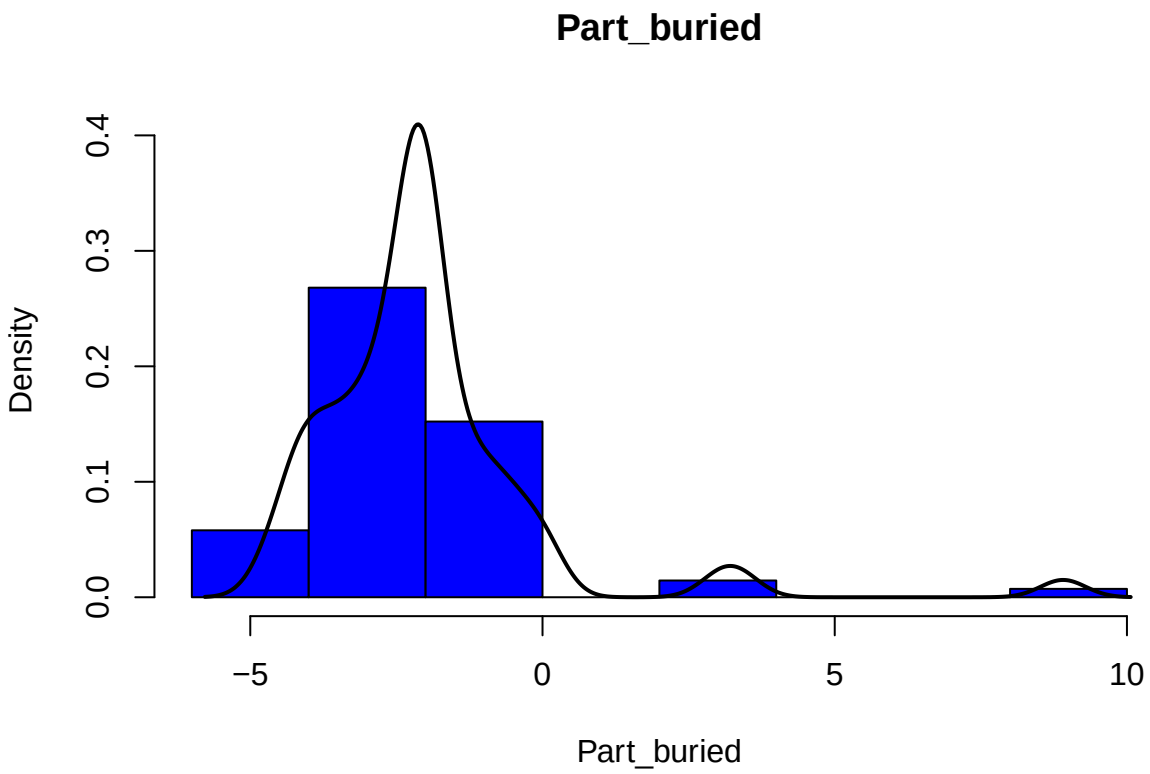
```
df <- df[-c(1)]
summary(df)
```

```
##   Ligand_clash     Ligand_torsion    Part_buried       Non.polar
##  Min.   :0.00000   Min.   :0.0000   Min.   :-4.621   Min.   :-72.20
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:-3.046   1st Qu.:-49.49
##  Median :0.00000   Median :0.0000   Median :-2.193   Median :-40.64
##  Mean   :0.02582   Mean   :0.0650   Mean   :-2.027   Mean   :-42.43
##  3rd Qu.:0.00000   3rd Qu.:0.0016   3rd Qu.:-1.708   3rd Qu.:-34.32
##  Max.   :1.78130   Max.   :1.0855   Max.   : 8.908   Max.   :-22.77
##  Ligand_flexibility    logKexp
##  Min.   :0.0000     Min.   :-1.000
##  1st Qu.:0.0000     1st Qu.: 1.860
##  Median :0.0000     Median : 3.600
##  Mean   :0.7971     Mean   : 3.167
##  3rd Qu.:1.0000     3rd Qu.: 4.300
##  Max.   :7.0000     Max.   : 8.000
```

