Prediction with decision tree method in algae dataset.

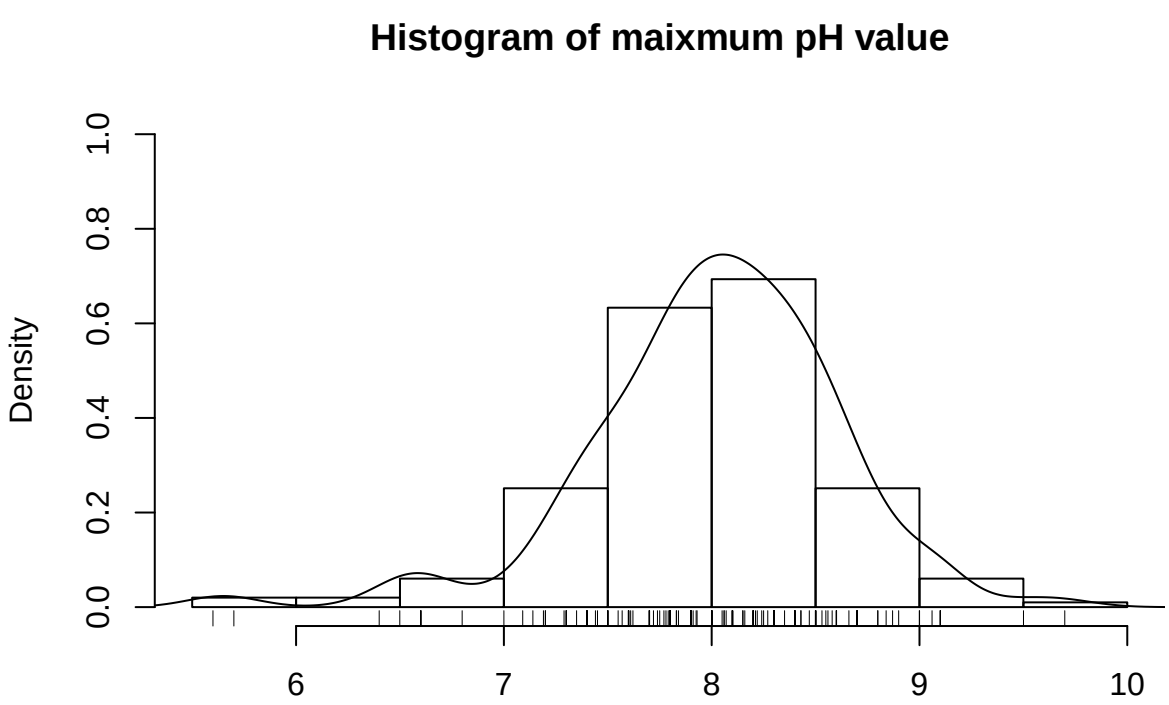
Yan Yan

February 22, 2015

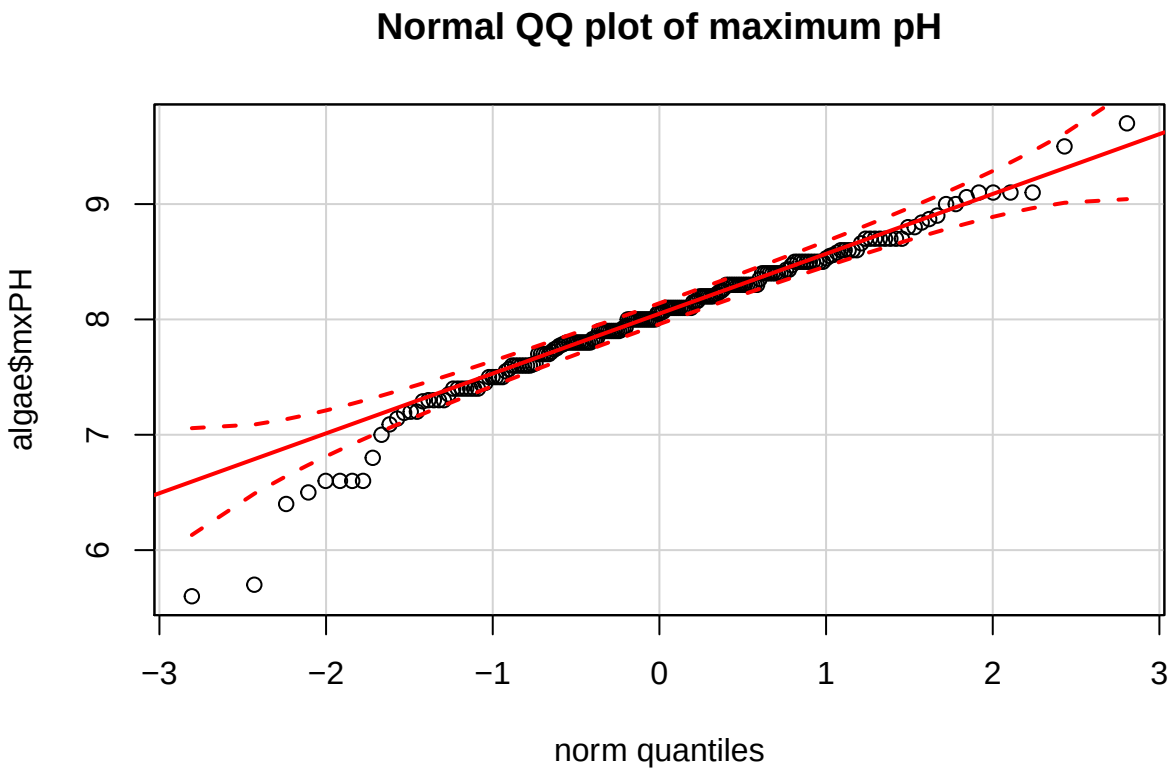
```
library(DMwR)
```

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

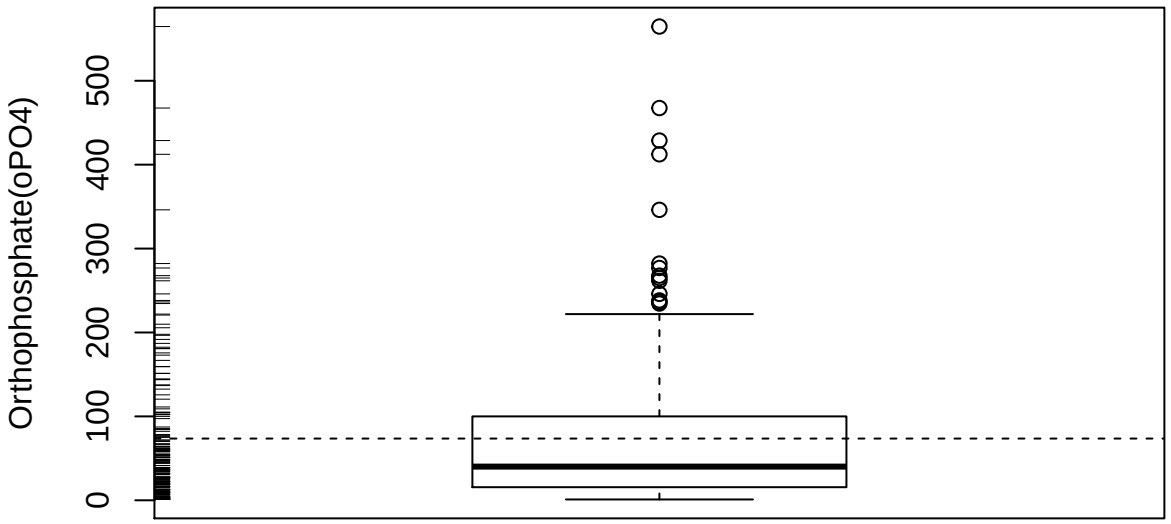
```
library(car)

#par(mfrow=c(1,2))
hist(algae$mxPH, prob=T, xlab='', main='Histogram of maixmum pH value', ylim=0:1)

# A smoothed version of hist graph
lines(density(algae$mxPH, na.rm=T))

#rug() used for plotting, jitter() randomized the original value to avoid overlap
rug(jitter(algae$mxPH))
```




```
