Fathead Minnow LC50 Concentration

Ian Brain

4/27/2022

Introduction

The purpose of this report is to perform basic exploratory data analysis on how six molecular variables are associated with levels of LC50. We are examining the fish toxicity data set which measures the concentration of chemicals that cause death in 50% of fathead minnows. The data set can be found at this link: https://archive.ics.uci.edu/ml/datasets/QSAR+fish+toxicity

There are six descriptors in this data set and one response variable. First, CIC0 contains complementary information content with neighborhood symmetry of 0 order. Secondly, SM1_Dz contains 2d matrix based descriptors. Moreover, GATS1i represents the geary autocorrelation of lag 1 weighted by ionization potential. Furthermore, NdsCH and NdssC are categorical variables that contain atom-type counts. MLOGP is a molecular property factor and LC50 is the response which is the concentration of chemicals that cause death in 50% of test fish over a test duration of 96 hours.

This report uses two packages. The first is the tidyverse package, a collection of R packages that are similar and designed to work together. The second package is GGally which is similar to ggplot2 but allows us to create more plots including the pairs plot.

```
library(tidyverse)
library(GGally)
```

First the data set is read in and assigned to the fishData object.

```
fishData <- read_csv2("/Users/ianbrain/Downloads/rstudio/qsar_fish_toxicity.csv", col_names = FALSE)
```

The columns of fishData are renamed to the molecular descriptors and quantitative response.

```
fishData <- fishData %>%
  rename("CIC0" = X1,
         "SM1_Dz" = X2,
         "GATS1i" = X3,
         "NdsCH" = X4,
         "NdssC" = X5,
         "MLOGP" = X6,
         "LC50" = X7)
```

The columns that should be numeric in fishData are made numeric and the first 10 rows of each column are displayed.

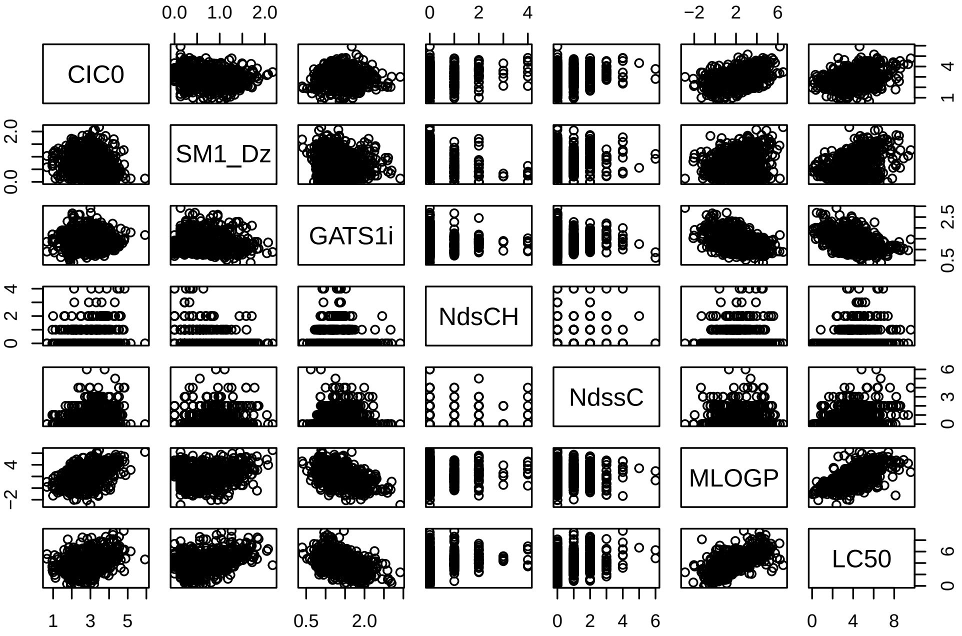
```
fishData <- transform(fishData, MLOGP = as.numeric(MLOGP),
                      CIC0 = as.numeric(CIC0),
                      GATS1i = as.numeric(GATS1i),
                      SM1_Dz = as.numeric(SM1_Dz),
                      LC50 = as.numeric(LC50))
fishData[1:10,]
```

```
##     CIC0 SM1_Dz GATS1i NdsCH NdssC MLOGP  LC50
## 1  3.260  0.829  1.676     0     1 1.453 3.770
## 2  2.189  0.580  0.863     0     0 1.348 3.115
## 3  2.125  0.638  0.831     0     0 1.348 3.531
## 4  3.027  0.331  1.472     1     0 1.807 3.510
## 5  2.094  0.827  0.860     0     0 1.886 5.390
## 6  3.222  0.331  2.177     0     0 0.706 1.819
## 7  3.179  0.000  1.063     0     0 2.942 3.947
## 8  3.000  0.000  0.938     1     0 2.851 3.513
## 9  2.620  0.499  0.990     0     0 2.942 4.402
## 10 2.834  0.134  0.950     0     0 1.591 3.021
```

EDA

We will explore the data in the EDA section of this report. First, a pairs plot is created to examine the relationships among the variables in fishData.

```
pairs(fishData)
```



Next, numeric summaries are made for the LC50 variable in combination with other variables. First, a 5 number summary for the LC50 variable alone is created. The mean of LC50 is also displayed.