Histogram with KDE line for clarity

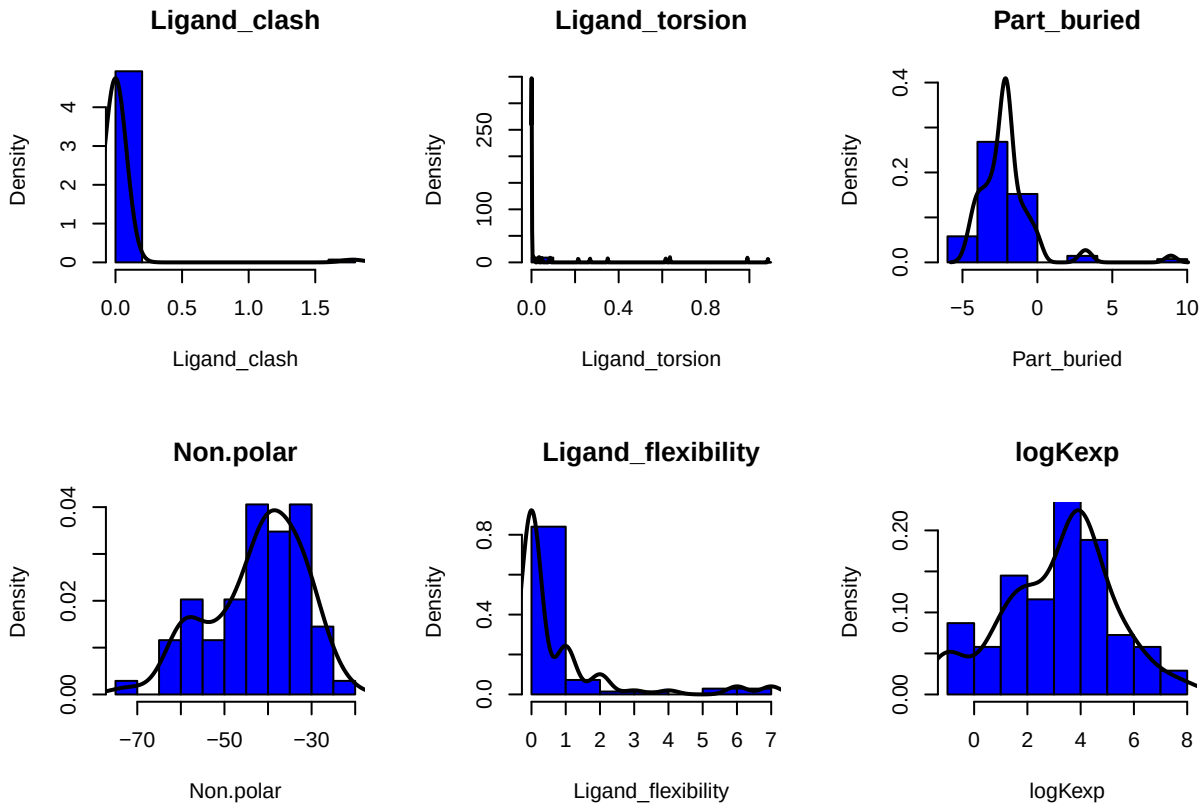
The histogram is in blue bars. The Kernel Density Estimation line is and estimation of the histogram represented as a continuous line and is shown as a black line.

```
hist_kde(df, "Part_buried")
```