#Q-Q graph, plot the scatterplot of value and Normal Distribution quantiles,
#and then the band chart of 95% confidence interval
qqPlot(algae$mxPH, main='Normal QQ plot of maximum pH')
```

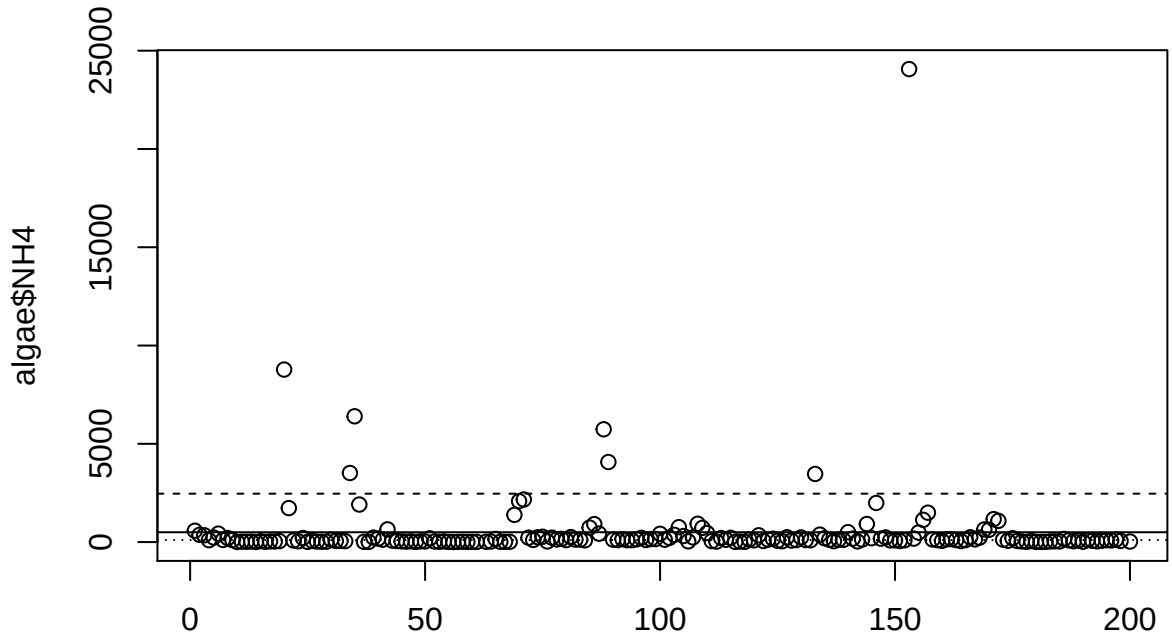



```
#par(mfrow=c(1,2))

#Boxplot of oPO4, the line in the middle of the box is the median, the upper
#boudary is the 3rd quantiles and the lower boundary is the 1st quantile.
boxplot(algae$oPO4, ylab = "Orthophosphate(oPO4)")

rug(jitter(algae$oPO4), side=2)
#Plot the average value line in the Boxplot
abline(h = mean(algae$oPO4, na.rm = T), lty = 2)
```



