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

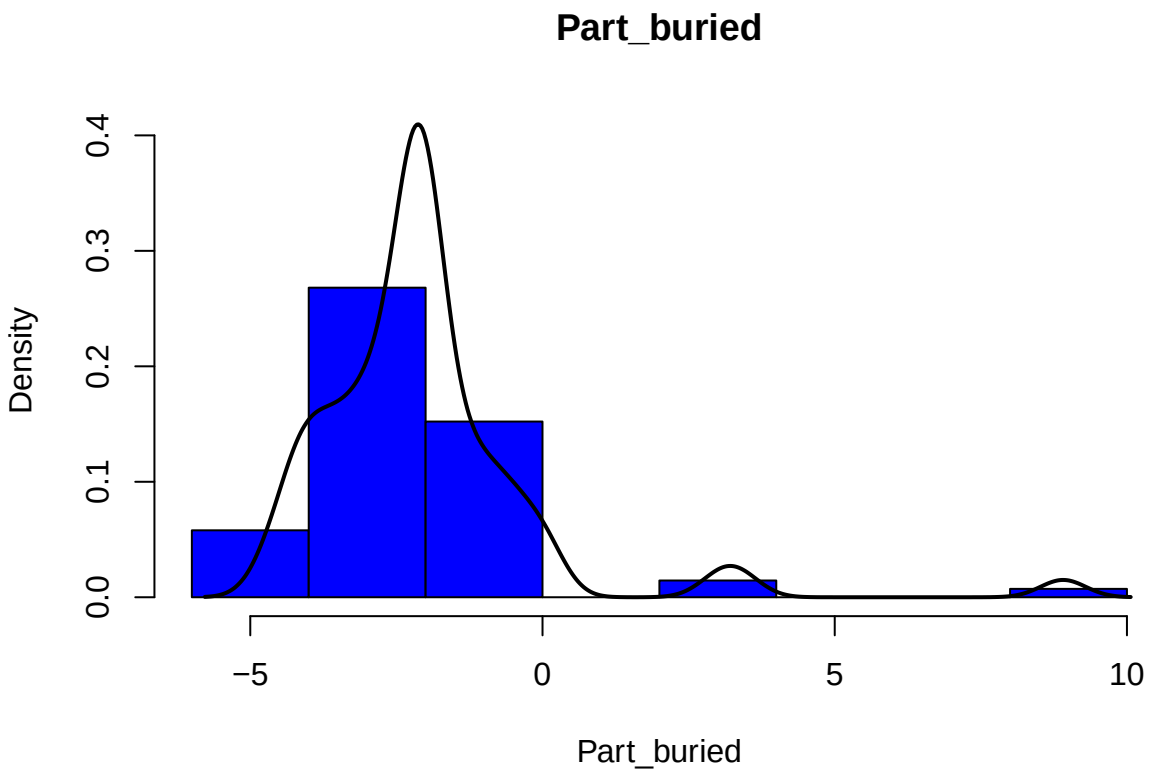
```
df <- df[-c(1)]
summary(df)
```

```
##   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

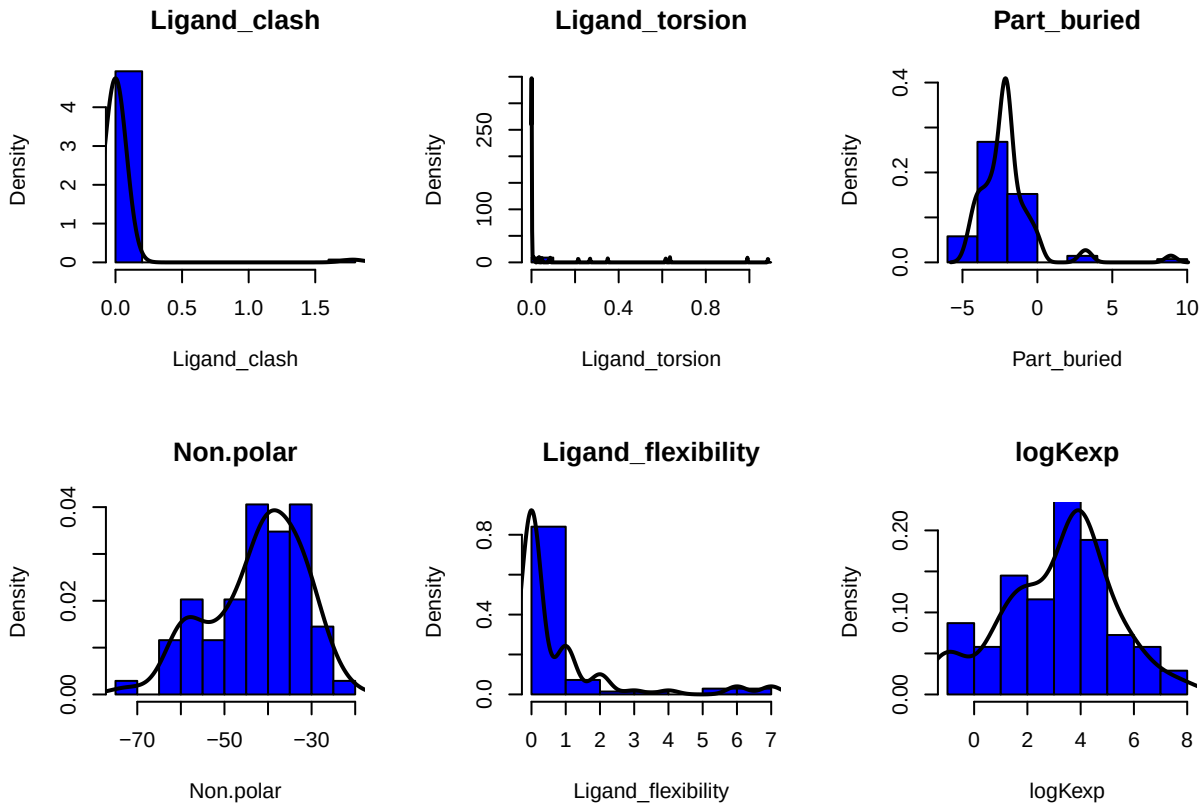
The histogram is in blue bars. The Kernel Density Estimation line is and estimation