Its helpful to plot all dataframe columns

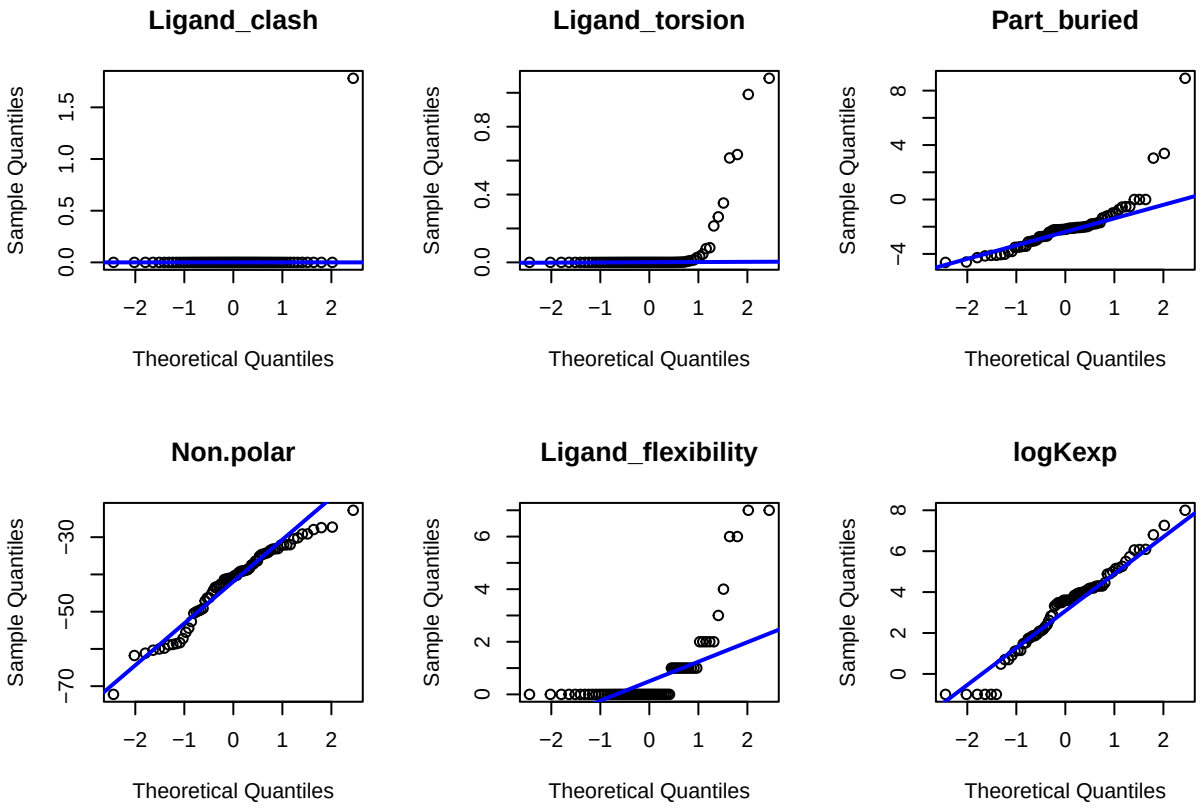
```
multi_plot(df, hist_kde)
```



Quantile quantile plots

Our quantile quantile plots show us weather our data fits to a theoretical distribution

```
multi_plot(df, q_q_plot)
```



This is experimental data from 69 individual data points our team generated, so we don't really expect it to fit a normal distribution. We would expect the replicates of **logKexp** to fit a normal distribution and the error quoted in the manuscript is at 95% confidence.

Correlation matrix heatmap

Over correlated molecular descriptor (GoldPLP functions) columns can cause problems with our regression models. Too many correlated models cause a misrepresentation of those molecular descriptors in our regression model.

```
cor_mat <- round(cor(df),2)
head(cor_mat)
```

```
##                    Ligand_clash Ligand_torsion Part_buried Non.polar
## Ligand_clash               1.00          -0.02        0.68     -0.19
## Ligand_torsion            -0.02           1.00        0.15     -0.24
## Part_buried                0.68           0.15        1.00     -0.61
## Non.polar                 -0.19          -0.24       -0.61      1.00
## Ligand_flexibility         0.02           0.36       -0.01     -0.26
## logKexp                   -0.25          -0.06        0.20     -0.46
##                    Ligand_flexibility logKexp
## Ligand_clash                     0.02   -0.25
## Ligand_torsion                   0.36   -0.06
## Part_buried                     -0.01    0.20
## Non.polar                       -0.26   -0.46
```