We can see the the distribution of oPO4 is concentrated in the lower values, so it is positive-skewed. For the sperated values, we can handle with the following method.

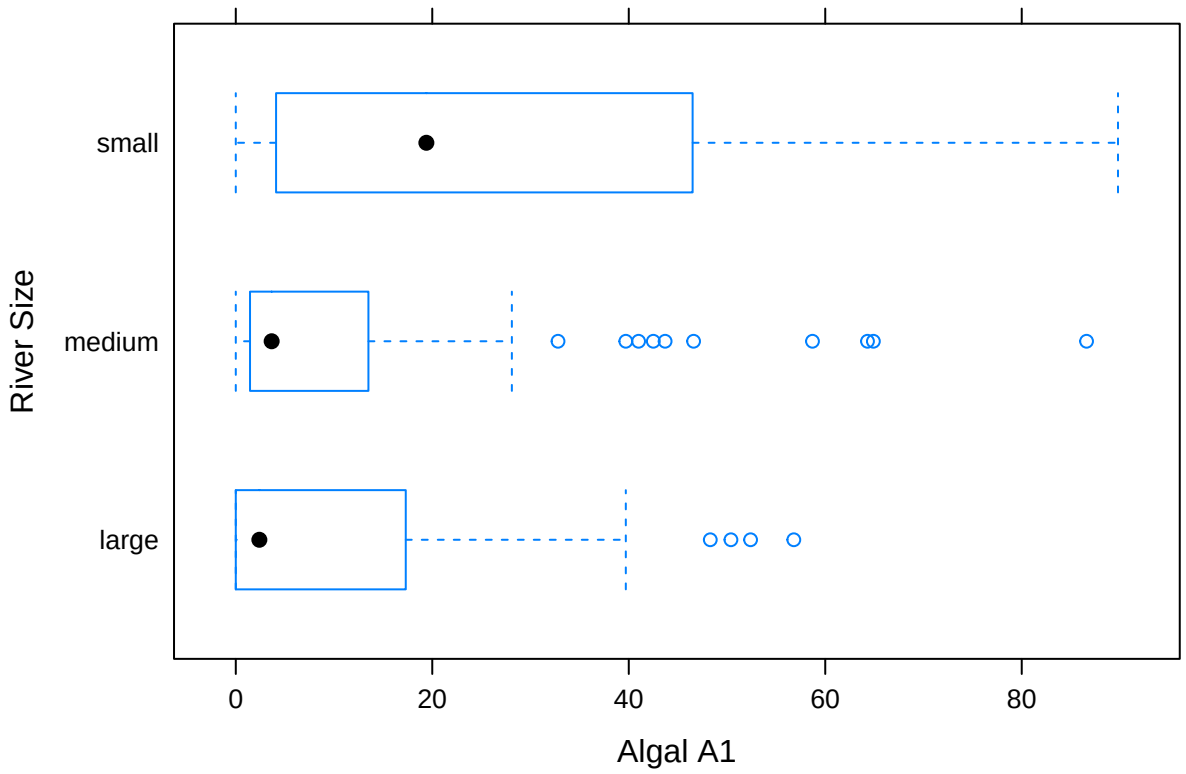
```
plot(algae$NH4, xlab = " ")
abline(h = mean(algae$NH4, na.rm = T), lty = 1) #avarage value
abline(h = mean(algae$NH4, na.rm = T) + sd(algae$NH4, na.rm = T), lty = 2)
abline(h = median(algae$NH4, na.rm = T), lty = 3) #median
```




```
#A interact method can be realized by using identify() function
#identify(algae$NH4)
#clicked.lines <- identify(algae$NH4)
#algae[clicked.lines]
```

To find how the distribution varies due to other variables.

```
library(lattice)
bwplot(size ~ a1, data=algae, ylab = 'River Size', xlab = 'Algal A1')
```

