Melting Point Data

Harper Schwab

9/22/2020

Introduction

Karthikeyan's article is proposing a new model, using a neural network to analyze compounds and predict the melting points of said compounds. What I had to do was analyze3e the data in order to find correlation and causality between 2D factors, and the compounds melting point. After finding basic correlations with pairwise plots, and regression plots, I was able to perform BIC analyses to find causation if there was any. I found that the 2D descriptors that I measured heavily affected the melting point with differences in BIC scores >100. The article uses p scores to find weather or not signs were statistical significant, whereas I can look at the BIC score. Overall it is interesting to see the different methods used between this deliverable and the article, and the fact that they can achieve similar results. ## Importing data and Library For this project I am using tidyverse, dyplr and reading melting point data from M. Karthikeyan.

```
library(tidyverse)
library(dplyr)
#
#Load mouse data
mp <- read.csv("dirtyMPdata.csv", header = TRUE)
```

Cleaning data

To start I looked at the data demographics.

```
names(mp)
```

```
##  [1] "ï..structure"  "mp"            "rings"         "heavy.atoms"
##  [5] "single.bonds"  "triple.bonds"  "reactive"      "molar.mass"
##  [9] "refractivity"  "formal.charge" "logP"          "tpsa"
## [13] "dipole.moment" "energy"        "density"       "volume"
## [17] "PCA1"          "PCA2"
```

```
dim(mp)
```

```
## [1] 4450   18
```

```
str(mp)
```

```
## 'data.frame':    4450 obs. of  18 variables:
##  $ ï..structure : chr  "O=C1Cc2ccccc21" "Clc1ccc(cc1)C1c2c(OC(N)=C1C#N)[nH][nH0]c2C(F)(F)F" "O=
C(OC)C(=Cc1ccccc1)Cc1ccccc1" "FC(F)(F)c1[nH0]cc2ccccc2c1" ...
##  $ mp           : num  14 20.5 27.5 30.5 31 31.5 32 32.5 33 34 ...
##  $ rings        : int  6 11 12 10 6 6 6 5 12 12 ...
##  $ heavy.atoms  : int  9 23 19 14 12 14 12 10 16 14 ...
##  $ single.bonds : int  9 20 22 10 16 19 14 12 21 16 ...
##  $ triple.bonds : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ reactive     : int  0 0 0 0 1 0 0 0 0 0 ...
##  $ molar.mass   : num  118 341 252 197 162 ...
##  $ refractivity : num  3.56 7.73 7.8 4.73 4.65 ...
##  $ formal.charge: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ logP         : num  1.43 3.64 3.49 3.57 1.94 ...
##  $ tpsa         : num  17.1 87.7 26.3 12.9 26.3 ...
##  $ dipole.moment: num  2.72 6.32 1.91 5.2 1.76 ...
##  $ energy       : num  -57.8 -189.1 -123.5 -119.7 -83.5 ...
##  $ density      : num  0.959 1.296 0.946 1.195 0.978 ...
##  $ volume       : num  123 263 267 165 166 ...
##  $ PCA1         : num  15.55 1.92 2.98 11.03 10.97 ...
##  $ PCA2         : num  1.622 0.502 2.723 1.826 -0.905 ...
```

```
sapply(mp, class)
```

```
##  ï..structure            mp         rings   heavy.atoms  single.bonds
##   "character"     "numeric"     "integer"     "integer"     "integer"
##  triple.bonds      reactive    molar.mass  refractivity formal.charge
##     "integer"     "integer"     "numeric"     "numeric"     "integer"
##          logP          tpsa dipole.moment        energy       density
##     "numeric"     "numeric"     "numeric"     "numeric"     "numeric"
##        volume          PCA1          PCA2
##     "numeric"     "numeric"     "numeric"
```

```
head(mp)
```

```
##                                      ï..structure   mp rings heavy.atoms
## 1                                  O=C1Cc2ccccc21 14.0     6           9
## 2 Clc1ccc(cc1)C1c2c(OC(N)=C1C#N)[nH][nH0]c2C(F)(F)F 20.5    11          23
## 3                     O=C(OC)C(=Cc1ccccc1)Cc1ccccc1 27.5    12          19
## 4                        FC(F)(F)c1[nH0]cc2ccccc2c1 30.5    10          14
## 5                              O=C(OC1Cc2ccccc21)C 31.0     6          12
## 6                            O=C(OC)C1=Cc2ccccc2C1C 31.5     6          14
##   single.bonds triple.bonds reactive molar.mass refractivity formal.charge
## 1            9            0        0    118.135     3.557232             0
## 2           20            1        0    340.692     7.729887             0
## 3           22            0        0    252.313     7.799344             0
## 4           10            0        0    197.159     4.729193             0
## 5           16            0        1    162.188     4.651153             0
## 6           19            0        0    188.226     5.533352             0
##       logP  tpsa dipole.moment     energy   density   volume    PCA1    PCA2
## 1 1.425370 17.07      2.722717  -57.76601 0.9589853 123.1875 15.5507  1.6219
## 2 3.637884 87.72      6.320718 -189.06262 1.2962539 262.8281  1.9216  0.5024
## 3 3.485670 26.30      1.911006 -123.45357 0.9463216 266.6250  2.9803  2.7226
## 4 3.565100 12.89      5.203625 -119.71445 1.1952425 164.9531 11.0316  1.8263
## 5 1.942370 26.30      1.759848  -83.47468 0.9782332 165.7969 10.9710 -0.9051
## 6 2.360100 26.30      2.046720  -95.11963 0.9741601 193.2188  8.9700  0.3246
```

```
tail(mp)
```

```