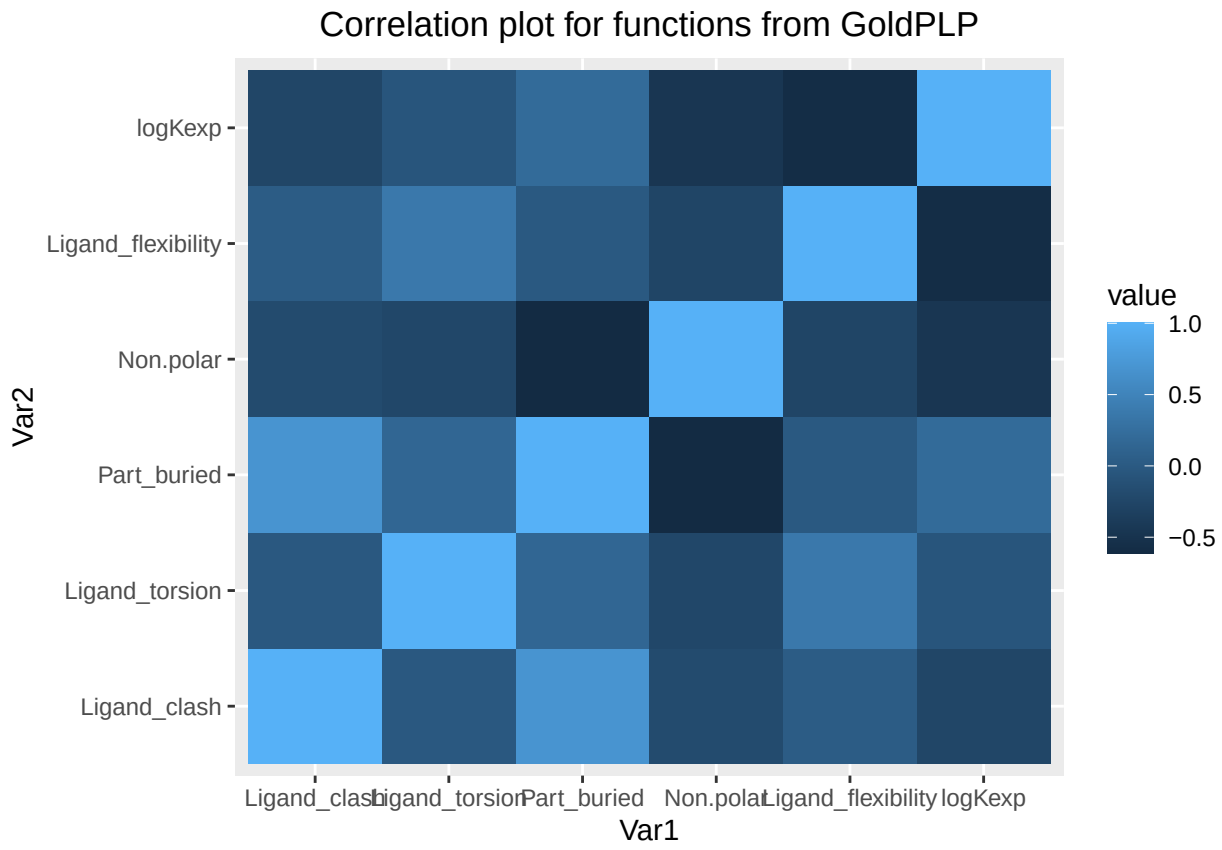
```
## Ligand_flexibility                1.00   -0.58
## logKexp                          -0.58    1.00
```