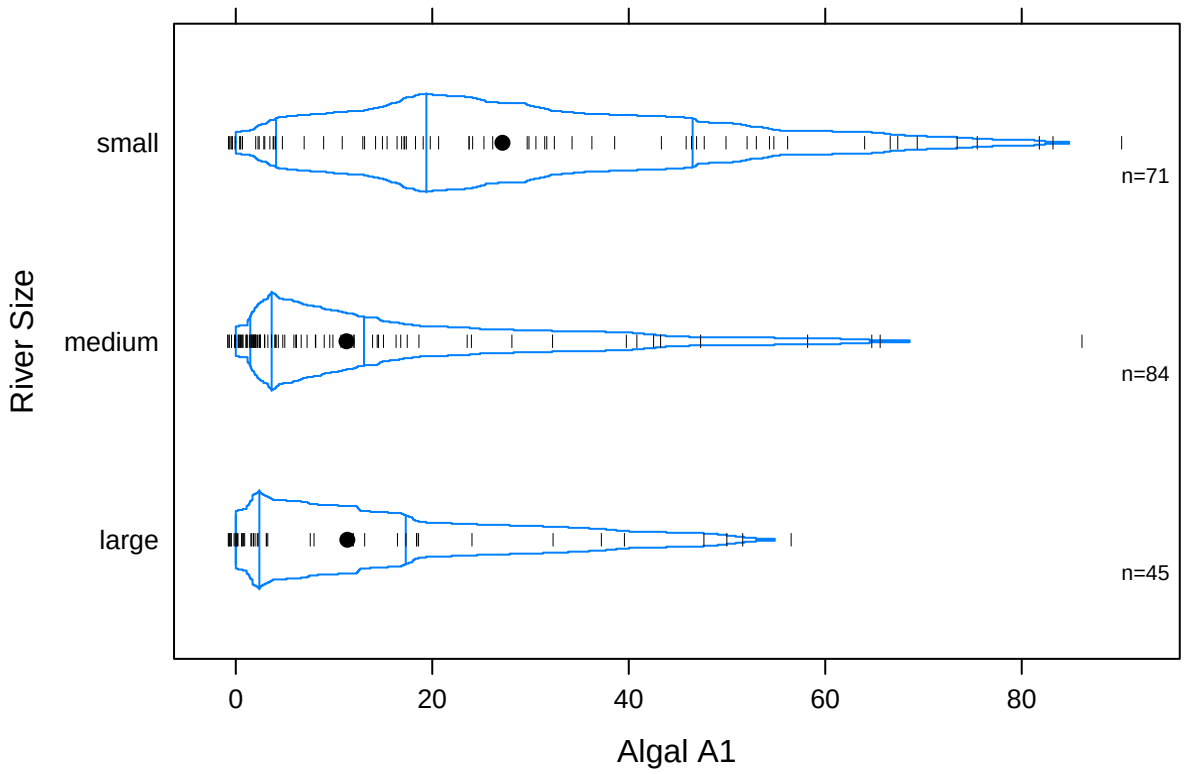


Also we can use the quantile box plot

```
library(Hmisc)
```

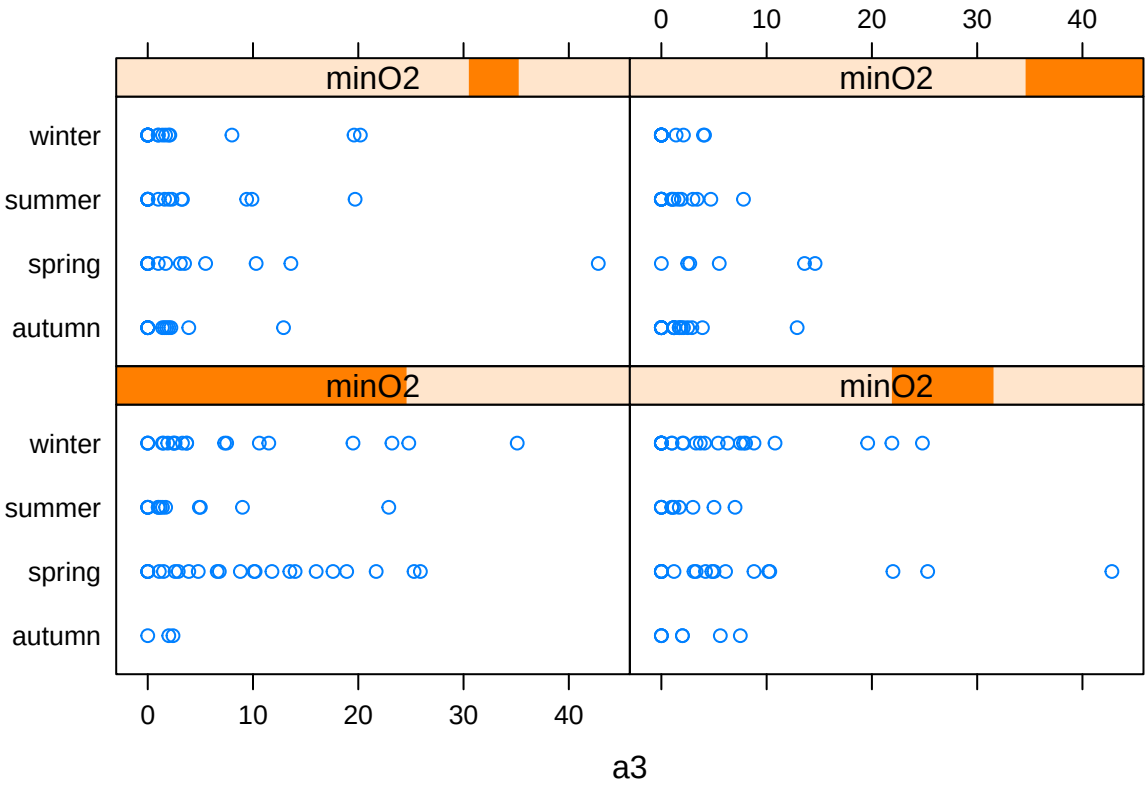
```
## Loading required package: survival
## Loading required package: splines
## Loading required package: Formula
##
## Attaching package: 'Hmisc'
##
## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
```

```
bwplot(size ~ a1, data = algae, panel = panel.bpplot,
       probs = seq(.01, .49, by = .01), datadensity = TRUE,
       ylab = 'River Size', xlab = 'Algal A1')
```