##                                             ï..structure  mp rings
## 4445                        N=1CCNC=1Cc1c(C)cc(cc1C)C(C)(C)C 131     6
## 4446            O=C(OC)C1C(O)CCC2CN3CCc4c5ccccc5[nH]c4C3CC21 234     9
## 4447              O=C1NC(=O)C(C)=CN1C1OC(CO)C(N=[N+]=[N-])C1 106     0
## 4448                              s1c2ccccc2cc1C(N(O)C(=O)N)C 157     9
## 4449 [S+2]([O-])([O-])(C)c1ccc(cc1)c1[nH0]c2[nH0](C=CC=C2)c1 242    11
## 4450                       Clc1ccc2Sc3ccccc3C=C(OCCN(C)C)c2c1  90    12
##      heavy.atoms single.bonds triple.bonds reactive molar.mass refractivity
## 4445          18           37            0        0    245.390     7.652000
## 4446          26           46            0        1    355.458     9.993037
## 4447          19           28            0        1    267.245     6.299167
## 4448          16           18            0        1    236.295     6.457717
## 4449          19           20            0        0    272.328     7.474073
## 4450          22           30            0        0    332.875     9.547561
##      formal.charge    logP   tpsa dipole.moment    energy   density   volume
## 4445             1 1.22571  26.00     13.028417 -112.7691 0.8720617 281.3906
## 4446             1 1.32547  66.76      8.226915 -177.9712 1.0188236 348.8906
## 4447             0 0.22820 103.59      4.728840 -146.1079 1.1644663 229.5000
## 4448             0 2.82770  66.56      1.821058 -112.0673 1.0833784 218.1094
## 4449             0 2.45110  51.96      8.410813 -127.4689 1.0959562 248.4844
## 4450             1 3.26210  13.67     23.597216 -146.3722 1.0327710 322.3125
##         PCA1    PCA2
## 4445  2.2338  1.6217
## 4446 -3.7905 -1.1574
## 4447  2.7954 -7.6893
## 4448  5.4399 -3.3301
## 4449  3.0028 -0.5994
## 4450 -0.2923  4.3546
```

```
glimpse(mp)
```

```
## Rows: 4,450
## Columns: 18
## $ ï..structure  <chr> "O=C1Cc2ccccc21", "Clc1ccc(cc1)C1c2c(OC(N)=C1C#N)[nH]...
## $ mp            <dbl> 14.0, 20.5, 27.5, 30.5, 31.0, 31.5, 32.0, 32.5, 33.0,...
## $ rings         <int> 6, 11, 12, 10, 6, 6, 6, 5, 12, 12, 6, 14, 11, 6, 6, 6...
## $ heavy.atoms   <int> 9, 23, 19, 14, 12, 14, 12, 10, 16, 14, 11, 22, 18, 12...
## $ single.bonds  <int> 9, 20, 22, 10, 16, 19, 14, 12, 21, 16, 13, 26, 23, 17...
## $ triple.bonds  <int> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ reactive      <int> 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,...
## $ molar.mass    <dbl> 118.135, 340.692, 252.313, 197.159, 162.188, 188.226,...
## $ refractivity  <dbl> 3.557232, 7.729887, 7.799344, 4.729193, 4.651153, 5.5...
## $ formal.charge <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ logP          <dbl> 1.425370, 3.637884, 3.485670, 3.565100, 1.942370, 2.3...
## $ tpsa          <dbl> 17.070000, 87.720001, 26.299999, 12.890000, 26.299999...
## $ dipole.moment <dbl> 2.7227166, 6.3207178, 1.9110060, 5.2036252, 1.7598478...
## $ energy        <dbl> -57.76601, -189.06262, -123.45357, -119.71445, -83.47...
## $ density       <dbl> 0.9589853, 1.2962539, 0.9463216, 1.1952425, 0.9782332...
## $ volume        <dbl> 123.1875, 262.8281, 266.6250, 164.9531, 165.7969, 193...
## $ PCA1          <dbl> 15.5507, 1.9216, 2.9803, 11.0316, 10.9710, 8.9700, 10...
## $ PCA2          <dbl> 1.6219, 0.5024, 2.7226, 1.8263, -0.9051, 0.3246, 0.75...
```

```
summary(mp)
```

```
##  ï..structure             mp            rings         heavy.atoms
##  Length:4450        Min.   : 14.0   Min.   : 0.000   Min.   : 6.00
##  Class :character   1st Qu.:117.5   1st Qu.: 6.000   1st Qu.:17.00
##  Mode  :character   Median :161.5   Median :11.000   Median :22.00
##                     Mean   :165.3   Mean   : 9.956   Mean   :22.24
##                     3rd Qu.:209.4   3rd Qu.:12.000   3rd Qu.:26.00
##                     Max.   :392.5   Max.   :36.000   Max.   :59.00
##
##   single.bonds     triple.bonds       reactive        molar.mass
##  Min.   :  4.00   Min.   :0.0000   Min.   :0.0000   Min.   : 84.08
##  1st Qu.: 19.00   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:243.27
##  Median : 24.00   Median :0.0000   Median :0.0000   Median :308.33
##  Mean   : 28.24   Mean   :0.1058   Mean   :0.2209   Mean   :317.41
##  3rd Qu.: 33.00   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:375.38
##  Max.   :119.00   Max.   :4.0000   Max.   :1.0000   Max.   :815.62
##
##   refractivity    formal.charge           logP             tpsa
##  Min.   : 1.992   Min.   :-3.00000   Min.   :-6.023   Min.   :  0.00
##  1st Qu.: 6.609   1st Qu.: 0.00000   1st Qu.: 2.106   1st Qu.: 38.33
##  Median : 8.310   Median : 0.00000   Median : 3.221   Median : 61.07
##  Mean   : 8.514   Mean   : 0.01236   Mean   : 3.337   Mean   : 67.50
##  3rd Qu.:10.179   3rd Qu.: 0.00000   3rd Qu.: 4.529   3rd Qu.: 88.06
##  Max.   :19.354   Max.   : 2.00000   Max.   :12.780   Max.   :413.24
##
##  dipole.moment         energy           density           volume
##  Min.   :  0.000   Min.   :-489.00   Min.   :0.7945   Min.   : 78.89
##  1st Qu.:  2.375   1st Qu.:-186.18   1st Qu.:0.9812   1st Qu.:230.77
##  Median :  3.811   Median :-150.42   Median :1.0540   Median :285.19
##  Mean   :  5.210   Mean   :-157.51   Mean   :1.0804   Mean   :294.47
##  3rd Qu.:  5.633   3rd Qu.:-118.36   3rd Qu.:1.1432   3rd Qu.:349.31
##  Max.   :248.303   Max.   : -44.35   Max.   :1.8956   Max.   :728.58
##                    NA's   :13
##       PCA1               PCA2
##  Min.   :-33.7975   Min.   :-19.9754
##  1st Qu.: -4.1177   1st Qu.: -2.8547
##  Median :  0.6655   Median :  0.3189
##  Mean   :  0.0000   Mean   :  0.0000
##  3rd Qu.:  4.6769   3rd Qu.:  3.2613
##  Max.   : 17.3486   Max.   : 12.4899
##
```