)

```
fishData %>%
  summarize(minimum = min(LC50, na.rm =TRUE),
            Q1 = quantile(LC50, probs = .25, na.rm = TRUE),
            mean = mean(LC50, na.rm =TRUE),
            median = median(LC50, na.rm =TRUE),
            Q3 = quantile(LC50, probs = .75, na.rm = TRUE),
            maximum = max(LC50, na.rm =TRUE))
```

```
##   minimum      Q1     mean median     Q3 maximum
## 1   0.053 3.15175 4.064431 3.9875 4.9075   9.612
```

Then, another 5 number summary is created for LC50 across each value of the NdssC variable.

```
fishData %>%
  group_by(NdssC) %>%
  summarize(minimum = min(LC50, na.rm =TRUE),
            Q1 = quantile(LC50, probs = .25, na.rm = TRUE),
            mean = mean(LC50, na.rm =TRUE),
            median = median(LC50, na.rm =TRUE),
            Q3 = quantile(LC50, probs = .75, na.rm = TRUE),
            maximum = max(LC50, na.rm =TRUE))
```

```
## # A tibble: 7 x 7
##   NdssC minimum    Q1  mean median    Q3 maximum
##   <dbl>   <dbl> <dbl> <dbl>  <dbl> <dbl>   <dbl>
## 1     0   0.053  3.19  3.96   3.92  4.80    8.13
## 2     1   0.855  2.95  3.96   3.84  4.79    9.35
## 3     2   0.89   3.77  4.80   4.64  5.54    8.60
## 4     3   1.65   3.09  4.13   4.02  4.56    8.20
## 5     4   3.15   4.90  5.94   5.86  6.71    9.61
## 6     5   6.68   6.68  6.68   6.68  6.68    6.68
## 7     6   4.82   5.18  5.54   5.54  5.90    6.25
```

```
#do this for the other categorical variable as well
```

In the same way a 5 number summary is created for LC50 but across each value of the NdsCH varaible rather than NdssC.

```
fishData %>%
  group_by(NdsCH) %>%
  summarize(minimum = min(LC50, na.rm =TRUE),
            Q1 = quantile(LC50, probs = .25, na.rm = TRUE),
            mean = mean(LC50, na.rm =TRUE),
            median = median(LC50, na.rm =TRUE),
            Q3 = quantile(LC50, probs = .75, na.rm = TRUE),
            maximum = max(LC50, na.rm =TRUE))
```

```
## # A tibble: 5 x 7
##   NdsCH minimum    Q1  mean median    Q3 maximum
##   <dbl>   <dbl> <dbl> <dbl>  <dbl> <dbl>   <dbl>
## 1     0   0.053  3.05  3.96   3.92  4.82    9.35
## 2     1   0.841  3.58  4.50   4.23  5.13    9.61
## 3     2   2.38   3.97  4.83   4.64  5.54    7.38
## 4     3   4.32   4.47  4.69   4.59  4.86    5.19
## 5     4   3.41   4.09  5.39   6.35  6.43    6.92
```

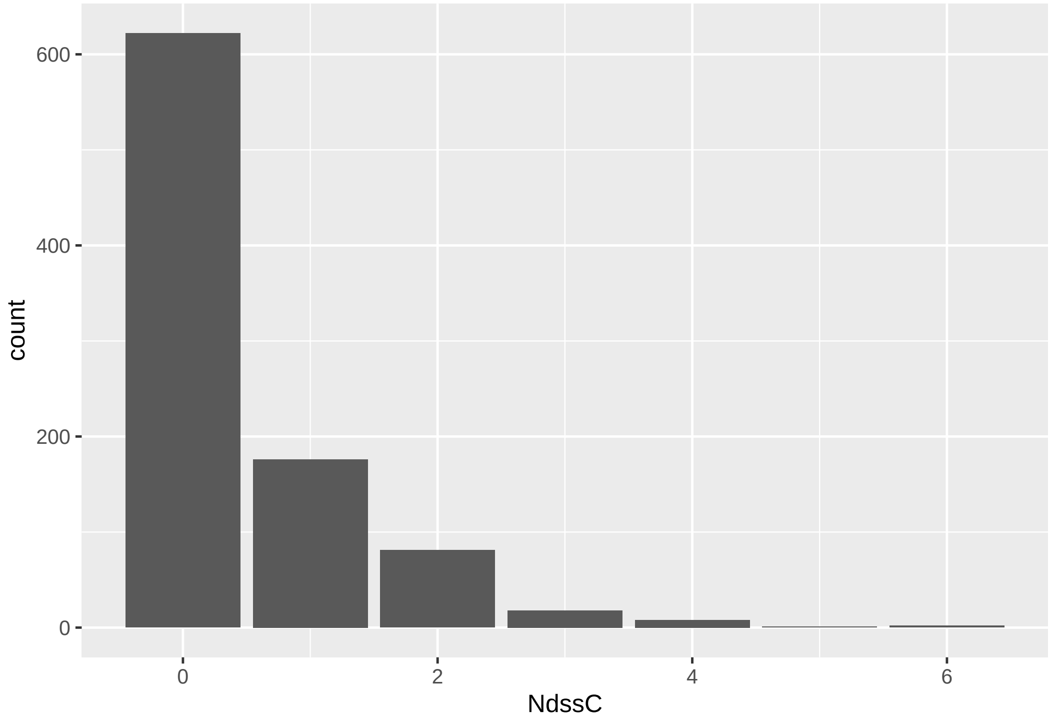
Finally, the variance of LC50 is displayed and the correlation between LC50 and each of the continuous variables is outputted.

```
fishData %>%
  summarize(variance = var(LC50, na.rm = TRUE),
            corMLOGP = cor(LC50, MLOGP, use =  "complete.obs"),
            corCIC0 = cor(LC50, CIC0, use =  "complete.obs"),
            corGATS1i = cor(LC50, GATS1i, use =  "complete.obs"),
            corSM1_Dz = cor(LC50, SM1_Dz, use =  "complete.obs"))
```