We can also discretize the data into several intervals, which means transfer the continus numerical data into factor data, for example:

```
minO2 <- equal.count(na.omit(algae$mnO2), number = 4, overlap = 1/5)
stripplot(season ~ a3|minO2, data=algae[!is.na(algae$mnO2),])
```



Then we go to the missing value process.

```
#caculate the numbers of lines with missing value
algae[!complete.cases(algae),]
```

```
##     season   size  speed mxPH mnO2    Cl   NO3 NH4    oPO4     PO4  Chla
## 28  autumn  small   high 6.80 11.1 9.000 0.630  20   4.000      NA  2.70
## 38  spring  small   high 8.00   NA 1.450 0.810  10   2.500   3.000  0.30
## 48  winter  small    low   NA 12.6 9.000 0.230  10   5.000   6.000  1.10
## 55  winter  small   high 6.60 10.8    NA 3.245  10   1.000   6.500    NA
## 56  spring  small medium 5.60 11.8    NA 2.220   5   1.000   1.000    NA
## 57  autumn  small medium 5.70 10.8    NA 2.550  10   1.000   4.000    NA
## 58  spring  small   high 6.60  9.5    NA 1.320  20   1.000   6.000    NA
## 59  summer  small   high 6.60 10.8    NA 2.640  10   2.000  11.000    NA
## 60  autumn  small medium 6.60 11.3    NA 4.170  10   1.000   6.000    NA
## 61  spring  small medium 6.50 10.4    NA 5.970  10   2.000  14.000    NA
## 62  summer  small medium 6.40   NA    NA    NA  NA      NA  14.000    NA
## 63  autumn  small   high 7.83 11.7 4.083 1.328  18   3.333   6.667    NA
## 116 winter medium   high 9.70 10.8 0.222 0.406  10  22.444  10.111    NA
## 161 spring  large    low 9.00  5.8    NA 0.900 142 102.000 186.000 68.05
## 184 winter  large   high 8.00 10.9 9.055 0.825  40  21.083  56.091    NA
## 199 winter  large medium 8.00  7.6    NA    NA  NA      NA      NA    NA
##       a1   a2  a3   a4  a5  a6  a7
## 28  30.3  1.9 0.0  0.0 2.1 1.4 2.1
## 38  75.8  0.0 0.0  0.0 0.0 0.0 0.0
## 48  35.5  0.0 0.0  0.0 0.0 0.0 0.0
## 55  24.3  0.0 0.0  0.0 0.0 0.0 0.0
## 56  82.7  0.0 0.0  0.0 0.0 0.0 0.0
## 57  16.8  4.6 3.9 11.5 0.0 0.0 0.0
```

```
## 58  46.8  0.0 0.0 28.8 0.0 0.0 0.0
## 59  46.9  0.0 0.0 13.4 0.0 0.0 0.0
## 60  47.1  0.0 0.0  0.0 0.0 1.2 0.0
## 61  66.9  0.0 0.0  0.0 0.0 0.0 0.0
## 62  19.4  0.0 0.0  2.0 0.0 3.9 1.7
## 63  14.4  0.0 0.0  0.0 0.0 0.0 0.0
## 116 41.0  1.5 0.0  0.0 0.0 0.0 0.0
## 161  1.7 20.6 1.5  2.2 0.0 0.0 0.0
## 184 16.8 19.6 4.0  0.0 0.0 0.0 0.0
## 199  0.0 12.5 3.7  1.0 0.0 0.0 4.9
```

```
nrow(algae[!complete.cases(algae),])
```

```
## [1] 16
```

```
#delete the lines with missing values
#algae <- na.omit(algae)

data(algae)
algae <- algae[-manyNAs(algae),]
algae <- centralImputation(algae)

#cor(algae[, 4:18], use = "complete.obs")
symnum(cor(algae[, 4:18], use = "complete.obs"))
```