Then I noticed that when the energy was missing the dipole moment was 0. So I changed the 0s to N/A's and replaced them. ### Replacing N/As Then to check for missing data I ran each column under a na check that looks like the following

```
any(is.na(mp$structure))#F
```

```
## [1] FALSE
```

```
any(is.na(mp$mp))#F
```

```
## [1] FALSE
```

```
any(is.na(mp$rings))#F
```

```
## [1] FALSE
```

```
#ETC.....
```

I discovered that the only column with missing data was the energy column. So, I replaced the missing calues with the mean of the column using the following code.

```
mp$energy <- ifelse(is.na(mp$energy), mean(mp$energy, na.rm=TRUE), mp$energy)
mp$dipole.moment[mp$dipole.moment == 0] <- NA
mp$dipole.moment <- ifelse(is.na(mp$energy), mean(mp$energy, na.rm = TRUE), mp$energy)
```

New table

I needed to somewhat replicate table one from the reading, which I noticed caontained data from the summary command. I created an empty table with arbitrary values to start with.

```
table1data <- data.frame(A= numeric(0), B= numeric(0)) #Creating a data frame with arbitrary numbe
rs so i can easily add things
table1data[1:6,] <- c(5,23) #filling in spots so I can add
```

Next I added the summary statistics from each column I needed. But first I created a list of names that would become the column titles.

```
names<- c("Min", "1 Quartile", "2 Quartile", "Mean", ' 3 Quartile', 'Max') #Will be thre names for
the columns
#
table1data <- mutate(table1data, row = names)  #Actually adding what I need
table1data <- mutate(table1data, Melting_Point_C = summary(mp$mp))
table1data <- mutate(table1data, Molecular_weight =summary(mp$molar.mass))
table1data <- mutate(table1data, Num_heavy_atoms=summary(mp$heavy.atoms))
table1data <- mutate(table1data, SlogP=summary(mp$logP))
table1data <- mutate(table1data, Molar_refractivity=summary(mp$refractivity))
table1data <- mutate(table1data, dipole_moment=summary(mp$dipole.moment))
```

Finally I got rid of the two ‘dummy’ rows, A and B.

```
table1data$A <- NULL
table1data$B <-NULL
```

This is the new data set:

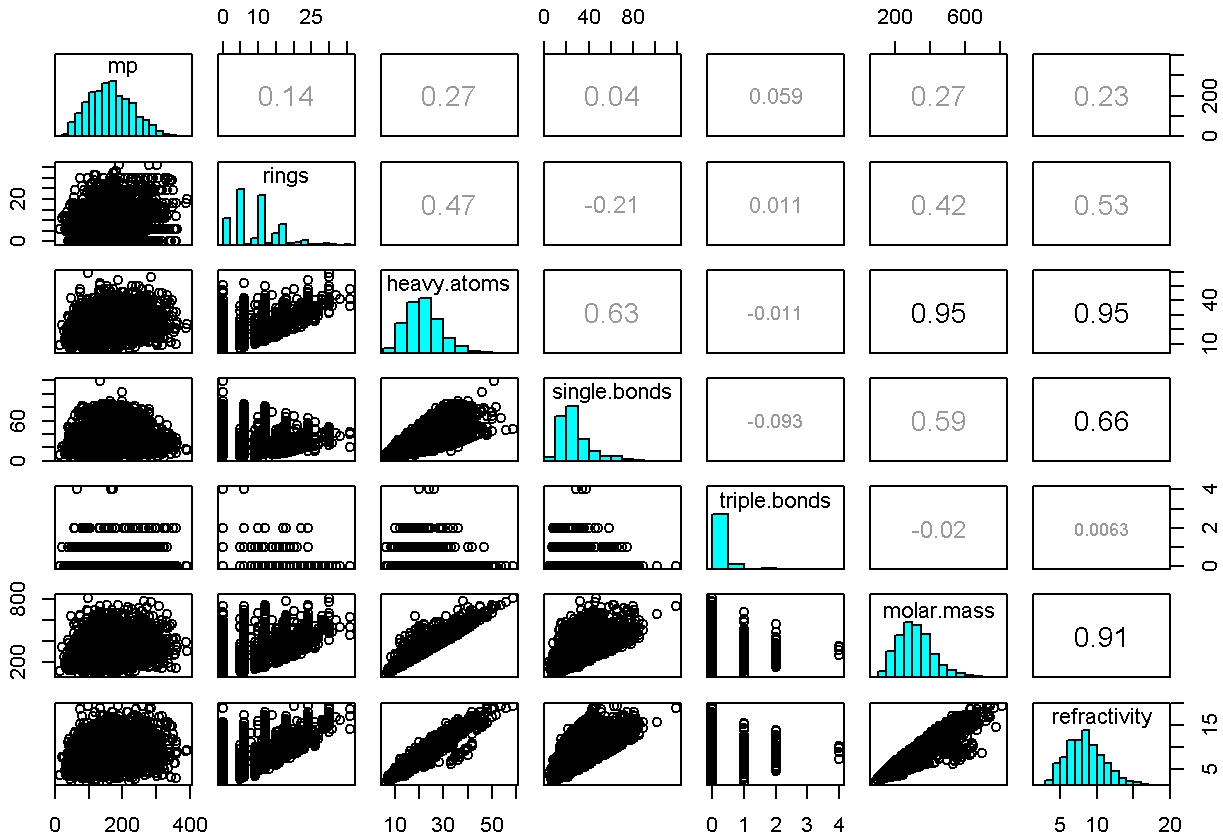
```
table1data
```

```
##           row Melting_Point_C Molecular_weight Num_heavy_atoms     SlogP
## 1         Min         14.0000          84.0780         6.00000 -6.023260
## 2  1 Quartile        117.5000         243.2710        17.00000  2.106125
## 3  2 Quartile        161.5000         308.3345        22.00000  3.220600
## 4        Mean        165.2896         317.4089        22.23506  3.337355
## 5  3 Quartile        209.3750         375.3785        26.00000  4.529044
## 6         Max        392.5000         815.6210        59.00000 12.780020
##   Molar_refractivity dipole_moment
## 1           1.992427     -488.9954
## 2           6.608528     -186.1653
## 3           8.309703     -150.6026
## 4           8.514301     -157.5131
## 5          10.179192     -118.4209
## 6          19.353979      -44.3485
```