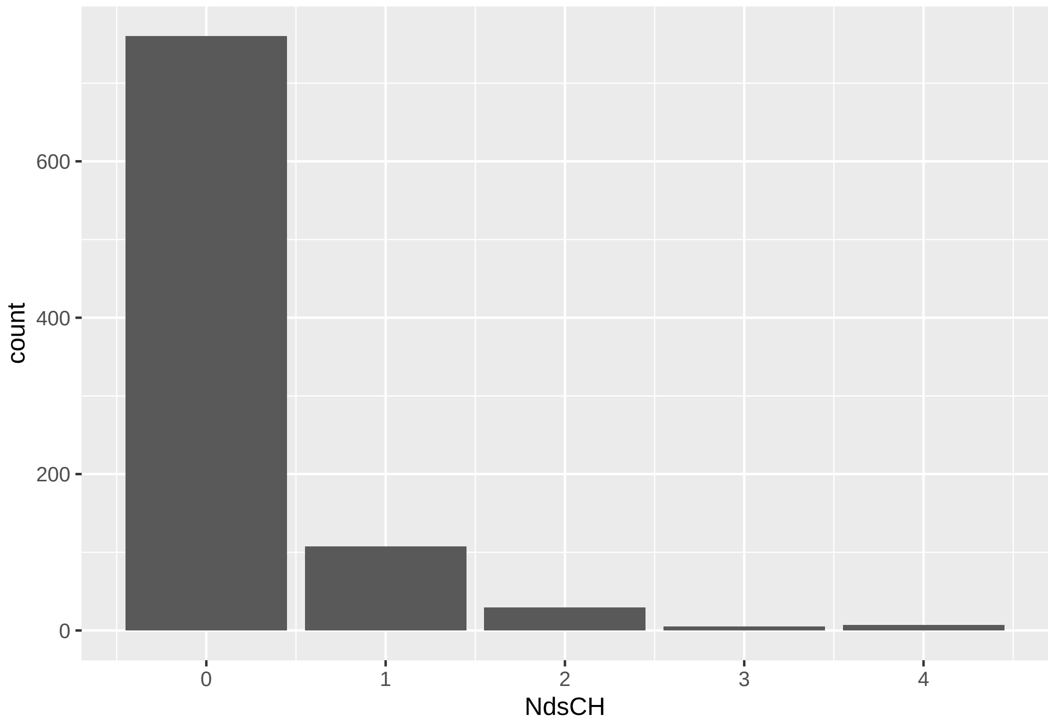
```
##   variance corMLOGP   corCIC0  corGATS1i corSM1_Dz
## 1 2.119058 0.651664 0.2918543 -0.3979647 0.4108932
```

Moving forward, different plots are created for the variables of fishData. First, the counts are examined for the two categorical variables: NdssC and NdsCH.

```
ggplot(fishData, aes(x = NdssC)) +
  geom_bar()
```