of the histogram represented as a continuous line and is shown as a black line.

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



Its helpful to plot all dataframe columns

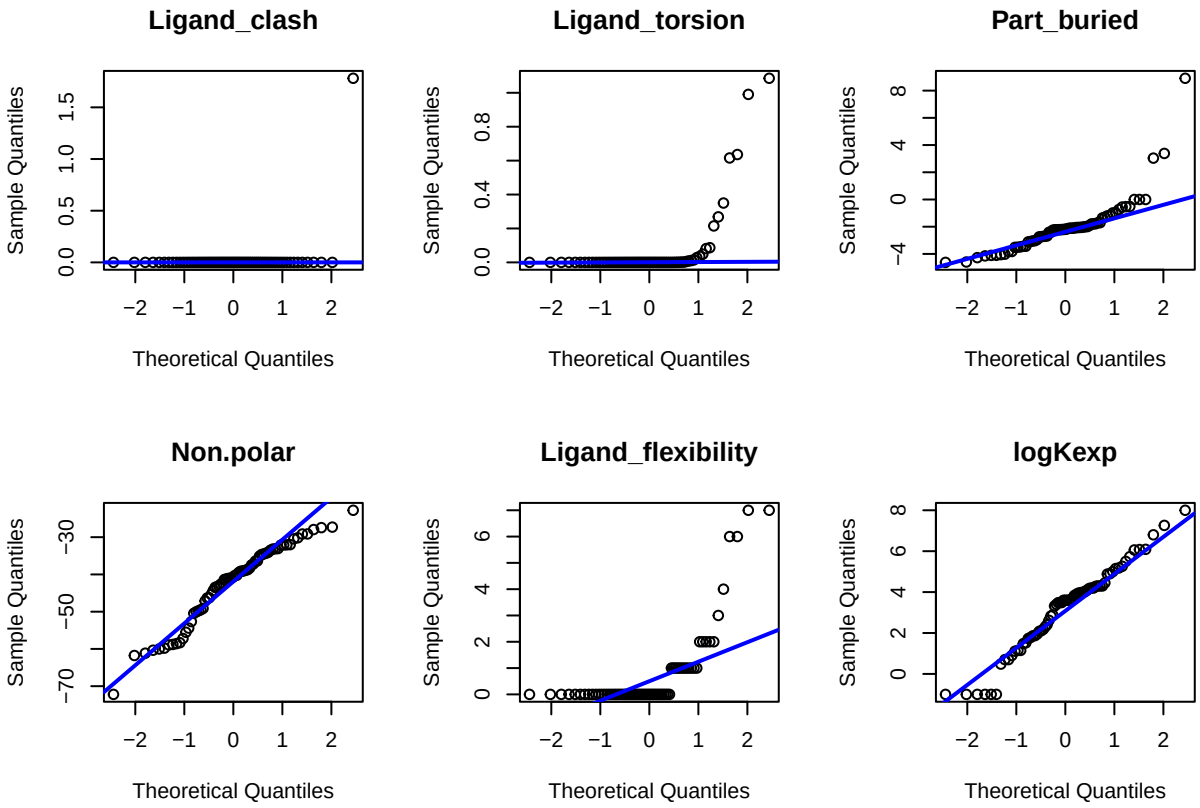
```
multi_plot(df, hist_kde)
```



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
```