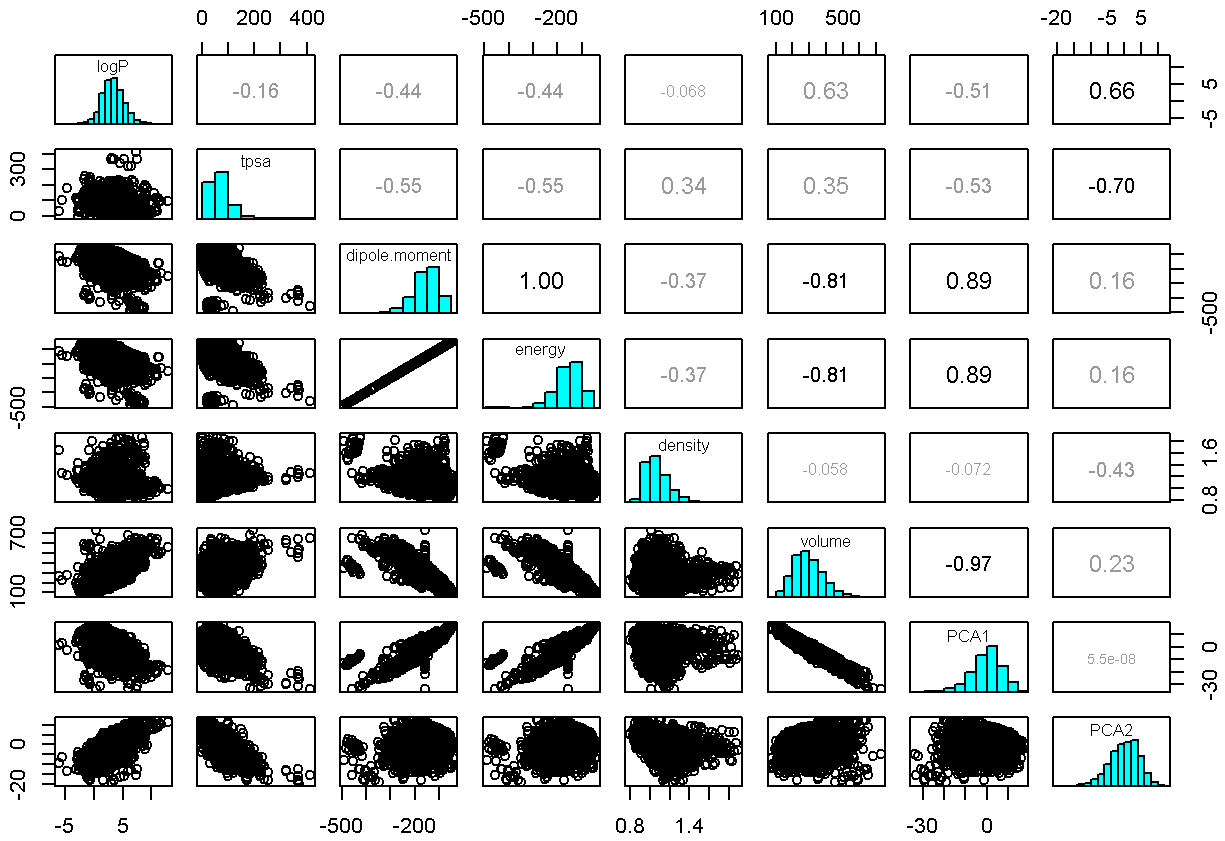
Analysis

To start my analysis I always like to do some pairwise plots to just get a grasp on the data. With this many columns one pairwise plot is too big so I split the data in half and just did two pairwise plots.

```
numerics1 <-mp[,c(2:6,8:9)] #small first half
numerics2 <-mp[,c(11:18)] #second half
pairs(numerics1,upper.panel = panel.cor,diag.panel=panel.hist)
```




```
pairs(numerics2,upper.panel = panel.cor,diag.panel=panel.hist)
```



Linear regressions

Next I needed to do some linear regressions of the three 2D descriptors. I first did melting point and the dformal charge of the compound and found that the charge has a negative corelation with the melting point.

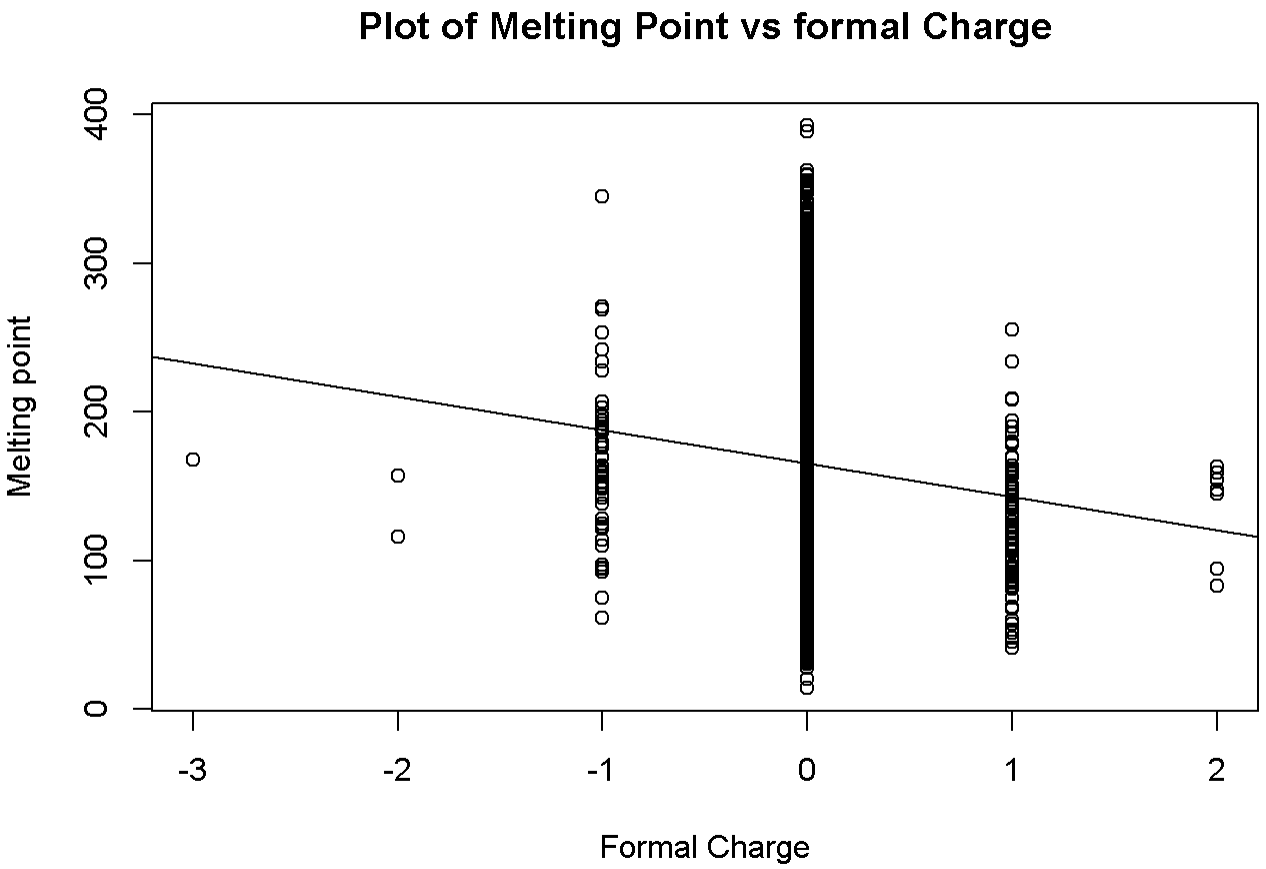
```
mpxcharge <- lm(mp$mp~mp$formal.charge) #comparing melting point and formal charge
mpxcharge
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$formal.charge)
##
## Coefficients:
##      (Intercept)  mp$formal.charge
##           165.57            -22.46
```

```
summary(mpxcharge)
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$formal.charge)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -151.567  -47.567   -3.567   43.933  226.933
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      165.5672     0.9602 172.430  < 2e-16 ***
## mp$formal.charge -22.4622     4.5180  -4.972 6.89e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 63.94 on 4448 degrees of freedom
## Multiple R-squared:  0.005526,   Adjusted R-squared:  0.005303
## F-statistic: 24.72 on 1 and 4448 DF,  p-value: 6.885e-07
```

```
plot(mp$mp~mp$formal.charge, main = "Plot of Melting Point vs formal Charge",        #doing an act
ual plot
     xlab = "Formal Charge", ylab = "Melting point")
abline(mpxcharge)
```




```
mpxcharge$coefficients
```

```
##      (Intercept) mp$formal.charge
##        165.56722        -22.46217
```

```
mpxchargefit <- mpxcharge$fitted.values #Just to take a look
#melting point has a negative correlation with formal charge
```