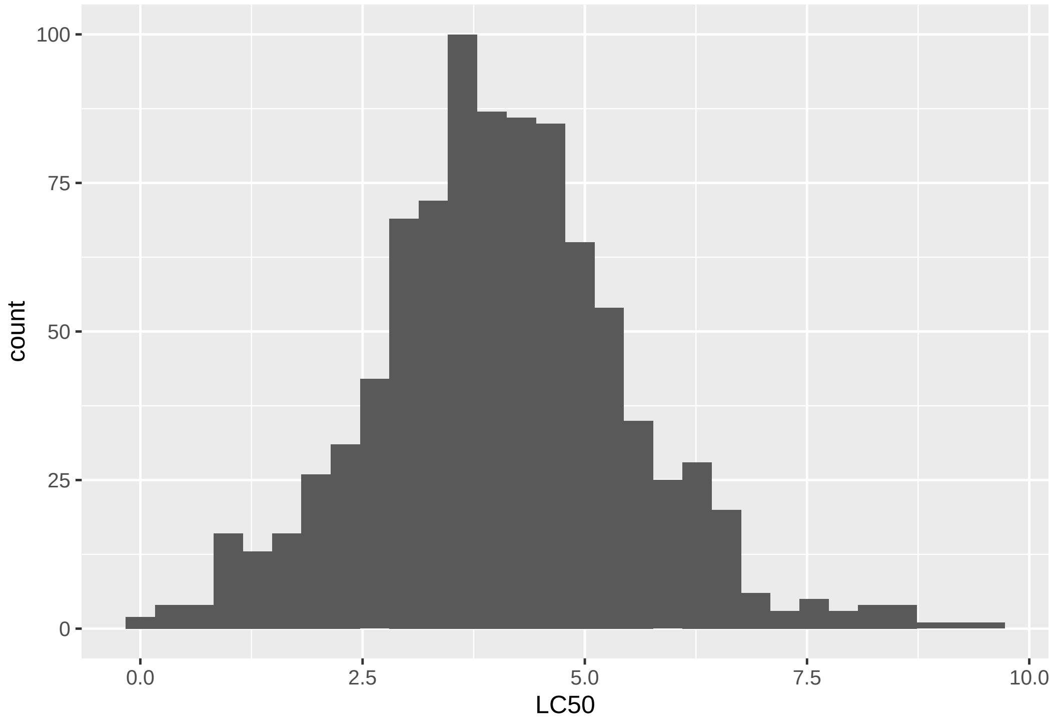


```
ggplot(fishData, aes(x = NdsCH)) +
  geom_bar()
```



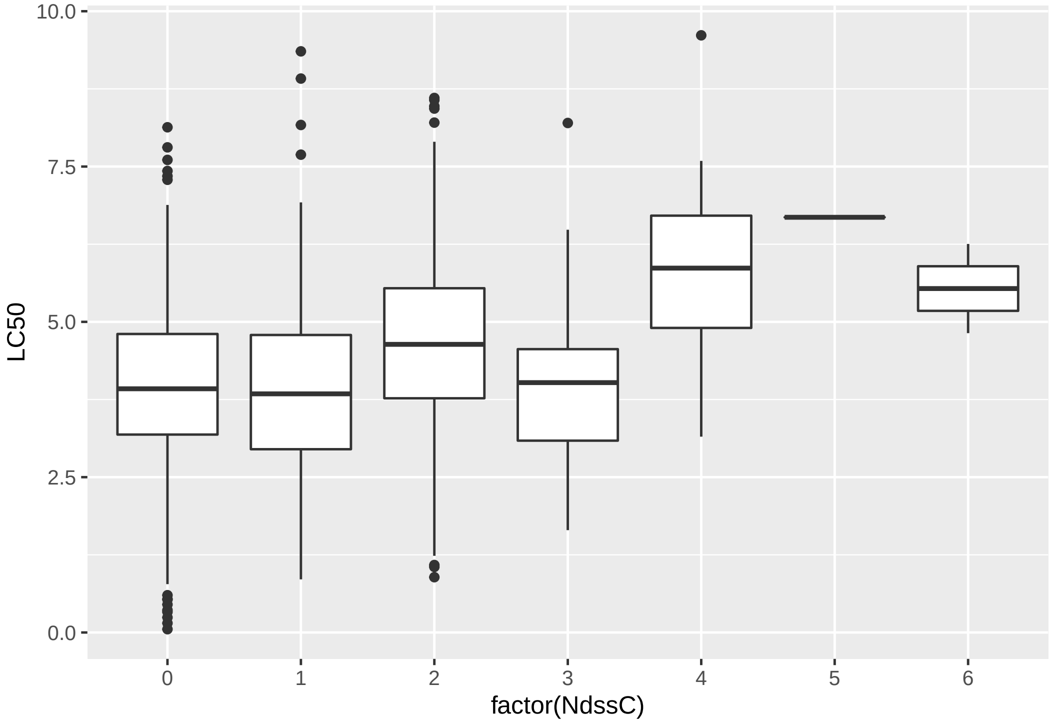
A histogram of the LC50 variable is then created to examine its distribution.

```r
ggplot(fishData, aes(x = LC50)) +
  geom_histogram()
```