Reshape our dataframe

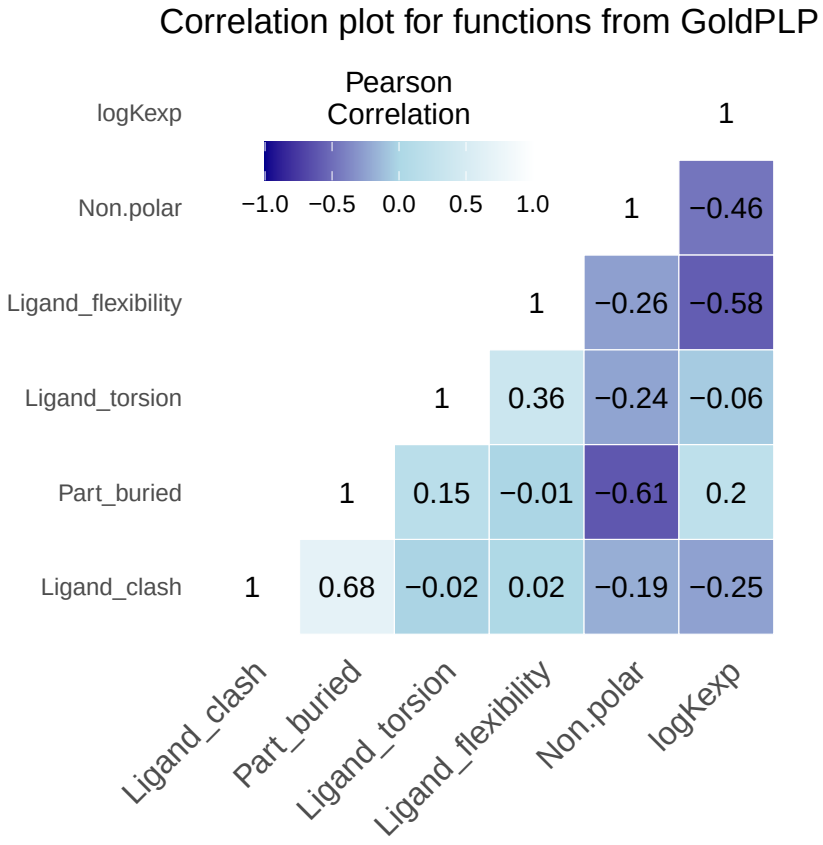
```
melted_cor_mat <- melt(cor_mat)
head(melted_cor_mat)
```

```
##                 Var1         Var2 value
## 1       Ligand_clash Ligand_clash  1.00
## 2     Ligand_torsion Ligand_clash -0.02
## 3        Part_buried Ligand_clash  0.68
## 4          Non.polar Ligand_clash -0.19
## 5 Ligand_flexibility Ligand_clash  0.02
## 6            logKexp Ligand_clash -0.25
```

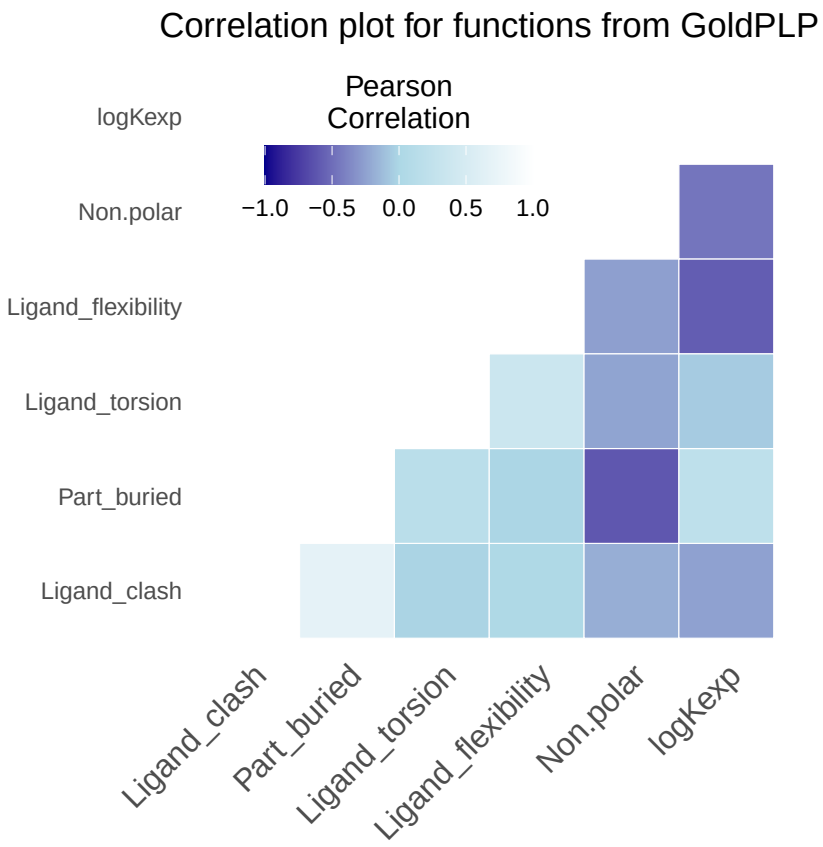
```
ggplot(data = melted_cor_mat, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  ggtitle("Correlation plot for functions from GoldPLP") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
correl_mat_plot_2(df, "Correlation plot for functions from GoldPLP")
```




```
correl_mat_plot_3(df, "Correlation plot for functions from GoldPLP")
```



You can make your correlation plot neater but they are not really pretty.