Next I did melting pont and volume and found there to be a positive correlation.

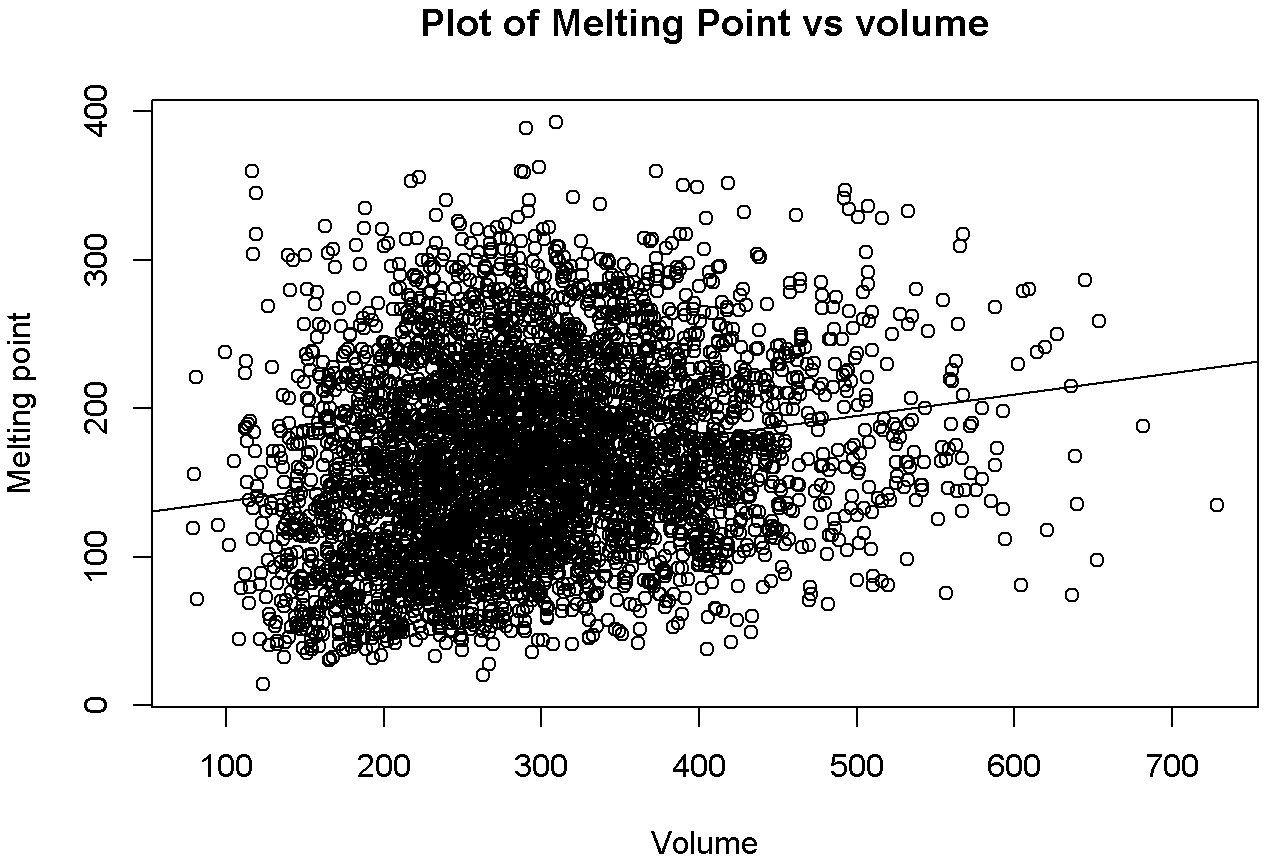
```
#Doing mp and  volume
mpxvol <- lm(mp$mp~mp$volume) #comparing melting point and formal charge
mpxvol
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$volume)
##
## Coefficients:
## (Intercept)    mp$volume
##    122.9602       0.1437
```

```
summary(mpxvol)
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$volume)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -143.177  -47.840   -5.304   42.298  225.089
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 122.96017    3.18370   38.62   <2e-16 ***
## mp$volume     0.14375    0.01033   13.92   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 62.77 on 4448 degrees of freedom
## Multiple R-squared:  0.04173,    Adjusted R-squared:  0.04151
## F-statistic: 193.7 on 1 and 4448 DF,  p-value: < 2.2e-16
```

```
plot(mp$mp~mp$volume, main = "Plot of Melting Point vs volume",        #doing an actual plot
     xlab = "Volume", ylab = "Melting point")
abline(mpxvol)
```




```
mpxvol$coefficients
```

```
## (Intercept)   mp$volume
## 122.9601674   0.1437456
```

```
mpxvolfit <- mpxvol$fitted.values
#There is a positive correlation between melting point and volume
```

And finally I did refractivity and melting point and found, again, a positive correlation.