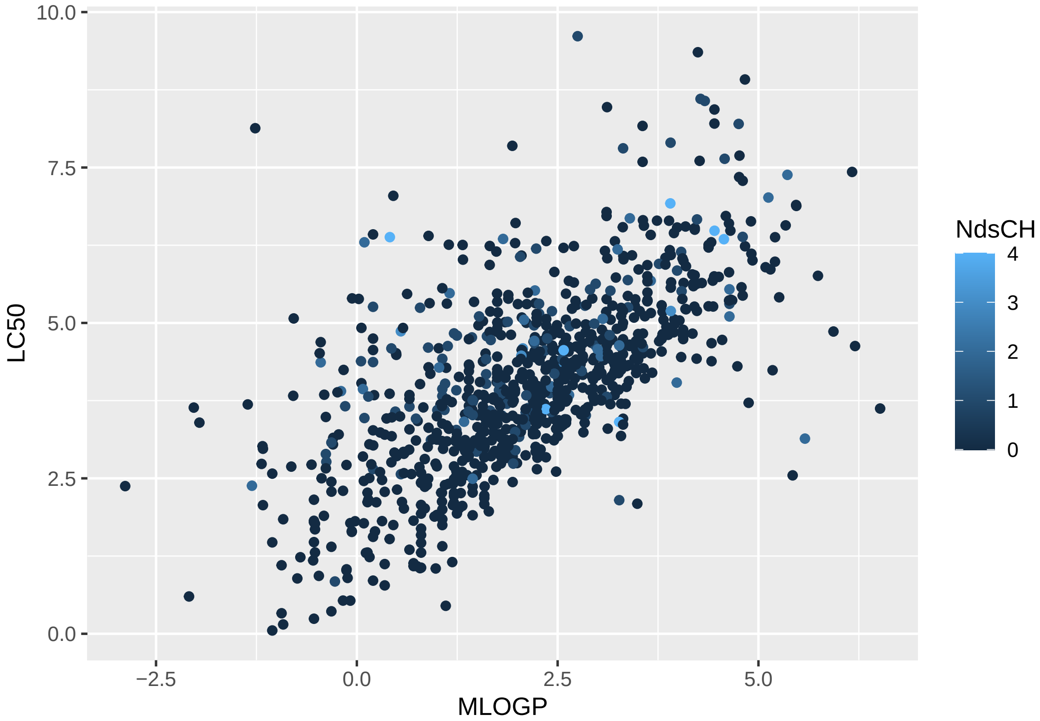
Similarly, side by side box plots are created for the values of LC50 that correspond with each value of NdssC. The factor() function is used as NdssC is a categorical variable.

```
ggplot(fishData, aes(x = factor(NdssC), y = LC50)) +
  geom_boxplot()
```



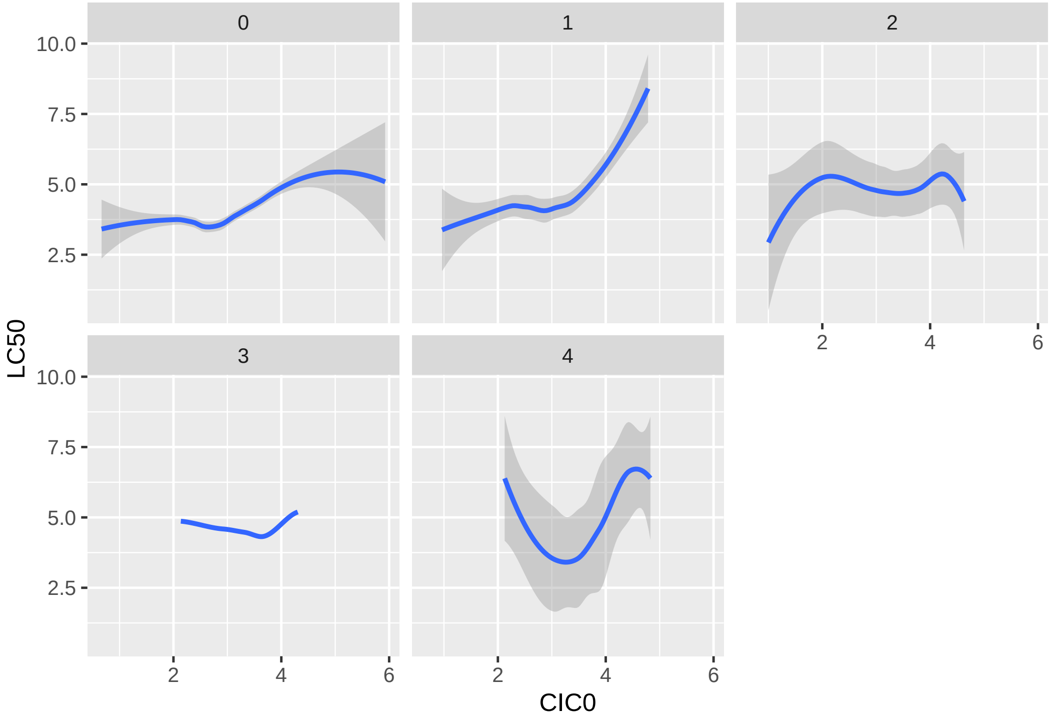
A scatter plot is created to examine the distribution of LC50 in correspondence with the MLOGP variable. The color of the points is determined using their NdsCH value.

```
ggplot(fishData, aes(x = MLOGP, y = LC50, color = NdsCH)) +
  geom_point()
```



Finally, smoothed plots are created to examine the relationship between CIC0 and LC50. The facet_wrap() function is used to display the relationships across the 5 values of NdsCH.

```
ggplot(fishData, aes(x = CIC0, y = LC50)) +
  geom_smooth() +
  facet_wrap(~NdsCH)
```



In the last section of EDA a function is created to transform the values of a variable into "low" or "high" depending on whether the value is less than or equal to, or greater than the median of the variable.

```
newFishData <- fishData
funLowHigh <- function(column) {
  return(if_else(column <= median(column), "low", "high"))
}
```

This function is then applied to to every column in the newFishData data set and the resulting data set is turned into a data frame.