Replication of our analysis in the manuscript using R

We take a set df cleaned and prepared in the python verson of this lecture.

We load cleaned data from the supporting information

```
df_chemplp_score = read.csv("chemplp_score.csv", header = TRUE)
dim(df_chemplp_score)
```

```
## [1] 54  3
```

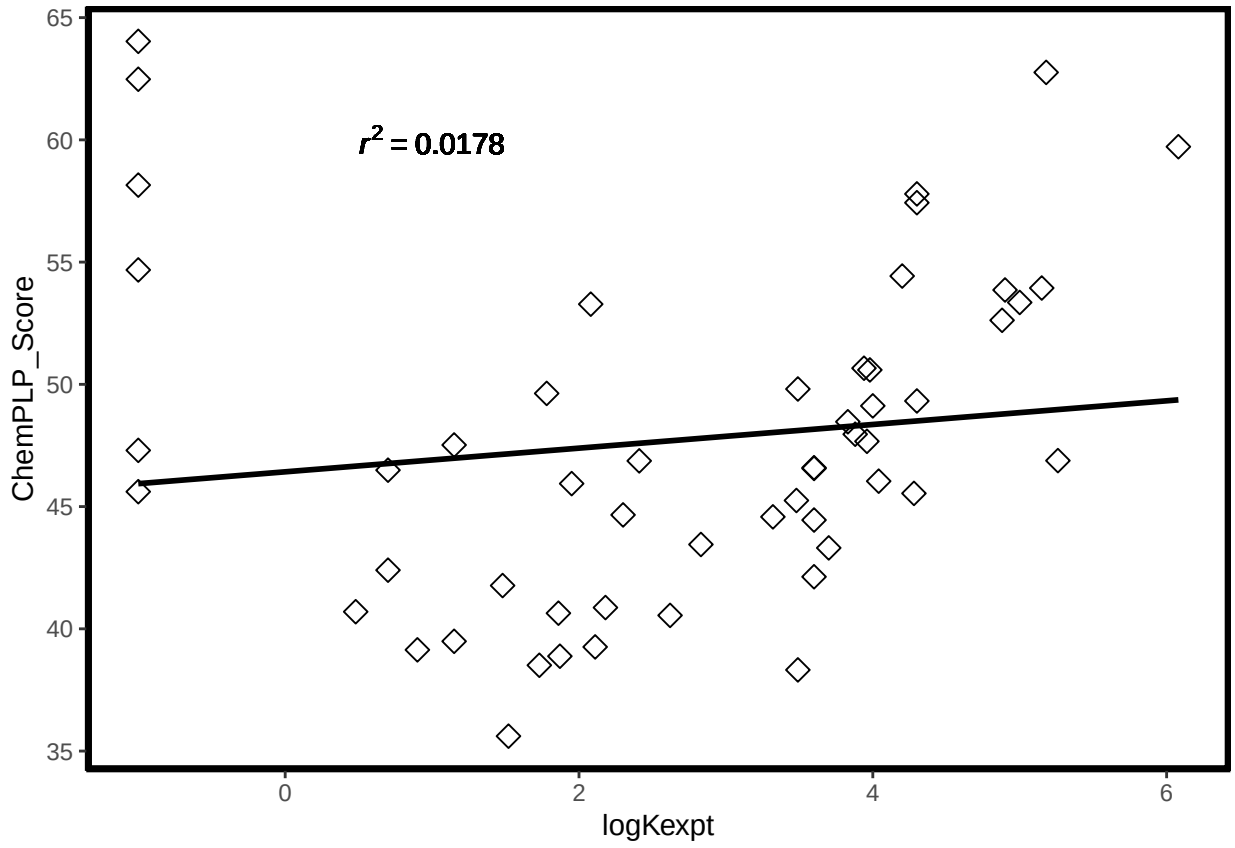
Comparision of logKexp and ChemPLP_Score for training set from the manuscript

```
corr_man = lm(ChemPLP_Score~logKexpt, data = df_chemplp_score)

summary(corr_man)$r.squared
```

```
## [1] 0.01782835
```

```r
lm_rsq <- function(mod){
    m <- mod;
    rsq <- substitute(italic(r)^2~"="~r2,
         list(r2 = format(summary(m)$r.squared, digits = 3)))
    as.character(as.expression(rsq));
}



ggplot(data = df_chemplp_score, aes(x=logKexpt, y=ChemPLP_Score)) +
  geom_point(size=3, shape=23) +
  theme(axis.line = element_line(colour = "black"),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_rect(color = 'black',
                                fill = NA,
                                size = 2),
    panel.background = element_blank()) +
  geom_smooth(method = "lm", se=FALSE, color="black", formula = y ~ x) +
  geom_text(x = 1, y = 60, label = lm_rsq(corr_man), parse = TRUE)
```

```
## Warning: The ‘size‘ argument of ‘element_rect()‘ is deprecated as of ggplot2 3.4.0.
## i Please use the ‘linewidth‘ argument instead.
## This warning is displayed once per session.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```