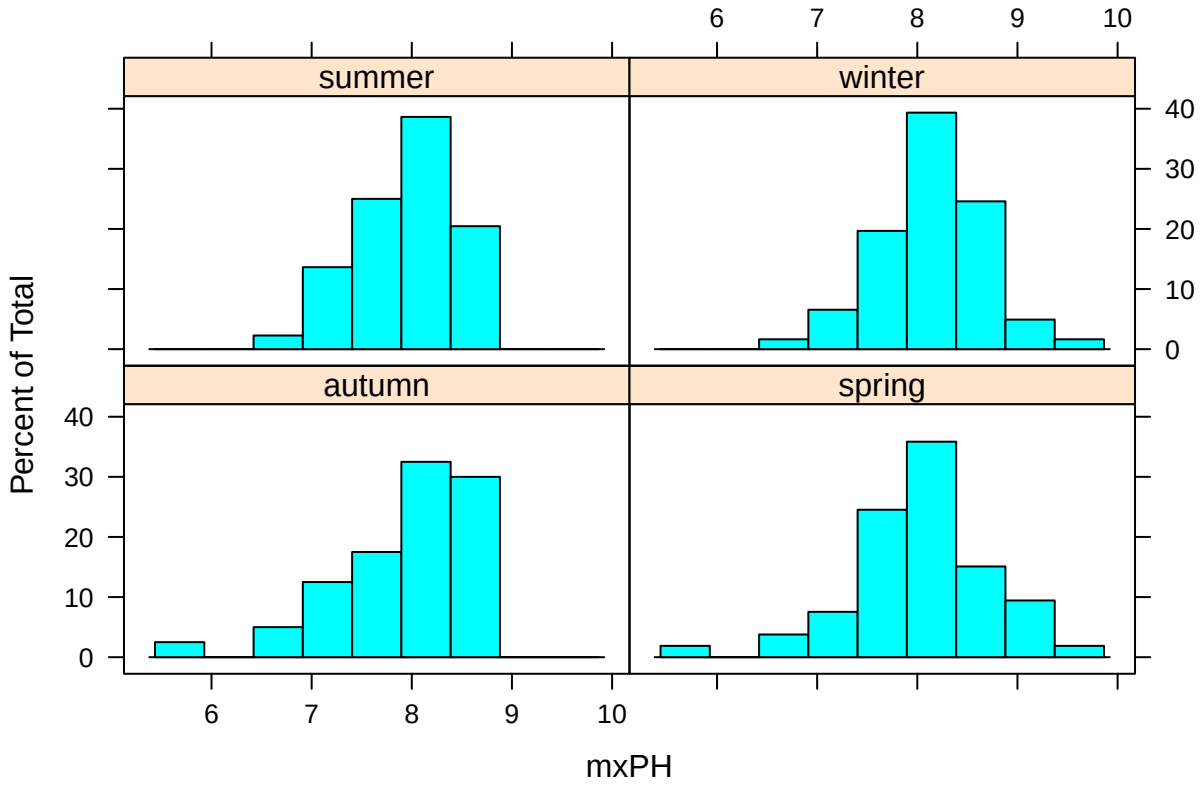
```
##      mP mO Cl NO NH o P Ch a1 a2 a3 a4 a5 a6 a7
## mxPH 1
## mnO2    1
## Cl         1
## NO3           1
## NH4           ,  1
## oPO4    .  .        1
## PO4     .  .        * 1
## Chla .                  1
## a1          .        . .    1
## a2   .                    .     1
## a3                               1
## a4      .             .             1
## a5                                   1
## a6            .  .                   .  1
## a7                                        1
## attr(,"legend")
## [1] 0 ' ' 0.3 '.' 0.6 ',' 0.8 '+' 0.9 '*' 0.95 'B' 1
```

```
data(algae)
algae <- algae[-manyNAs(algae),]
lm(PO4 ~ oPO4, data = algae)
```

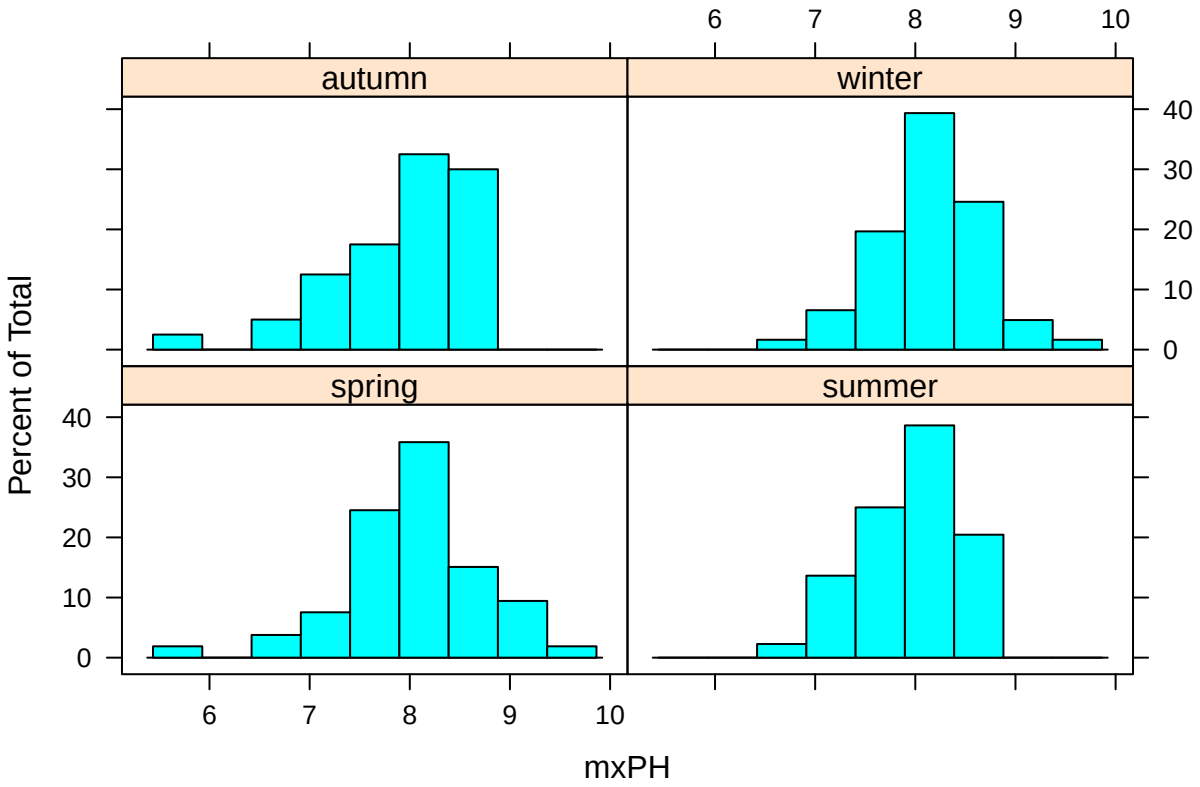
```
##
## Call:
## lm(formula = PO4 ~ oPO4, data = algae)
##
```

```
## Coefficients:
## (Intercept)         oPO4
##       42.90         1.29
```