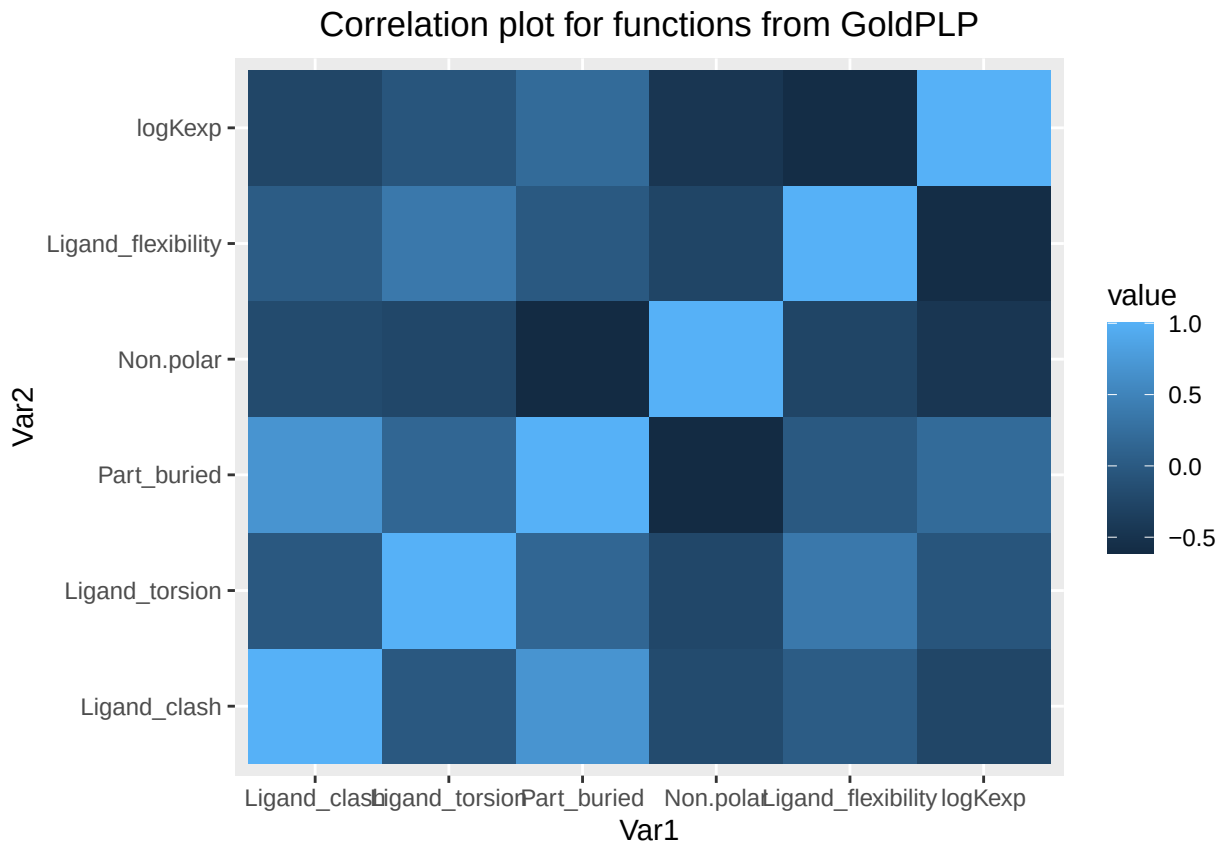
```
## Ligand_flexibility                1.00   -0.58
## logKexp                          -0.58    1.00
```

Reshape our dataframe

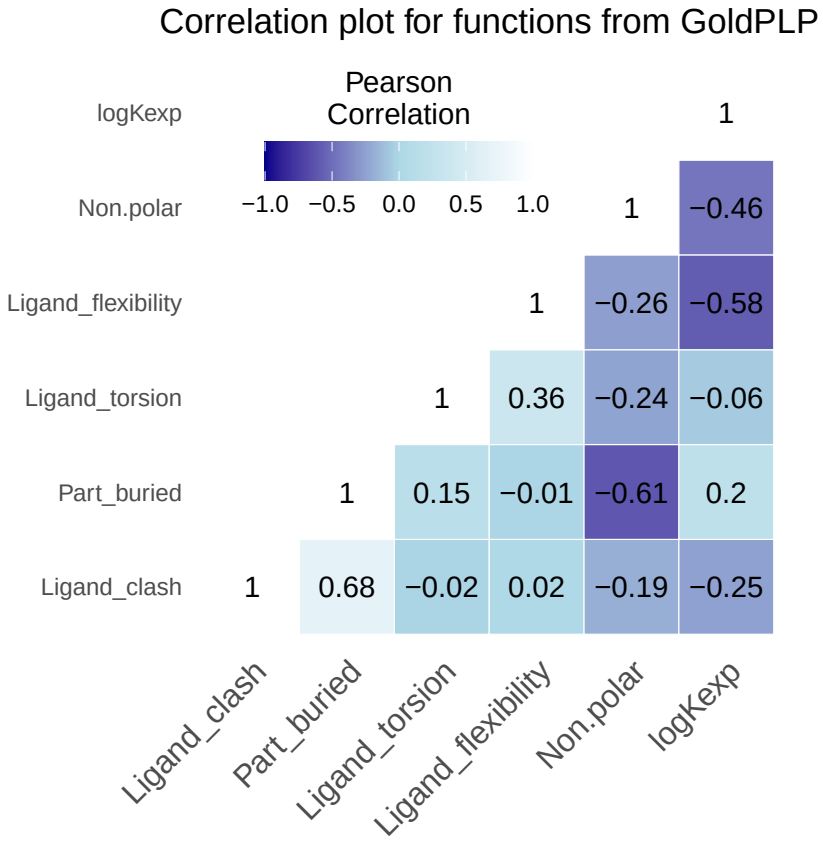
```
melted_cor_mat <- melt(cor_mat)
head(melted_cor_mat)
```

```
##                 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
```