```
#print(lm_rsq(corr_man))
```

Multivariate linear regression on the training set from the manuscript using eq. 2

Slice our data set

```
df_train <- df[1:54,]
df_test <- df[55:69,]
print("Dimensions of training set")
```

```
## [1] "Dimensions of training set"
```

```
print(dim(df_train))
```

```
## [1] 54  6
```

```
print("Dimensions of test set")
```

```
## [1] "Dimensions of test set"
```

```
print(dim(df_test))
```

```
## [1] 15  6
```

```
summary(df_train)
```

```
##   Ligand_clash     Ligand_torsion      Part_buried       Non.polar
##  Min.   :0.00000   Min.   :0.000000   Min.   :-4.621   Min.   :-60.02
##  1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:-3.078   1st Qu.:-45.14
##  Median :0.00000   Median :0.000000   Median :-2.224   Median :-39.06
##  Mean   :0.03299   Mean   :0.061415   Mean   :-2.349   Mean   :-39.97
##  3rd Qu.:0.00000   3rd Qu.:0.004375   3rd Qu.:-2.048   3rd Qu.:-33.13
##  Max.   :1.78130   Max.   :0.990400   Max.   : 8.908   Max.   :-22.77
##  Ligand_flexibility    logKexp
##  Min.   :0.0000     Min.   :-1.000
##  1st Qu.:0.0000     1st Qu.: 1.573
##  Median :0.0000     Median : 3.400
##  Mean   :0.8704     Mean   : 2.688
##  3rd Qu.:1.0000     3rd Qu.: 3.995
##  Max.   :7.0000     Max.   : 6.080
```

```
eq2_lm = lm(logKexp~Ligand_clash + Ligand_torsion + Part_buried + Non.polar,data = df_train)
summary(eq2_lm)
```

```
##
## Call:
## lm(formula = logKexp ~ Ligand_clash + Ligand_torsion + Part_buried +
##     Non.polar, data = df_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.0396 -0.7326  0.6488  1.1804  2.1094
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.1812     2.0172   0.586   0.5609
## Ligand_clash    -4.2954     2.1439  -2.004   0.0507 .
## Ligand_torsion  -2.3586     1.4062  -1.677   0.0998 .
## Part_buried      0.2338     0.3260   0.717   0.4767
## Non.polar       -0.0586     0.0365  -1.605   0.1148
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.743 on 49 degrees of freedom
## Multiple R-squared:  0.1994, Adjusted R-squared:  0.134
## F-statistic:  3.05 on 4 and 49 DF,  p-value: 0.02535
```

```
logKcalc = predict(eq2_lm, select(df_train,c(1:5)))
```

```
#print(logKcalc)

df_train["logKcalc"] = logKcalc

df_train["test"] = df_train["logKcalc"] - df_train["logKexp"]
```

```r
#rmse_value <- sqrt(mean((df_train["logKexp"] - df_train["logKcalc"])^2))