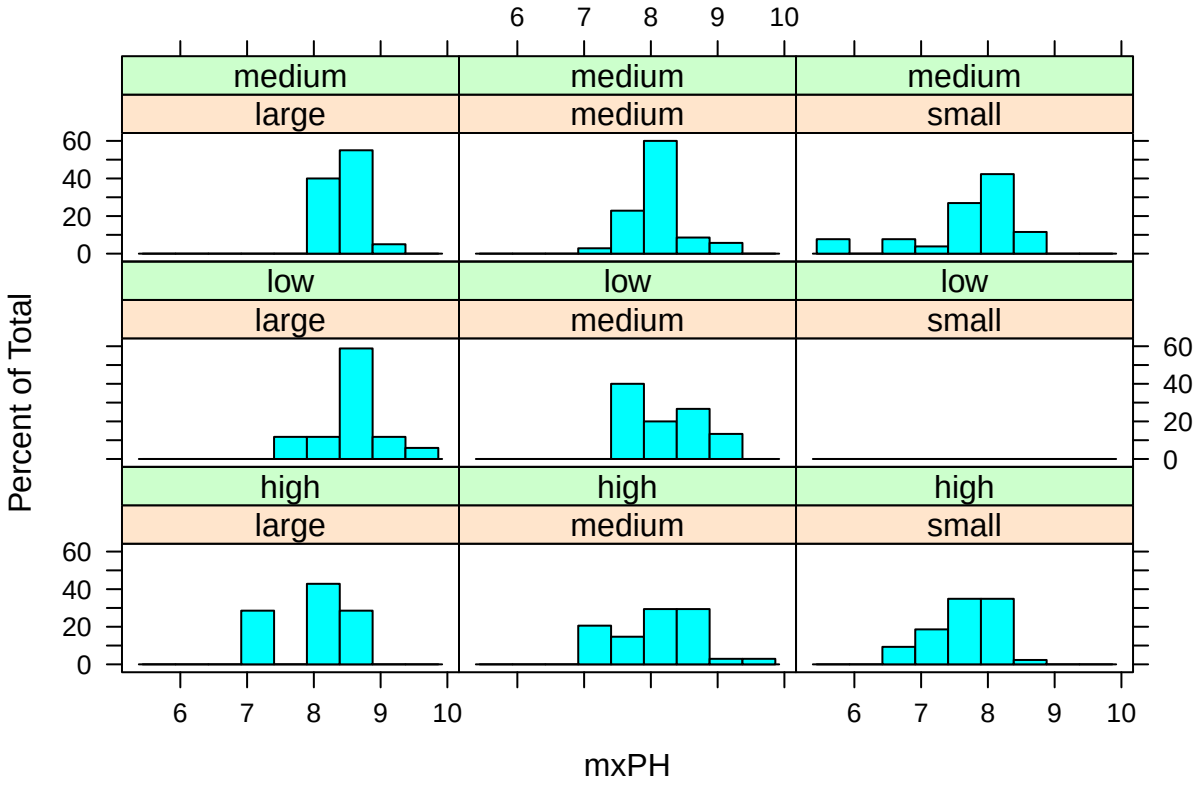
```
algae[28, "PO4"] <- 42.897 + 1.293 * algae[28, "oPO4"]

data(algae)
algae <- algae[-manyNAs(algae),]
fillPO4 <- function(oP){
        if(is.na(oP))
                return(NA)
        else return(42.897 + 1.293 * oP)
}

algae[is.na(algae$PO4), "PO4"] <- sapply(algae[is.na(algae$PO4), "oPO4"], fillPO4)

histogram(~mxPH | season, data = algae)
```