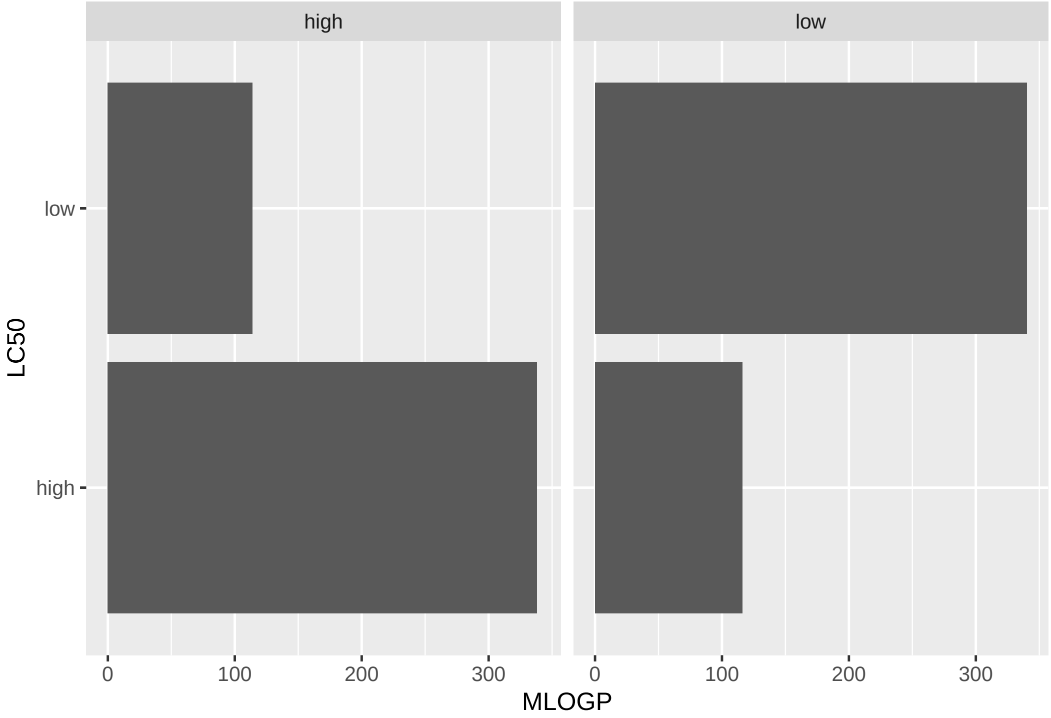
```
newFishData <- apply(X = newFishData,
      MARGIN = 2,
      FUN = funLowHigh)
newFishData <- data.frame(newFishData)
```

A two way contingency table comparing the binary values of the LC50 and MLOGP variables is created. Similarly, side by side box plots comparing the two variables are also created.

```
table(newFishData[,7], newFishData[,6])
```

```
##
##         high low
##   high  338 116
##   low   114 340
```

```
ggplot(newFishData, aes(y = newFishData[,7])) +
  geom_bar() +
  labs(y = "LC50", x = "MLOGP") +
  facet_wrap(~newFishData[,6])
```