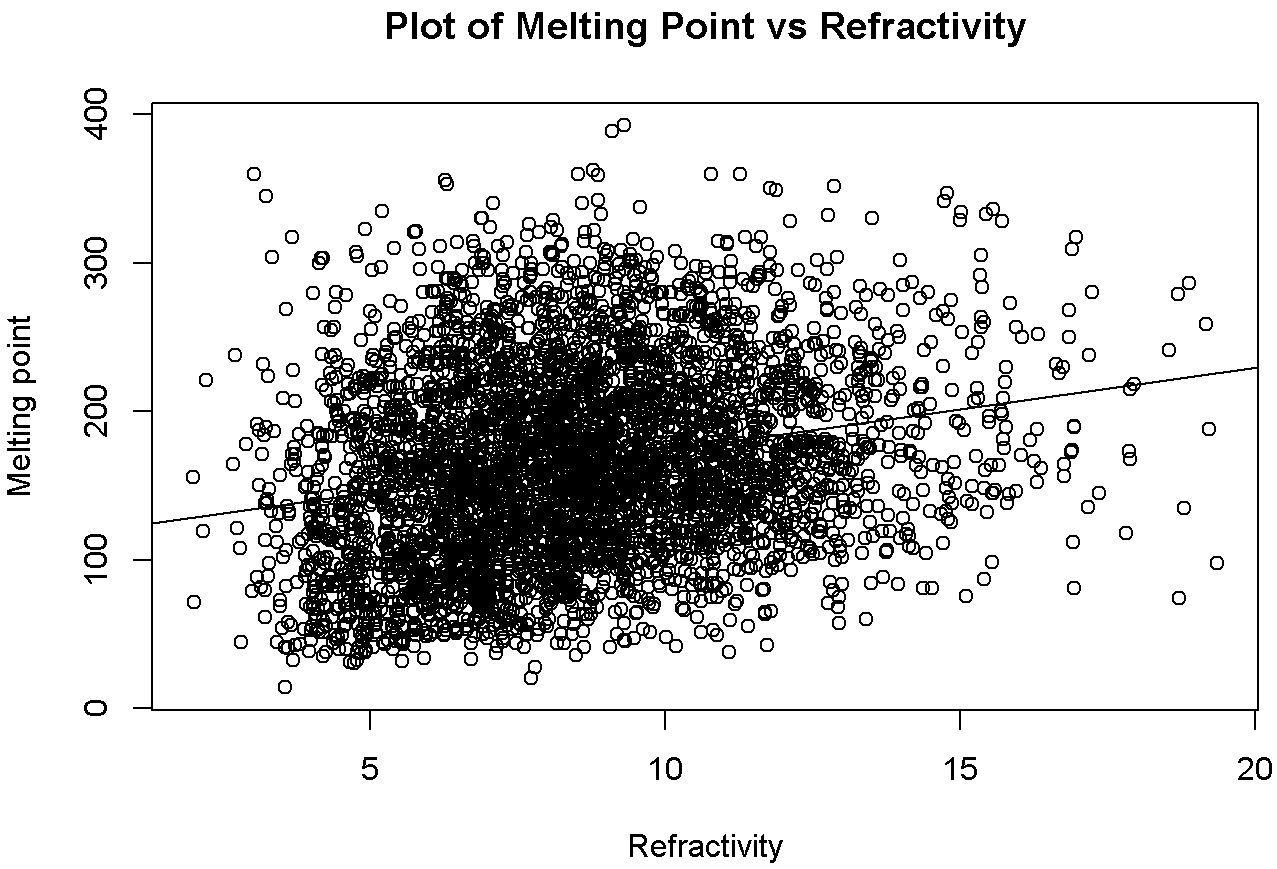
```
#Now doing refracvtivity and melting point
mpxref <- lm(mp$mp~mp$refractivity) #comparing melting point and formal charge
mpxref
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$refractivity)
##
## Coefficients:
##     (Intercept)  mp$refractivity
##         117.474            5.616
```

```
summary(mpxref)
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$refractivity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -148.530  -47.402   -5.525   41.857  225.477
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     117.4745     3.1181   37.67   <2e-16 ***
## mp$refractivity   5.6159     0.3494   16.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 62.34 on 4448 degrees of freedom
## Multiple R-squared:  0.05489,    Adjusted R-squared:  0.05468
## F-statistic: 258.4 on 1 and 4448 DF,  p-value: < 2.2e-16
```

```
plot(mp$mp~mp$refractivity, main = "Plot of Melting Point vs Refractivity",        #doing an actua
l plot
     xlab = "Refractivity", ylab = "Melting point")
abline(mpxref)
```




```
mpxref$coefficients
```