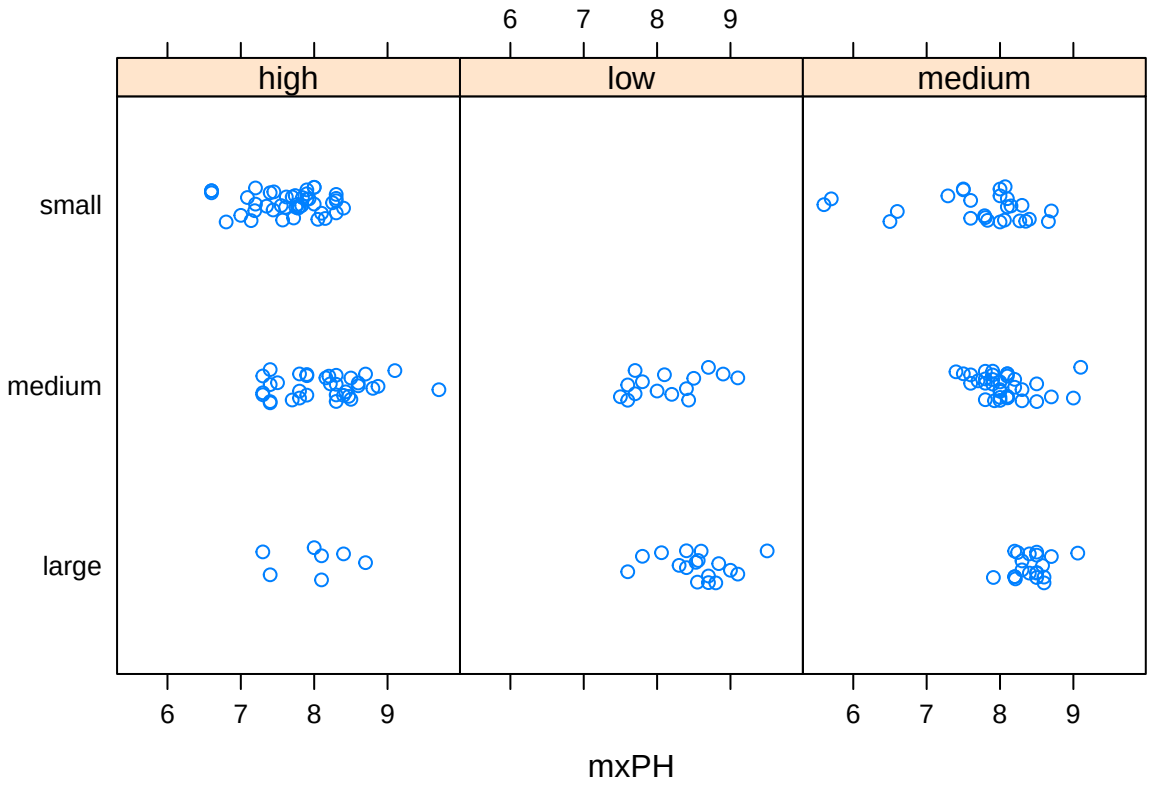

```
algae$season <- factor(algae$season, levels = c("spring", "summer", "autumn", "winter"))
histogram(~mxPH | season, data = algae)
```




```
histogram(~mxPH | size * speed, data =algae)
```



```
stripplot(size ~ mxPH | speed, data = algae, jitter = T)
```



If we assume the two kinds of water were similar, then in one sample there were some missing value, the missing value might be similar to the correspond one in the other kind of water.

```
data(algae)
algae <- algae[-manyNAs(algae),]

#algae <- knnImputation(algae, k = 10)
#or
algae <- knnImputation(algae, k = 10, meth = "median")
```

First we build a multivariate analysis for prediction

```
data(algae)
algae <- algae[-manyNAs(algae),]
clean.algae <- knnImputation(algae, k = 10)

lm.al <- lm(a1 ~., data = clean.algae[,1:12])
summary(lm.al)
```

```
##
## Call:
## lm(formula = a1 ~ ., data = clean.algae[, 1:12])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -37.68 -11.89  -2.57   7.41  62.19
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  42.94206   24.01088    1.79   0.0754 .
## seasonspring  3.72698    4.13774    0.90   0.3689
## seasonsummer  0.74760    4.02071    0.19   0.8527
## seasonwinter  3.69295    3.86539    0.96   0.3406
## sizemedium    3.26373    3.80205    0.86   0.3918
## sizesmall     9.68214    4.17997    2.32   0.0217 *
## speedlow      3.92208    4.70631    0.83   0.4057
## speedmedium   0.24676    3.24187    0.08   0.9394
## mxPH         -3.58912    2.70353   -1.33   0.1860
## mnO2          1.05264    0.70502    1.49   0.1372
## Cl           -0.04017    0.03366   -1.19   0.2343
## NO3          -1.51124    0.55134   -2.74   0.0067 **
## NH4           0.00163    0.00100    1.63   0.1052
## oPO4         -0.00543    0.03988   -0.14   0.8918
## PO4          -0.05224    0.03075   -1.70   0.0911 .
## Chla         -0.08802    0.08000   -1.10   0.2727
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.6 on 182 degrees of freedom
## Multiple R-squared:  0.373,  Adjusted R-squared:  0.321
## F-statistic: 7.22 on 15 and 182 DF,  p-value: 2.44e-12
```