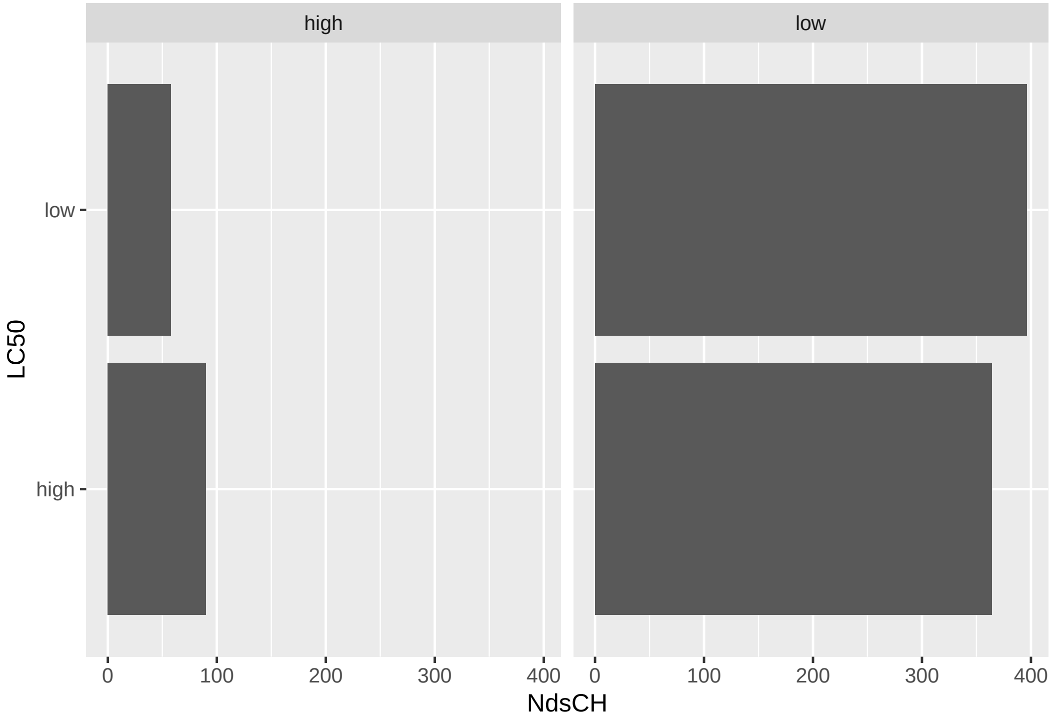


In the same way, a two way contingency table comparing the binary values in the LC50 and NdsCH variables is created. Side by side box plots comparing the two variables are also created.

```
table(newFishData[,7], newFishData[,4])
```

```
##
##        high low
##   high   90 364
##   low    58 396
```

```
ggplot(newFishData, aes(y = newFishData[,7])) +
  geom_bar() +
  labs(y = "LC50", x = "NdsCH") +
  facet_wrap(~newFishData[,4])
```



Multiple linear regression

The final section of this report covers linear regression. A model is first created using CIC0 and SM1_Dz.

```
fitSM1_Dz <- lm(LC50 ~ CIC0 + SM1_Dz, data = fishData)
summary(fitSM1_Dz)
```

```
##
## Call:
## lm(formula = LC50 ~ CIC0 + SM1_Dz, data = fishData)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -4.0856 -0.7129  0.0297  0.7096  4.8215
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.68512    0.18440   3.715 0.000215 ***
## CIC0         0.79197    0.05395  14.679  < 2e-16 ***
## SM1_Dz       1.72495    0.09521  18.117  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.194 on 905 degrees of freedom
## Multiple R-squared:  0.3287, Adjusted R-squared:  0.3272
## F-statistic: 221.5 on 2 and 905 DF,  p-value: < 2.2e-16
```

A model is then created using only MLOGP.

```
fitMLOGP <- lm(LC50 ~ MLOGP, data = fishData)
summary(fitMLOGP)
```

```
##
## Call:
## lm(formula = LC50 ~ MLOGP, data = fishData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7118 -0.6684 -0.1488  0.5559  6.3010
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.66829    0.06526   40.89   <2e-16 ***
## MLOGP        0.66190    0.02560   25.86   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.105 on 906 degrees of freedom
## Multiple R-squared:  0.4247, Adjusted R-squared:  0.424
## F-statistic: 668.7 on 1 and 906 DF,  p-value: < 2.2e-16
```

A model is created using GATS1i and the polynomial term of GATS1i.

```
fitGATS1i <- lm(LC50 ~ GATS1i + I(GATS1i^2), data = fishData)
summary(fitGATS1i)
```

```
##
## Call:
## lm(formula = LC50 ~ GATS1i + I(GATS1i^2), data = fishData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4825 -0.9112 -0.1091  0.7605  5.7505
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.3404     0.4292  12.442   <2e-16 ***
## GATS1i       -0.5182     0.6214  -0.834    0.405
## I(GATS1i^2)  -0.3312     0.2128  -1.556    0.120
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.335 on 905 degrees of freedom
## Multiple R-squared:  0.1606, Adjusted R-squared:  0.1588
## F-statistic: 86.59 on 2 and 905 DF,  p-value: < 2.2e-16
```