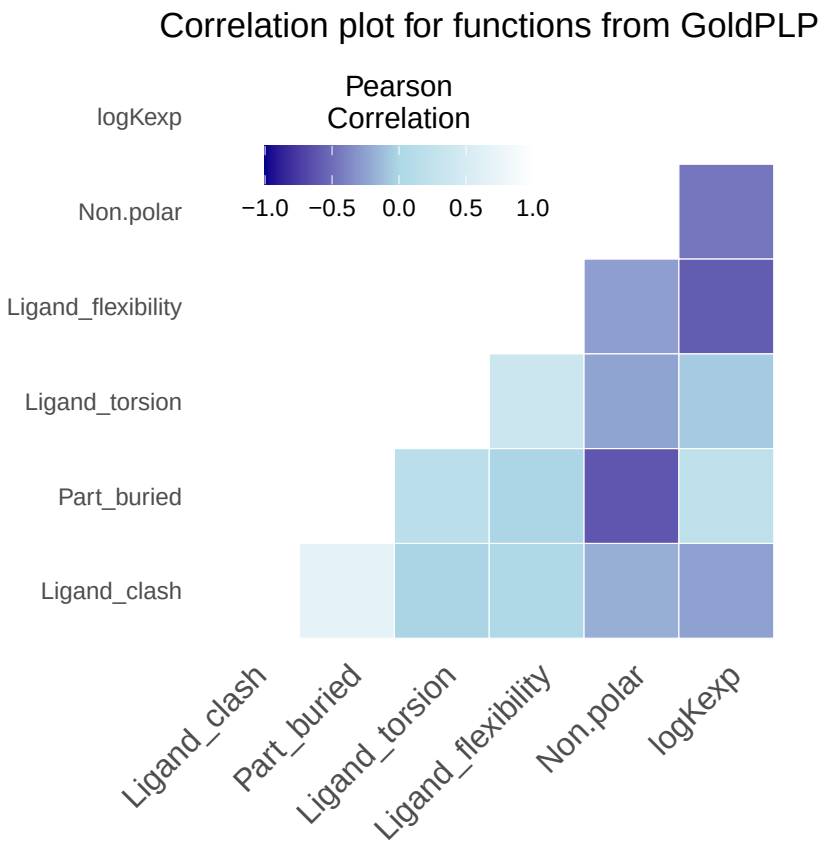
```
ggplot(data = melted_cor_mat, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  ggtitle("Correlation plot for functions from GoldPLP") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
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.