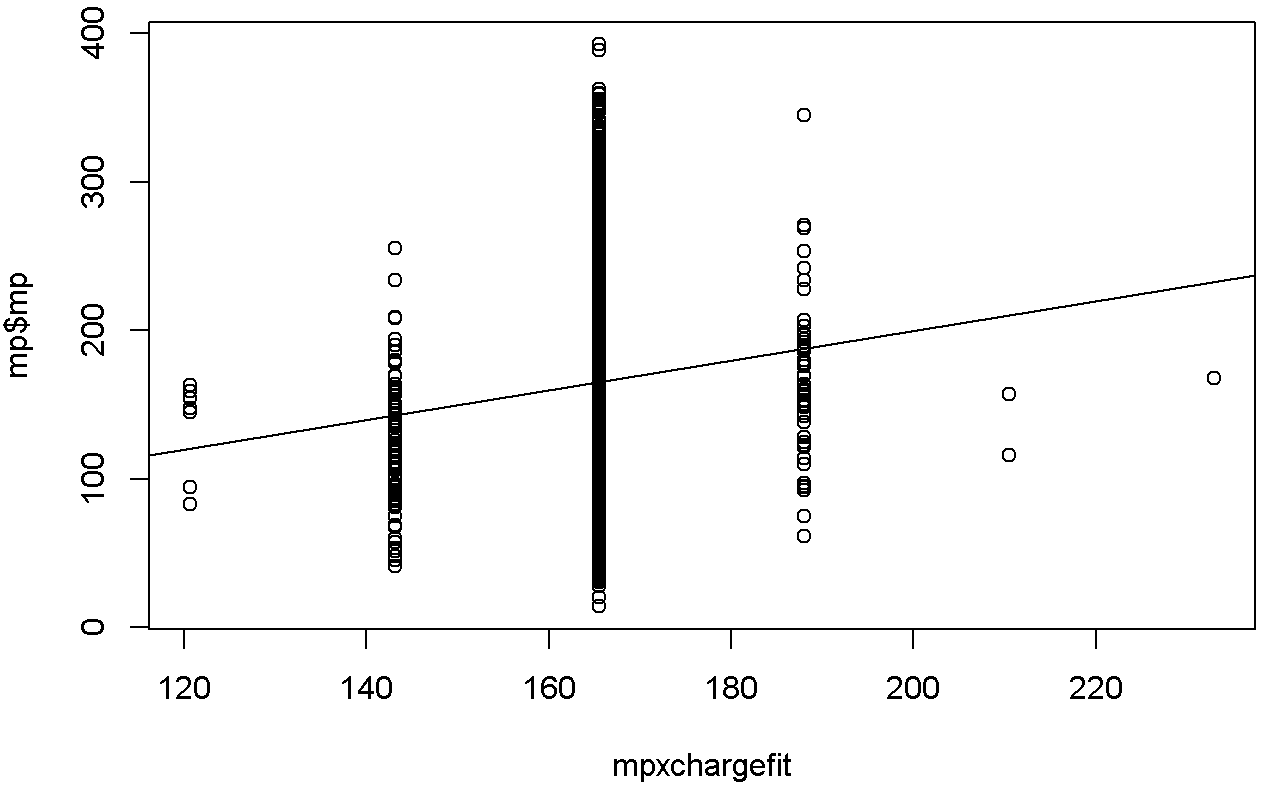
```
##     (Intercept) mp$refractivity
##      117.474495        5.615857
```

```
mpxreffit <- mpxref$fitted.values
#There is a positive corelation between refractivity and melting point
```

Expirimental vs predicted

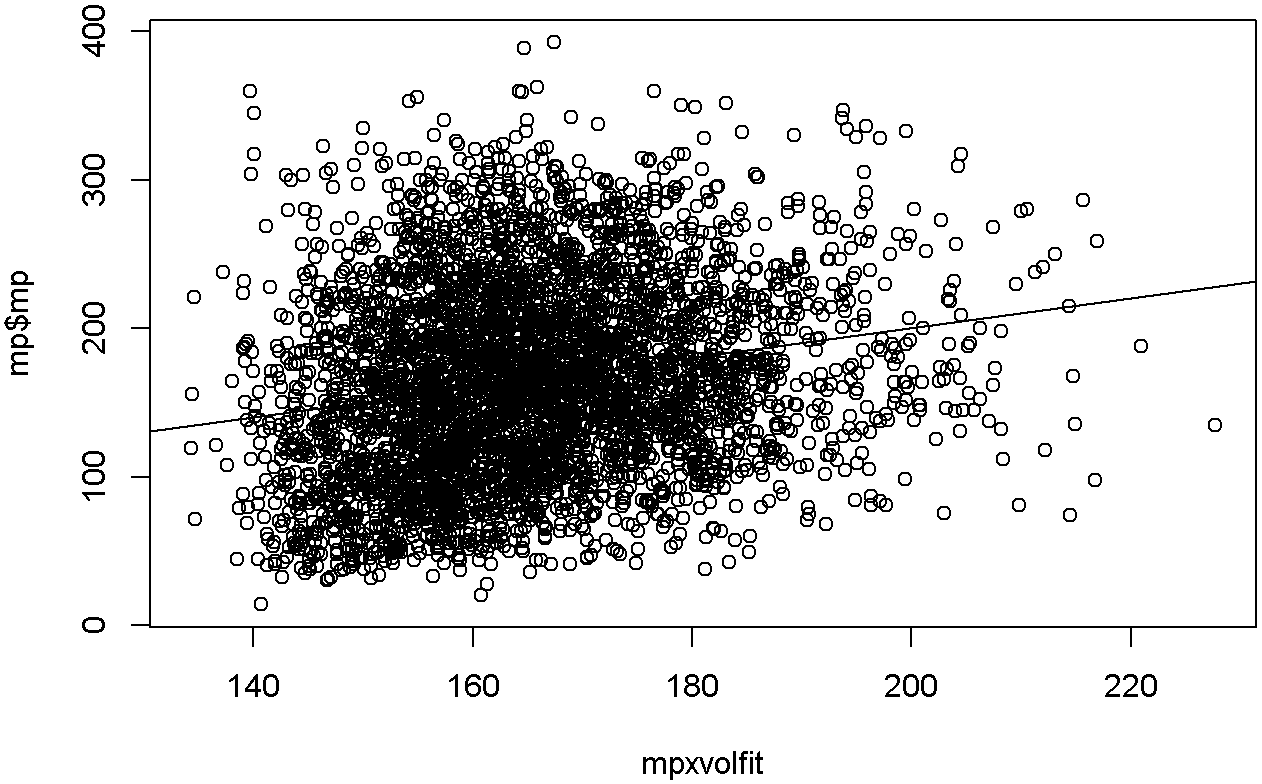
Next I had to do a plot of the expirimential (from the data) values and the predicted (fitted) values from the regressions I had done. First I did charge and melting point

```
#charge and mp
mpmpchargefit <- lm(mp$mp~mpxchargefit)
plot(mp$mp~mpxchargefit)
abline(mpmpchargefit)
```