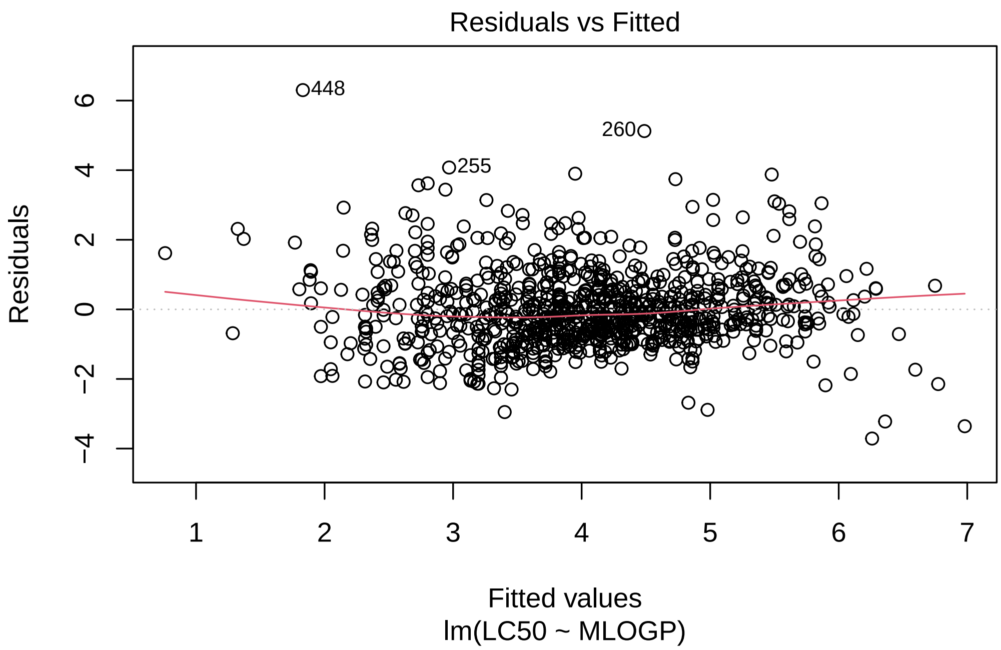
A model is created using GATS1i, NdsCH, and the interaction term of these two variables.

```
fitNdsCH <- lm(LC50 ~ GATS1i + NdsCH + GATS1i:NdsCH, data = fishData)
summary(fitNdsCH)
```

```
##
## Call:
## lm(formula = LC50 ~ GATS1i + NdsCH + GATS1i:NdsCH, data = fishData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3261 -0.9083 -0.1515  0.7873  5.4429
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.92030    0.15709  37.687   <2e-16 ***
## GATS1i       -1.50586    0.11578 -13.006   <2e-16 ***
## NdsCH        -0.01928    0.34072  -0.057    0.955
## GATS1i:NdsCH  0.32851    0.25877   1.269    0.205
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.314 on 904 degrees of freedom
## Multiple R-squared:  0.188,  Adjusted R-squared:  0.1853
## F-statistic: 69.75 on 3 and 904 DF,  p-value: < 2.2e-16
```

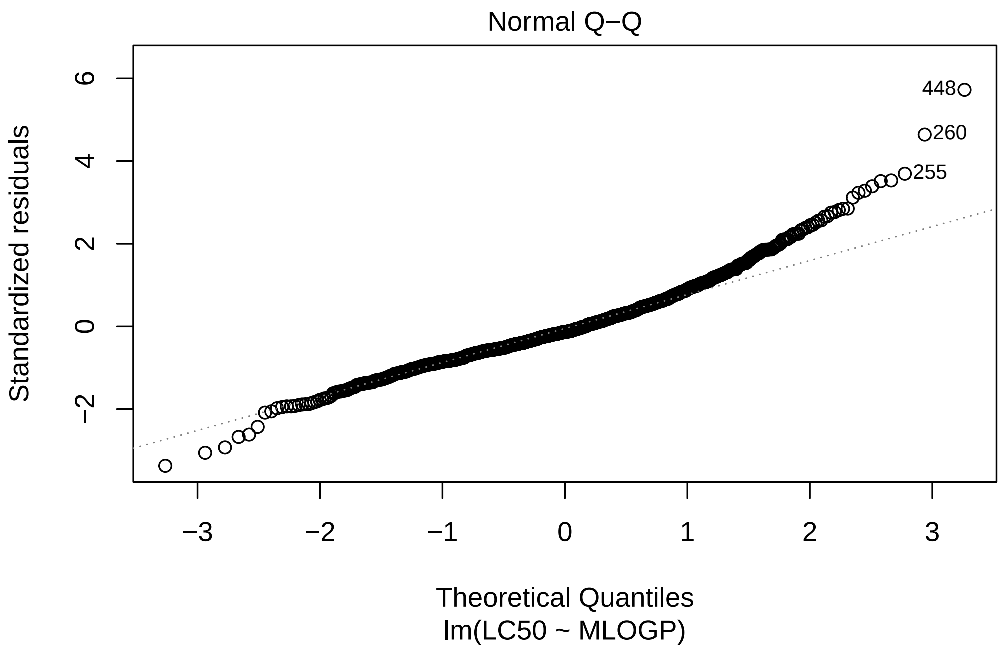
Subsequently, diagnostic plots are created to examine the model fit and normality assumptions. The fitted value vs residuals plot of LC50 and MLOGP displays a relatively random scatter and therefore the homoscedasticity assumption holds. This being said, the points in the bottom right of the plot are worrisome so we may need to transform our model.

```
plot(fitMLOGP, which = 1)
```



The QQ plot displays a relatively straight line indicating the normality assumption is met.

```
plot(fitMLOGP, which = 2)
```



Finally, the fitMLOGP and fitNdsCH models created previously are used to predict the value of LC50. The values used are the median values of the variables in each model.

```
predict(fitMLOGP, newdata = data.frame(MLOGP = median(fishData$MLOGP)))
```

```
##        1
## 4.076156
```

```
predict(fitNdsCH, newdata = data.frame(GATS1i = median(fishData$GATS1i), NdsCH = median(fishData$NdsCH)
```

```
##        1
## 4.052282
```