library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

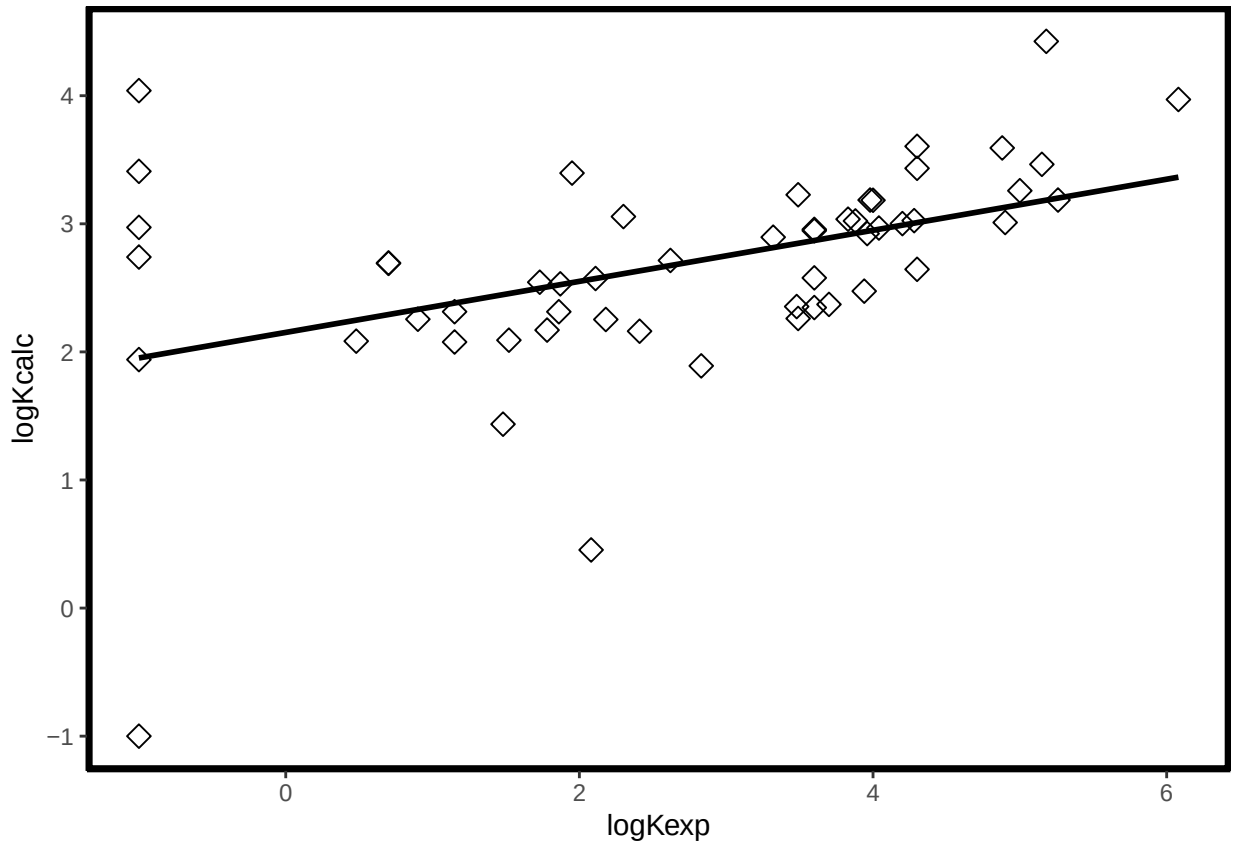
```
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
rmse_value <- RMSE(df_train$logKcalc, df_train$logKexp)

print(rmse_value)
```

```
## [1] 1.659942
```

```r
#print(df_train)
```

```r
#install.packages("caret")
#library(caret)

#rmse_value <- RMSE(logKcalc, logKexpt)

#print(df_train["logKexp"])

#rmse_value <- sqrt(mean((df_train["logKexp"] - df_train["logKcalc"])^2))

#print(rmse_value)

ggplot(data = df_train, aes(x=logKexp, y=logKcalc)) +
  geom_point(size=3, shape=23) +
  theme(axis.line = element_line(colour = "black"),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_rect(color = 'black',
                                fill = NA,
                                size = 2),
    panel.background = element_blank()) +
  geom_smooth(method = "lm", se=FALSE, color="black", formula = y ~ x) +
  geom_text(x = 1, y = -2, label = rmse_value, parse = TRUE)
```




```
paste("rmse = ", rmse_value)
```

```
## [1] "rmse =  1.65994233649364"
```