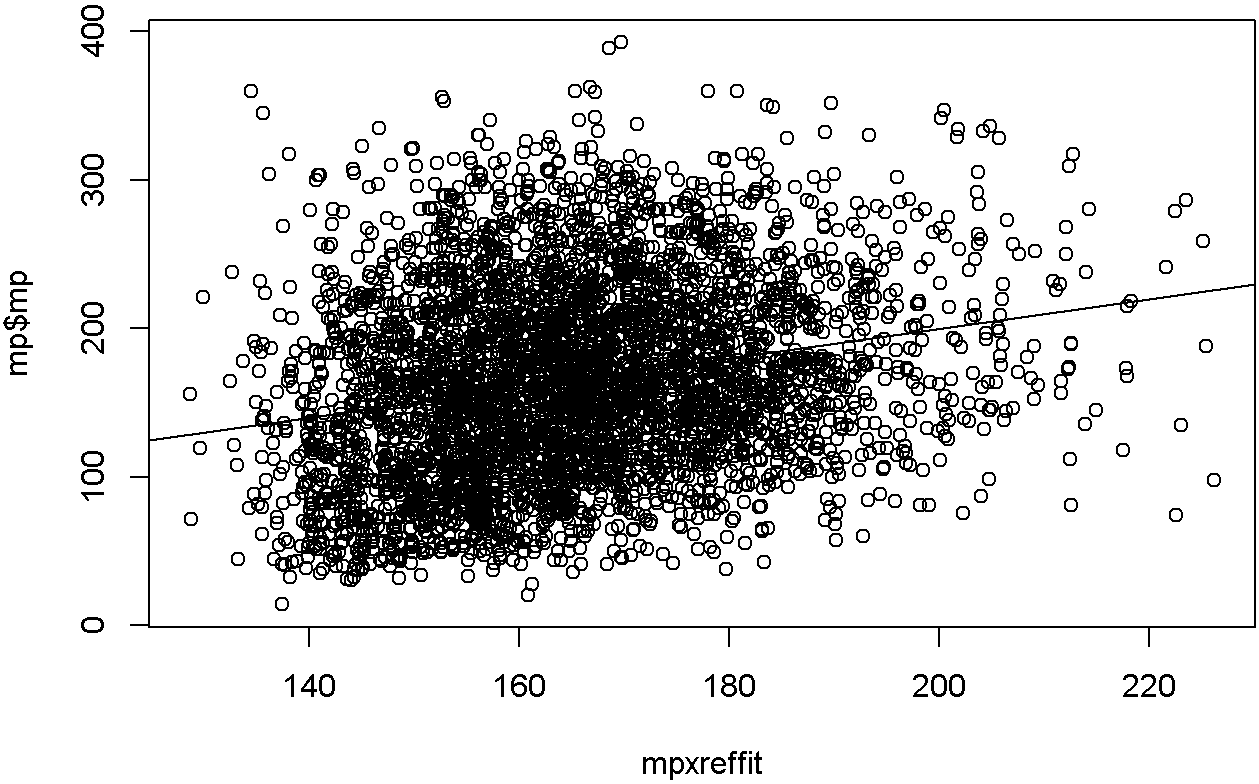
Then I did volume and melting point.

```
#mp/volume
mpmpvolfit <- lm(mp$mp~mpxvolfit)
plot(mp$mp~mpxvolfit)
abline(mpmpvolfit)
```



Finally did refractivity and melting point.

```
#And finally refractivity
mpmpreflfit <- lm(mp$mp~mpxreffit)
plot(mp$mp~mpxreffit)
abline(mpmpreflfit)
```



MultiFit

Regression Next I did a multifit regression to look at how all of those correlate to melting point.

```
#Multiple linear regression
multifit <- lm(mp$mp~mp$formal.charge+mp$volume+mp$refractivity)
multifit
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$formal.charge + mp$volume + mp$refractivity)
##
## Coefficients:
##      (Intercept)  mp$formal.charge         mp$volume   mp$refractivity
##         123.0117          -22.7805           -0.7187           29.8553
```

```
summary(multifit)
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$formal.charge + mp$volume + mp$refractivity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -151.529  -46.495   -5.889   42.009  230.036
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      123.01167    3.10703  39.591  < 2e-16 ***
## mp$formal.charge -22.78054    4.33599  -5.254 1.56e-07 ***
## mp$volume         -0.71869    0.06285 -11.435  < 2e-16 ***
## mp$refractivity   29.85526    2.14004  13.951  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61.22 on 4446 degrees of freedom
## Multiple R-squared:  0.0888, Adjusted R-squared:  0.08818
## F-statistic: 144.4 on 3 and 4446 DF,  p-value: < 2.2e-16
```

BIC Analysis

Now to look at causation I performed a BIC analysis on melting point in reference to volume, and meltingpoint in reference to refractivity, with refractivify first.

```
#BIC analysis
#Does refractivity have an affect on Melting point?
refmel<-lm(mp$mp~mp$refractivity) #Comparing here to look at the summary
summary(refmel)
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$refractivity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -148.530  -47.402   -5.525   41.857  225.477
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     117.4745     3.1181   37.67   <2e-16 ***
## mp$refractivity   5.6159     0.3494   16.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 62.34 on 4448 degrees of freedom
## Multiple R-squared:  0.05489,    Adjusted R-squared:  0.05468
## F-statistic: 258.4 on 1 and 4448 DF,  p-value: < 2.2e-16
```

```
BIC(lm(mp$mp~1)) #Comparing to one
```

```
## [1] 49674.4
```

```
BIC(lm(mp$mp~mp$refractivity)) #looking at difference
```

```
## [1] 49431.56
```

```
#YES
```

There was an obvious difference that was greater than 10. So refractivity truly does affect the melting point of the compounds.

```
#Does volume have an affect on melting point
volmel<-lm(mp$mp~mp$volume) #Comparing here to look at the summary
summary(volmel)
```

```
##
## Call:
## lm(formula = mp$mp ~ mp$volume)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -143.177  -47.840   -5.304   42.298  225.089
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 122.96017    3.18370   38.62   <2e-16 ***
## mp$volume     0.14375    0.01033   13.92   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 62.77 on 4448 degrees of freedom
## Multiple R-squared:  0.04173,    Adjusted R-squared:  0.04151
## F-statistic: 193.7 on 1 and 4448 DF,  p-value: < 2.2e-16
```

```
BIC(lm(mp$mp~1)) #Comparing to one
```

```
## [1] 49674.4
```

```
BIC(lm(mp$mp~mp$volume)) #looking at difference
```

```
## [1] 49493.12
```

```
#Yes
```

Just like refractivity, volume also affects the melting point.

Reference

For reference I used the article General Melting Point Prediction Based on a Diverse Compound Data Set and Artificial Neural Networks by M. Karthikeyan.